

**Το μοντέλο της Ιεραρχικής Αξιολόγησης
Γνώσεων Προγραμματισμού:
μια εμπειρική μελέτη**

Στυλιανή Ζιώγα

Μεταπτυχιακή εργασία

2018

© Στυλιανή Ζιώγα



Πανεπιστήμιο Ιωαννίνων
Σχολή Επιστημών της Αγωγής
Παιδαγωγικό Τμήμα Δημοτικής Εκπαίδευσης

Πρόγραμμα Μεταπτυχιακών Σπουδών
«Επιστήμες της Αγωγής»
Κατεύθυνση: Φυσικές Επιστήμες στην Εκπαίδευση

Το μοντέλο της Ιεραρχικής Αξιολόγησης Γνώσεων Προγραμματισμού: μια εμπειρική μελέτη

Μεταπτυχιακή Διπλωματική εργασία

Στυλιανή Ζιώγα

A.M. 464

Συμβουλευτική επιτροπή

Επιβλέπων: Μικρόπουλος Αναστάσιος

Μέλη: Κώσης Κω/νος
Μαυρίδης Δημήτριος

Ιωάννινα 2018

ΠΕΡΙΛΗΨΗ

Ο προγραμματισμός είναι μια από τις βασικές και τις πιο δύσκολες δεξιότητες που χρειάζεται να αποκτήσουν οι μαθητές. Οι εκπαιδευτικοί που ασχολούνται με τη διδασκαλία του επιδιώκουν να βρουν τις καταλληλότερες τεχνικές αξιολόγησης ώστε να μπορούν να βελτιώσουν τη διδασκαλία τους. Η εργασία παρουσιάζει τις δυσκολίες του προγραμματισμού, περιγραφή των εκπαιδευτικών ταξινομιών Bloom και Solo και την εφαρμογή τους στον προγραμματισμό ως πλαίσιο για αξιολόγηση. Στη βιβλιογραφική επισκόπηση εντοπίζονται οι δυσκολίες στη χρήση τους. Στην παρούσα εργασία περιγράφεται το μοντέλο της Ιεραρχικής Αξιολόγησης Προγραμματισμού, το οποίο βασίζεται στην ταξινομία Solo, και διερευνά την αξιολόγηση των γνώσεων προγραμματισμού σε τελειόφοιτους μαθητές Λυκείου καθώς και σύγκριση των αποτελεσμάτων αξιολόγησης με την παραδοσιακή εμπειρική αξιολόγηση από εκπαιδευτικούς. Τα αποτελέσματα της έρευνας επιβεβαιώνουν ευρήματα που υπάρχουν στη διεθνή βιβλιογραφία για την αξιολόγηση των γνώσεων προγραμματισμού και τις δυσκολίες που εντοπίστηκαν. Από τα συμπεράσματα της έρευνας διαφαίνεται ότι η Ιεραρχική Αξιολόγηση Προγραμματισμού αποτελεί έναν αναλυτικό τρόπο ποιοτικής αξιολόγησης γνώσεων προγραμματισμού αλλά χρειάζεται επιπλέον διερεύνηση για τα δύο υψηλότερα επίπεδα.

Λέξεις κλειδιά: Προγραμματισμός, αξιολόγηση, ταξινομία Bloom, ταξινομία Solo, Ιεραρχική Αξιολόγηση Προγραμματισμού.

ABSTRACT

Programming is one of the basic and most difficult skills that students need to acquire. Teachers dealing with their teaching seek to find the most appropriate assessment techniques so they can improve their teaching. The paper presents the programming difficulties, describing the Bloom and Solo educational taxonomies and their application to programming as a framework for assessment. The bibliographic overview identifies the difficulties in their use. This paper describes the Model Hierarchical Assessment of Programming based on the Solo taxonomy and explores the evaluation of programming skills in the senior students of a high school and a comparison of the evaluation results with the traditional empirical assessment by teachers. The results of the survey confirm findings found in the international literature to assess the programming knowledge and the difficulties identified. Based on the findings of the survey, it appears that the Hierarchical Assessment of Programming is an analytical way of qualitative assessment of programming skills but needs further investigation for the two higher levels.

Key words: *Programming, assessment, Bloom's taxonomy, SOLO taxonomy, Hierarchical Assessment of Programming*

Ευχαριστίες

Πρωτίστως θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Αναστάσιο Μικρόπουλο για την αμέριστη ηθική συμπαράσταση, επιστημονική καθοδήγηση, τις συμβουλές και τις ιδέες του κατά τη συγγραφή της μεταπτυχιακής εργασίας όπως και τα μέλη της επιτροπής Κώτση Κων/νο και Μαυρίδη Δημήτριο για την άριστη συνεργασία και τις συμβουλές τους. Ακόμη, επιθυμώ να ευχαριστήσω τη Σχολική Σύμβουλο κ. Ιωάννα Μπέλλου για τις χρήσιμες συμβουλές και προτάσεις, καθώς και τους συναδέλφους εκπαιδευτικούς Αθανασιάδου Ε., Μιτράκα Β., Βακαλίδου Ε., Γκορόγια Δ., Μινέτου Ε. και Χρήστου Π., για το χρόνο που αφιέρωσαν στη βαθμολόγηση των γραπτών. Θα ήθελα, επίσης, να εκφράσω τις ευχαριστίες μου στους μαθητές της Γ' τάξης του Γενικού Λυκείου Ανατολής Σχ. Έτους 2017-18 που συμμετείχαν στην έρευνα. Τέλος, οφείλω να ευχαριστήσω την οικογένειά μου για την ενθάρρυνση, τη συμπαράστασή και την υπομονή τους καθ' όλη τη διάρκεια εκπόνησης της μεταπτυχιακής εργασίας.

Περιεχόμενα

ΠΕΡΙΛΗΨΗ.....	3
ABSTRACT.....	4
Ευχαριστίες	5
Εισαγωγή.....	8
A. Θεωρητική προσέγγιση	11
1. Προγραμματισμός	11
1.1 Η πρώτη γλώσσα προγραμματισμού	11
1.2 Ο ρόλος και η αναγκαιότητα του Αλγόριθμου	12
1.3 Η διδασκαλία του προγραμματισμού	13
1.4 Οι δυσκολίες στον προγραμματισμό	16
1.5 Η αξιολόγηση και οι δυσκολίες της	22
2. Ταξινόμηση εκπαιδευτικών στόχων	27
2.1 Η αρχική ταξινόμια Bloom	27
2.2 Η εφαρμογή των ταξινομιών Bloom στον προγραμματισμό	32
2.3 Δυσκολίες στη χρήση της ταξινόμιας Bloom	37
2.4 Ταξινόμια SOLO.....	40
2.5 Εφαρμογή της ταξινόμιας SOLO	42
3. Το εκπαιδευτικό πλαίσιο του Προγραμματισμού στο Γενικό Λύκειο στην Ελλάδα	51
B. Εμπειρική μελέτη.....	52
4. Μεθοδολογία.....	52
4.1 Ερευνητικά ερωτήματα	52
4.2 Το πλαίσιο της έρευνας	52
4.3 Δείγμα	54
4.4 Δραστηριότητες αξιολόγησης και ανάλυση δεδομένων.....	56
5. Αποτελέσματα	105
5.1 Αποτελέσματα ως προς το Επίπεδο Θυμάμαι.....	105
5.2 Αποτελέσματα ως προς το Επίπεδο Κατανόω.....	109
5.3 Αποτελέσματα ως προς το Επίπεδο Εφαρμόζω	116
5.4 Αποτελέσματα ως προς το Επίπεδο Αναλύω	124
5.5 Αποτελέσματα ως προς το Επίπεδο Αξιολογώ	131
5.6 Αποτελέσματα ως προς το Επίπεδο Δημιουργώ.....	139

6. Συμπεράσματα - Συζήτηση	144
Βιβλιογραφικές αναφορές.....	156
ΠΑΡΑΡΤΗΜΑ	164

Εισαγωγή

Το σχολείο είναι οργανισμός μάθησης αλλά πρωτίστως ένα κοινωνικό περιβάλλον στο οποίο ο εκπαιδευτικός ως φορέας που μεταδίδει τη γνώση αντιμετωπίζει καθημερινά πολλές προκλήσεις στην προσπάθειά του να διαμορφώσει το σχολείο του μέλλοντος, όπου η γνώση δεν θα είναι μόνο στατικά προσφερόμενη αλλά μεταφερόμενη στο πλαίσιο μιας δυναμικής και εξελισσόμενης διαδικασίας. Σε ένα τέτοιο περιβάλλον η επιστήμη της πληροφορικής διαδραματίζει έναν από τους σημαντικότερους ρόλους. Η επιστήμη των υπολογιστών επιδιώκει με συνεχή προσπάθεια να παρουσιάσει και να μεταφέρει χρήσιμες γνώσεις και επιπλέον να βοηθήσει τους μαθητές στην προσπάθειά τους για κριτική αξιολόγηση των πληροφοριών που παρουσιάζονται (Mallios & Vassilakopoulos, 2015).

Ο Προγραμματισμός ως γνωστική δραστηριότητα και ταυτόχρονα ως αντικείμενο διδασκαλίας προκαλεί εξαιρετικό ενδιαφέρον και ταυτόχρονα αποτελεί ένα ιδιαίτερα σύνθετο εννοιολογικό πεδίο (Κόμης, 2001). Θεωρείται ένα ισχυρό μέσο για την ανάπτυξη δεξιοτήτων υψηλού επιπέδου (ανάλυση - σύνθεση, διατύπωση συνθηκών κλπ), για τη διδασκαλία βασικών εννοιών που βρίσκουν εφαρμογή στα μαθηματικά, τη φυσική και τη λογική, καθώς και τη μεταφορά δεξιοτήτων επίλυσης προβλημάτων σε άλλα γνωστικά αντικείμενα.

Υπάρχει ένα μακροχρόνιο και έντονο ενδιαφέρον για την ανάπτυξη του προγραμματισμού στην εκπαίδευση και στην έρευνα. Οι περισσότερες έρευνες αφορούν τις μεθόδους διδασκαλίας, την εκπαιδευτική τεχνολογία, την κατανόηση καθώς και τις παρανοήσεις των μαθητών. Συνήθως, η αξιολόγηση του αποτελέσματος μάθησης γίνεται κατά τη διάρκεια ή αμέσως μετά την εκπαιδευτική δραστηριότητα. Υπάρχει πολύ έρευνα στην εισαγωγική εκπαίδευση στον προγραμματισμό. Σε γενικές γραμμές επικεντρώνεται στην επίδοση των μαθητών ή τη συμπεριφορά τους με βάση τα δεδομένα που συλλέχθηκαν κατά τη διάρκεια ή αμέσως μετά το μάθημα με εστίαση τα μαθησιακά αποτελέσματα, τα εμπόδια μάθησης, τις παρανοήσεις και τους δείκτες επιτυχίας (Bennedsen & Caspersen, 2012). Γενικά η επιτυχία καθορίζεται από το πόσο καλά οι μαθητές ανταποκρίνονται στις εξετάσεις κατά τη διάρκεια ή στο τέλος του μαθήματος.

Πολλά εισαγωγικά μαθήματα Πληροφορικής έχουν τον προγραμματισμό ως βασική δραστηριότητα και στόχο μάθησης. Από παιδαγωγική άποψη, θεωρείται ευεργετικός στη βελτίωση του υψηλού επιπέδου σκέψης για την ανάπτυξη δεξιοτήτων επίλυσης αλγοριθμικών προβλημάτων (DiSessa, 2000; Papert, 1983). Από διδακτική άποψη, ο προγραμματισμός αποτελεί βασική δεξιότητα που επιτρέπει στους μαθητές να εμβαθύνουν την κατανόησή τους σε διάφορους τομείς γνώσης που σχετίζονται με την Πληροφορική.

Τα περιβάλλοντα προγραμματισμού θεωρούνται ισχυρά εργαλεία που βοηθούν τους μαθητές να αναλύουν, να οργανώνουν, να εκφράζουν και να αξιολογούν τις σκέψεις τους με τρόπο σαφή και

συνοπτικό κατά τη διάρκεια της διαδικασίας επίλυσης προβλημάτων. Ενώ προσπαθούν να δείξουν στον υπολογιστή πώς να λύσει ένα πρόβλημα, οι μαθητές έχουν την ευκαιρία να παρακολουθήσουν τα αποτελέσματα, να αποσαφηνίσουν τη διαδικασία σκέψης τους και να λάβουν άμεση ανατροφοδότηση, μια διαδικασία που διευκολύνει την ανάπτυξη όχι μόνο των προβλημάτων αλλά και μεταγνωστικές δεξιότητες (Clements & Nastasi, 1999).

Ελάχιστες πληροφορίες υπάρχουν σχετικά με τον τρόπο εισαγωγής και αξιολόγησης του προγραμματισμού σε μαθητές δευτεροβάθμιας εκπαίδευσης. Οι εκπαιδευτικοί και οι υπεύθυνοι που σχεδιάζουν τα Προγράμματα Σπουδών Προγραμματισμού πρέπει να γνωρίζουν και να δίνουν ιδιαίτερη προσοχή σε κάθε πρόκληση που αντιμετωπίζουν οι φοιτητές/μαθητές (Paranlasorouliou, Giannakos, & Jaccheri, 2015).

Κρίνεται αναγκαίο να δοθεί μια περιγραφή της φιλοσοφίας, των πεποιθήσεων, των αξιών και των προοπτικών στο μάθημα του προγραμματισμού μέσω του οποίου οι μαθητές μαθαίνουν δεξιότητες προγραμματισμού. Σε διεθνές επίπεδο, υπάρχουν διαφορές στα εισαγωγικά μαθήματα προγραμματισμού, γεγονός που επαληθεύεται από τη συντριπτική πλειοψηφία των διδακτικών εγχειριδίων. Έτσι η διδασκαλία του προγραμματισμού δεν ακολουθεί τις ίδιες προδιαγραφές σε όλα τα μαθήματα προγραμματισμού (Bennedsen & Caspersen, 2012).

Τα επιθυμητά μαθησιακά αποτελέσματα καθορίζονται και ελέγχονται από την απόκτηση ορισμένων στοιχείων γνώσης ή από την επίτευξη προκαθορισμένων μαθησιακών στόχων. Προκειμένου να αξιολογηθούν τα μαθησιακά αποτελέσματα, απαιτούνται κατάλληλα καθορισμένα και έγκυρα μοντέλα δεξιοτήτων. Το επίκεντρο των αποτελεσμάτων της σχολικής εκπαίδευσης μετατοπίστηκε ολοένα και περισσότερο προς την κατεύθυνση των μαθησιακών στόχων σε πολλές χώρες. Για να εναρμονιστούν οι διαδικασίες μάθησης και διδασκαλίας και να μετρηθεί η επιτυχία, αυτές οι ικανότητες-στόχοι πρέπει να καθοριστούν και να διαρθρωθούν σωστά με κατάλληλα εμπειρικά αποδεκτά μοντέλα δεξιοτήτων. Για το σκοπό αυτό, χρησιμοποιούνται διάφορα είδη μοντέλων, τα οποία μπορούν να επικεντρωθούν στη δομή, στα διαφορετικά ιεραρχικά επίπεδα ή στην ανάπτυξη των σχετικών δεξιοτήτων (Mühling, Hubwieser, & Berges, 2015). Έτσι, η έρευνα αυτή εστιάζει στη διερεύνηση των καταλληλότερων μοντέλων δεξιοτήτων, τα οποία αξιολογούν γνώσεις προγραμματισμού.

Στην συγκεκριμένη έρευνα περιγράφονται δύο εκπαιδευτικές ταξινομίες. Πρώτα η αναθεωρημένη ταξινομία Bloom η οποία ταξινομεί τους μαθησιακούς στόχους σύμφωνα με τη γνωστική πολυπλοκότητα και χρησιμοποιήθηκε στην έρευνα για την δημιουργία των δραστηριοτήτων στις οποίες αξιολογήθηκαν οι μαθητές. Στη συνέχεια η ταξινομία SOLO, η οποία ταξινομεί τα μαθησιακά αποτελέσματα σύμφωνα με τα πέντε επίπεδα αυξανόμενης δομικής πολυπλοκότητας. Στην

παρούσα έρευνα για την αξιολόγηση των γνώσεων των μαθητών χρησιμοποιείται το μοντέλο της Ιεραρχικής Αξιολόγησης Προγραμματισμού το οποίο βασίζεται στην ταξινομία SOLO.

Η παρούσα Μεταπτυχιακή Διπλωματική Εργασία αποτελείται από δύο μέρη: το πρώτο συνιστά το θεωρητικό πλαίσιο, ενώ το δεύτερο περιλαμβάνει τη μεθοδολογία και την υλοποίηση της πραγματοποιηθείσας έρευνας, την αναλυτική παρουσίαση των αποτελεσμάτων που προέκυψαν, καθώς επίσης και τη διατύπωση συμπερασμάτων και προτάσεων για μελλοντική έρευνα.

A. Θεωρητική προσέγγιση

1. Προγραμματισμός

1.1 Η πρώτη γλώσσα προγραμματισμού

Σήμερα, μεταξύ άλλων προκλήσεων, η διδασκαλία των μαθητών για το πώς να γράψουν προγράμματα υπολογιστών για πρώτη φορά μπορεί να είναι ένα σημαντικό κριτήριο για το αν θα ασχοληθούν με τον τομέα της πληροφορικής. Ο προγραμματισμός είναι μια θεμελιώδης δεξιότητα για τον ειδικό στην Πληροφορική. Επομένως, ιδιαίτερης σημασίας είναι οι μαθητές να μάθουν να προγραμματίζουν σωστά χωρίς να αποθαρρύνονται (Karlan, 2010).

Κατά τη διάρκεια των ετών, οι μελέτες εξελίσσονται ανάλογα με τις διάφορες τοπικές και παγκόσμιες απαιτήσεις και τάσεις στον τομέα της διδασκαλίας προγραμματισμού. Το πιο σοβαρό πρόβλημα που παρατηρήθηκε ήταν η επιλογή της πρώτης γλώσσας προγραμματισμού καθώς είναι σημαντική για την μεταγενέστερη ανάπτυξη ενός προγραμματιστή (Karlan, 2010; Savić, Ivanoni, Budimac, & Radovanović, 2016). Ένας από τους λόγους είναι η ύπαρξη πολλών γλωσσών προγραμματισμού που οι περισσότερες ανταγωνίζονται για την ευρεία χρήση τους στη βιομηχανία και σε εφαρμογές στην καθημερινή ζωή.

Υπάρχουν στη βιβλιογραφία πολλές έρευνες για την αποτελεσματικότητα και τα χαρακτηριστικά μιας γλώσσας προγραμματισμού η οποία να ανταποκρίνεται στις εκπαιδευτικές ανάγκες των εκπαιδευτικών και των εκπαιδευόμενων. Από τη μία πλευρά οι εκπαιδευτικοί πρέπει να είναι σε θέση να διδάξουν τις θεμελιώδεις έννοιες της επιστήμης των υπολογιστών και η γλώσσα προγραμματισμού πρέπει να μπορεί να υποστηρίξει αυτές τις θεμελιώδεις έννοιες. Από την άλλη πλευρά, μια γλώσσα προγραμματισμού για αρχάριους πρέπει να επιτρέπει στον μαθητή να μάθει αυτές τις έννοιες χωρίς απαραίτητα να μάθει όλες τις λεπτομέρειες της γλώσσας προγραμματισμού. Ο προγραμματισμός είναι μια δεξιότητα που περιλαμβάνει την επίλυση προβλημάτων καθώς και τη δημιουργία αλγορίθμων. Αυτά τα χαρακτηριστικά πρέπει να ενσωματώνονται σε μια γλώσσα προγραμματισμού (Karlan, 2010).

Οι εκπαιδευτικοί χρειάζεται συνεχώς να κάνουν επανεκτίμηση της επιλογής της πρώτης γλώσσας προγραμματισμού και της μεθοδολογίας διδασκαλίας της. Ως συνέπεια της ταχείας ανάπτυξης στον χώρο της Πληροφορικής, οι προγραμματιστές χρειάζεται να είναι σε θέση να καλύψουν πολλές γλώσσες προγραμματισμού. Ποια γλώσσα προγραμματισμού θα επιλέξει να χρησιμοποιήσει ο εκπαιδευτικός στη διδασκαλία του προγραμματισμού είναι δύσκολη απόφαση επειδή όλες έχουν πλεονεκτήματα καθώς και μειονεκτήματα. Οι Savić et al. (2016) με την έρευνα τους υποστηρίζουν ότι η επιλογή της πρώτης γλώσσας προγραμματισμού δεν επηρεάζει σε βάθος την επιτυχία των μαθητών στο μάθημα προγραμματισμού.

1.2 Ο ρόλος και η αναγκαιότητα του Αλγόριθμου

Οι επιστήμονες από τη δεκαετία του 1970 έδωσαν ιδιαίτερη προσοχή στο εκπαιδευτικό περιεχόμενο του προγραμματισμού. Επεσήμαναν ότι η έννοια του αλγόριθμου βρίσκεται στη βάση του προγραμματισμού, που ορίζεται ως μια διαδικασία ανάπτυξης και περιγραφής ενός αλγορίθμου με τη βοήθεια μιας συγκεκριμένης γλώσσας προγραμματισμού. Πολλές ανθρώπινες δραστηριότητες μπορούν να περιγραφούν ως διαδικασία η οποία ακολουθεί έναν συγκεκριμένο αλγόριθμο παρότι υπάρχουν τομείς που ο αλγόριθμος δεν είναι εφαρμόσιμος. Η έννοια του αλγόριθμου σχηματίζεται έμμεσα κατά τη διάρκεια της μελέτης διαφόρων επιστημονικών κλάδων. Από τα μέσα της δεκαετίας του 1980, πρωταρχικό καθήκον των μαθημάτων προγραμματισμού ήταν ο σχηματισμός αλγοριθμικής κουλτούρας των μαθητών. Στην πραγματικότητα, η ικανότητα προσδιορισμού, συνδυασμού, επαλήθευσης και εκτέλεσης σωστά ενός μαθηματικού αλγορίθμου αποτελούσε πάντα ένα σημαντικό συστατικό στοιχείο της μαθηματικής κουλτούρας του μαθητή. Σε αυτό το πλαίσιο, η έλλειψη λειτουργικής παιδείας, η οποία είναι η δεξιότητα εξαγωγής και ερμηνείας πληροφοριών από ένα κείμενο, θα μπορούσε να δυσχεράνει την οικοδόμηση ενός σωστού αλγορίθμου και προγράμματος.

Μια άλλη παράμετρος που αφορά τα εισαγωγικά μαθήματα Προγραμματισμού είναι ότι το σύνολο των δραστηριοτήτων και των παραδειγμάτων που χρησιμοποιούνται είναι αρκετά απλά. Έτσι οι σπουδαστές δεν μπορούν να προσεγγίσουν το πρόβλημα δημιουργώντας αποτελεσματικές λύσεις. Ως εκ τούτου, σημαντικό μεθοδολογικό ζήτημα είναι η χρήση μιας προσεκτικής επιλογής δραστηριοτήτων και παραδειγμάτων, που να παρακινούν τους φοιτητές/μαθητές να προσεγγίσουν τα προβλήματα που εξετάζουν χρησιμοποιώντας διαφορετικούς εναλλακτικούς αλγορίθμους πριν τη δημιουργία προγράμματος (Teodosiev & Nachev, 2015).

Συχνά, οι αρχάριοι δημιουργούν προγράμματα μόλις τους δοθεί ένα πρόβλημα, προσπαθώντας να βρουν άμεσα μια σωστή λύση παραβλέποντας έτσι το στάδιο ανάλυσης αλγορίθμων, που είναι σημαντικό για την εφαρμογή ενός «καλού» αλγορίθμου (Chakraborty, Saxena, & Katti, 2013; Sahoo & Chin, 2010). Οι σπουδαστές συνήθως ασχολούνται μόνο με ένα πρόγραμμα στην εργασία τους, ανεξάρτητα από το εάν είναι αρκετά ευανάγνωστο ή εάν μπορεί να γίνει πιο αποτελεσματικό. Λόγω της φύσης των δραστηριοτήτων προγραμματισμού, είναι δύσκολο να πεισθούν ότι η διαδικασία συλλογισμού που πραγματοποιούν κατά την ανάπτυξη ενός αλγορίθμου χρησιμεύει ως λογική βάση για την τελική μορφή του προγράμματος. Ως εκ τούτου η αλγοριθμική σκέψη είναι αναγκαία για τον σχηματισμό δεξιοτήτων που είναι απαραίτητες για την εκτέλεση του αλγορίθμου, ο οποίος συνεπάγεται καλά δομημένα και αποτελεσματικά προγράμματα ηλεκτρονικών υπολογιστών (Teodosiev & Nachev, 2015).

1.3 Η διδασκαλία του προγραμματισμού

Τα εισαγωγικά μαθήματα προγραμματισμού αποσκοπούν να δώσουν στους μαθητές/φοιτητές βασικές γνώσεις στον προγραμματισμό, αλλά παράλληλα δημιουργούν ένα έδαφος για πιο προηγμένες δεξιότητες και γνώσεις. Χρειάζεται να δίνεται έμφαση στην οικοδόμηση δεξιοτήτων για την επίλυση προβλημάτων, και όχι μόνο στην εκμάθηση των χαρακτηριστικών της γλώσσας προγραμματισμού. Η σχεδίαση ενός εισαγωγικού μαθήματος προγραμματισμού δεν πρέπει να στοχεύει στη πλήρη διδασκαλία μιας γλώσσας αλλά στην απόκτηση γνώσεων και δεξιοτήτων σχετικά με μεθόδους αλγοριθμικής και μάθησης καθώς και προσεγγίσεις για λύσεις διαφόρων κατηγοριών προβλημάτων. Η διαδικασία αυτή είναι πολύ περίπλοκη και συχνά οδηγεί τους αρχάριους σε παρανοήσεις και απογοητεύσεις (Govender, 2007; Van Diepen, 2005).

Η εκμάθηση του προγραμματισμού απαιτεί από τους φοιτητές/μαθητές κυρίως νοητική προσπάθεια, συγκέντρωση, προσοχή, λογική και φαντασία. Η βασική απαίτηση, ωστόσο, είναι να έχουν αλγοριθμική σκέψη. Πρέπει, με σαφήνεια και ακρίβεια, να μάθουν να γράφουν και να εφαρμόζουν σωστά τον αλγόριθμο και κατόπιν να εισάγουν το πρόγραμμα στον υπολογιστή. Έτσι, σταδιακά, θα μάθουν να μην κάνουν λανθασμένες επιλογές κατά τη δημιουργία του αλγορίθμου (Teodosiev & Nachev, 2015).

Ο αλγόριθμος αποτελεί κεντρικό στοιχείο της διδασκαλίας του μαθήματος και στη συνέχεια, η διδασκαλία θα πρέπει να προχωρήσει σε ειδικά θέματα της γλώσσας προγραμματισμού. Η αλγοριθμοποίηση περιλαμβάνει έναν προσεκτικό σχεδιασμό του αλγορίθμου προσδιορίζοντας τις απαιτήσεις απόδοσης και σχεδιάζοντας τις δομές δεδομένων που απαιτούνται για αυτή τη λύση. Βέβαια οι λανθασμένες δεξιότητες και συνήθειες που έχουν αποκτήσει οι φοιτητές/μαθητές είναι δύσκολο να αλλάξουν μεταγενέστερα. Πολύ σημαντική είναι η επιλογή των δραστηριοτήτων που θα διδάξουν οι εκπαιδευτικοί. Σε κάποιες περιπτώσεις, η διατύπωση περιλαμβάνει αρκετές ασάφειες ή παραπλανητικές απαιτήσεις, επειδή ο χρήστης δεν γνωρίζει εκ των προτέρων τι θέλει. Τα προβλήματα εμφανίζονται μεταγενέστερα, όταν πολύχρονη προσπάθεια, και πόροι έχουν δαπανηθεί χωρίς να καταλήγουν στο σωστό αποτέλεσμα (Teodosiev & Nachev, 2015).

Ο συνήθης τρόπος διδασκαλίας στα εισαγωγικά μαθήματα προγραμματισμού έχει ως επιδίωξη να παρέχει ενιαία μάθηση σε όλους τους μαθητές και βασίζεται πλήρως στα εγχειρίδια και στην πρακτική εξοικείωση του αντικειμένου, όπως αναφέρεται στο πρόγραμμα σπουδών του κάθε εκπαιδευτικού ιδρύματος. Παράγοντες όπως, ο υψηλός και μεταβλητός ρυθμός αποτυχίας, η έλλειψη ικανοποιητικών επιδόσεων, ακόμα και η εγκατάλειψη καθιστούν αναγκαίο να προταθούν άλλοι τρόποι που να βελτιώνουν την ποιότητα της διδασκαλίας προγραμματισμού (López & García-Reñalvo, 2016).

Όταν ο προγραμματισμός αποτελεί απλά μια πράξη κωδικοποίησης έτοιμων αλγορίθμων χωρίς να δίνει προσοχή στο νόημά και στη δυνατότητα βελτίωσής τους, δεν μπορεί να είναι ο στόχος ενός

μαθήματος προγραμματισμού είτε γενικά είτε εξειδικευμένα. Ο κύριος στόχος της διδασκαλίας δεν πρέπει να επικεντρώνεται στις συντακτικές και σημασιολογικές πτυχές της γλώσσας προγραμματισμού, αλλά στην παροχή γνώσεων και δεξιοτήτων για τον αλγόριθμο. Ο τελικός σκοπός του μαθήματος πρέπει να είναι η ανάπτυξη δεξιοτήτων για την επίλυση δραστηριοτήτων και προβλημάτων, με τη βοήθεια της γλώσσας προγραμματισμού. Οι ικανότητες σύνθεσης προγραμμάτων σχετίζονται κυρίως με τη γνώση στα μαθηματικά και τη λογική, χωρίς αναγκαστικά να κατακτούν πλήρως τη σύνταξη της γλώσσας προγραμματισμού (Teodosiev & Nachev, 2015).

Οι βασικές έννοιες στον προγραμματισμό, ιδιαίτερα οι δομές (ακολουθίας, επιλογής και επανάληψης), οι πίνακες, οι δείκτες, οι συναρτήσεις κ.λπ. μπορούν να συζητηθούν παράλληλα με τη μελέτη της τυπικής σύνταξης στη γλώσσα προγραμματισμού. Η εκμάθηση μιας γλώσσας προγραμματισμού χρειάζεται να βρίσκεται στο πλαίσιο του προβλήματος που επιλύεται, επομένως πρέπει να διδάσκονται νέα αντικείμενα μόλις αυτό απαιτείται από τη συγκεκριμένη δραστηριότητα. Τελικά, ο στόχος της διδασκαλίας είναι οι φοιτητές/μαθητές να μάθουν τις βασικές αρχές προγραμματισμού και τις έννοιες, ανεξάρτητα από τη γλώσσα προγραμματισμού που διδάσκονται (Teodosiev & Nachev, 2015).

Κατά τη διδασκαλία, όταν ο εκπαιδευτικός εισάγει μια νέα έννοια, χρησιμοποιεί συνήθως κάποια απλά παραδείγματα και κατόπιν περιμένει οι μαθητές να μπορούν να λύσουν προβλήματα που είναι είτε καινοφανή είτε παρόμοια με εκείνα που έχει επιδείξει. Έτσι παρουσιάζονται προβλήματα τα οποία οφείλονται και στον εκπαιδευτικό αλλά και στον εκπαιδευόμενο. Όσον αφορά τον εκπαιδευτικό, επειδή πρέπει να βρει κατάλληλες τεχνικές και παραδείγματα για να αφομοιώσουν αφηρημένες έννοιες οι μαθητές, καθώς αυτοί δεν έχουν πραγματικές αναπαραστάσεις για να ενσωματώσουν αφηρημένες έννοιες, ιδιαίτερα στην περίπτωση της εκμάθησης του προγραμματισμού που αποτελεί μια διαδικασία πολλαπλών αφηρημένων διαδικασιών. (Bey, Bensebaa, & Benselem, 2010).

Οι γλώσσες προγραμματισμού αναπτύσσονται γρήγορα, αποκτούν νέα χαρακτηριστικά, όμως δεν αλλάζει η φύση του προγραμματισμού, οι βασικές αρχές και οι κανόνες του. Σύμφωνα με έρευνες στον τομέα αυτό, όσοι μαθαίνουν προγραμματισμό από μικρή ηλικία και ακολουθούν τεχνικές επίλυσης προβλημάτων μπορούν να μάθουν νέες γλώσσες ευκολότερα. Ο προγραμματισμός είναι δύσκολος, απαιτεί ένα μεγάλο βάθος γνώσης ώστε να κατανοηθούν οι διάφορες πτυχές του (σύνταξη, σημασιολογία, πραγματισμός, επίλυση προβλημάτων κλπ.) (Taheri, Sasaki, & Ngetha, 2015).

Όταν μαθητές με διαφορετικές μαθησιακές ανάγκες εισάγονται στον προγραμματισμό και στην υλοποίηση δραστηριοτήτων του, είναι δύσκολο να ικανοποιηθούν όλες οι ανάγκες τους με τις δραστηριότητες που σχεδίασε ο εκπαιδευτικός. Οι μαθητές μικρής ηλικίας αλληλεπιδρούν με

οπτικά εργαλεία προγραμματισμού για να μάθουν να προγραμματίζουν. Όταν η διαδικασία αυτή γίνεται παιγνιωδώς με ψηφιακά αντικείμενα που παρουσιάζουν φυσικά και αισθητικά χαρακτηριστικά (ρομποτική), ενδεχομένως οι μαθητές αυτοί να συμμετάσχουν ξανά σε τέτοιες δραστηριότητες επειδή νιώθουν ικανοποίηση. Κάθε έννοια προγραμματισμού εισάγεται όποτε κρίνεται απαραίτητη για την ανάπτυξη της ικανότητας δημιουργίας προγραμμάτων. Όσα προβλήματα αντιμετώπισαν οι μαθητές, δεν εμπόδισαν την εισαγωγή των απαραίτητων εννοιών. Ως εκ τούτου ο σχεδιασμός της διδασκαλίας χρειάζεται να προσαρμόζεται στις διαφορετικές ανάγκες των μαθητών, όπως η ηλικία και οι προηγούμενες γνώσεις (Paranlasoroulou et al., 2015).. Ο τομέας αυτός απαιτεί επιπλέον έρευνα, είτε στο επίπεδο της θεωρητικής προσέγγισης, είτε στο επίπεδο της ανάπτυξης νέων εργαλείων, είτε στο επίπεδο της εφαρμογής στην τάξη (Κόμης, 2016). Όταν ένας μαθητής χρειάζεται μια εξατομικευμένη υποστήριξη, ο εκπαιδευτικός δεν είναι πάντα σε θέση να την προσφέρει και οι παραδοσιακές μέθοδοι διδασκαλίας δεν δίνουν τέτοια υποστήριξη. Οι διδακτικές προσεγγίσεις που χρησιμοποιούν διαδικτυακές τεχνολογίες επιτρέπουν στους μαθητές πιο ανοιχτή και συνεργατική μάθηση, με περισσότερο επικοινωνιακή προσέγγιση. Κατά την εκμάθηση του προγραμματισμού, η χρήση τεχνολογιών ιστού είναι ζωτικής σημασίας για την υποστήριξη παιδαγωγικών προσεγγίσεων που βελτιώνουν τη μάθηση (Barbosa, Lira, & De, 2013). Μεγάλη πρόκληση αποτελεί για τους εκπαιδευτικούς η επιλογή της προσέγγισης που πρέπει να ακολουθήσουν κατά τη διδασκαλία του οπτικού προγραμματισμού. Αν πρέπει πρώτα να διδάξουν τα αντικείμενα ή τις δομές, καθώς και να επιλέξουν την πρώτη γλώσσα προγραμματισμού. Οι Ivanonić, Xinogalos, Pitner, & Savić, (2015) θεωρούν καλύτερη προσέγγιση πρώτα να διδάσκονται οι βασικές έννοιες του δομημένου προγραμματισμού και αργότερα οι αντικειμενοστρεφείς έννοιες. Θεωρούν ότι μια γλώσσα σχεδιασμένη για εκπαιδευτικούς σκοπούς δίνει στους εκπαιδευτικούς καλύτερη ισορροπία στο διαχωρισμό των εκπαιδευτικών πτυχών του προγραμματισμού. Επομένως, τα μαθήματα προγραμματισμού χρειάζεται να εστιάζουν σε θεμελιώδεις εργασίες ανάπτυξης λογισμικού και σε έννοιες προγραμματισμού και όχι στη δημιουργία προγραμμάτων σε μια συγκεκριμένη γλώσσα προγραμματισμού. Το πρόγραμμα σπουδών προγραμματισμού σε κάθε βαθμίδα εκπαίδευσης μπορεί να ποικίλει με βάση πολλούς παράγοντες όπως η φύση του εκπαιδευτικού ιδρύματος και ο σκοπός του, τα προγενέστερα σχετικά μαθήματα, τα επιδιωκόμενα αποτελέσματα κ.α. Παρόλο που τα εκπαιδευτικά ιδρύματα φαίνεται να έχουν τους ίδιους στόχους όσον αφορά την πορεία των μαθημάτων, υπάρχουν διαφορές στο επίκεντρο κάθε μαθήματος. Εντούτοις, ανεξάρτητα από τις διαφορές στην ύλη, υπάρχουν κάποιες ουσιαστικές έννοιες που υπάρχουν σε κάθε μάθημα προγραμματισμού (Ivanonić, Xinogalos, Pitner, & Savić, 2015).

1.4 Οι δυσκολίες στον προγραμματισμό

Κατά τη διδασκαλία και την εκμάθηση του προγραμματισμού εκπαιδευτικοί και μαθητές αντιμετωπίζουν μεγάλες δυσκολίες. Υπάρχει μεγάλη βιβλιογραφία σε σχέση με τις δυσκολίες που αντιμετωπίζουν όσοι εκπαιδεύονται στον προγραμματισμό είτε είναι αρχάριοι είτε όχι. Κυρίως εντοπίστηκε μεγάλος όγκος ερευνών που αφορά τις δυσκολίες των αρχαρίων προγραμματιστών. Η εστίαση δίνεται στις αφηρημένες έννοιες, τη σύνταξη και τη δομή μιας γλώσσας προγραμματισμού, την επίλυση προβλημάτων, τη δημιουργία αλγόριθμου, τα συντακτικά και λογικά σφάλματα και των συνδυασμό εννοιών. Άλλοι λόγοι μπορεί να είναι η ανεπαρκής γνώση τεχνικών επίλυσης προβλημάτων, η έλλειψη ενδιαφέροντος και το κίνητρο για μάθηση, τα μη αποτελεσματικά συστήματα υποστήριξης διδασκαλίας και μάθησης, η έμφαση στην εκμάθηση με επανάληψη και στο στατικό περιεχόμενο (Kiran & Moudgalya, 2015).

Όπως αναφέρθηκε ο προγραμματισμός δεν είναι εύκολο αντικείμενο για μελέτη επειδή απαιτεί κατανόηση αφηρημένων εννοιών. Οι δομές επανάληψης, οι πίνακες, οι εντολές εκχωρήσεις, και οι αλφαριθμητικοί τύποι δεδομένων είναι αφηρημένες έννοιες και, ως εκ τούτου, θεωρούνται πολύπλοκες για να κατανοηθούν. Πολλοί μαθητές αντιμετωπίζουν δυσκολίες στη μάθηση λόγω της φύσης του προγραμματισμού (Barbosa et al., 2013; Kiran & Moudgalya, 2015; Lahtinen, Ala-Mutka, & Järvinen, 2005). Οι αρχάριοι προγραμματιστές αντιδρούν με διαφορετικούς τρόπους όταν τους δοθούν εργασίες που απαιτούν αφηρημένη συλλογιστική. Μερικοί ενεργοποιούνται και προσηλώνονται στην εργασία τους, ενώ κάποιοι άλλοι παραιτούνται εντελώς επειδή αδυνατούν να τα καταφέρουν (Hansen & Eddy, 2007). Τα προγράμματα και οι αλγόριθμοι είναι αφηρημένες και σύνθετες δομές, καθώς περιλαμβάνουν έννοιες και διαδικασίες που συχνά υπάρχει δυσκολία στη διδασκαλία και στην εκμάθηση τους (Eckerdal, 2009).

Η κατανόηση των εννοιών προγραμματισμού συνδέει δύο εννοιολογικές ομάδες: την αλγοριθμική και τη δομική, οι οποίες αντιστοιχούν σε δεξιότητες επίλυσης προβλημάτων και σε δεξιότητες προγραμματισμού, δηλαδή σε δύο διαφορετικές ομάδες γνωστικών δεξιοτήτων. Η πρώτη, η αλγοριθμική, αφορά κυρίως τις δομές ελέγχου, επανάληψης και επιλογής (Cabo, 2015). Διαπιστώθηκε ότι πολλοί μαθητές δυσκολεύονται να γράψουν τη συνθήκη στη δομή επιλογής ή επανάληψης (Lahtinen et al., 2005), επειδή δεν χρησιμοποιούν σωστά τους λογικούς και τους συγκριτικούς τελεστές (Makihara et al., 2016).

Η δεύτερη ομάδα συσχετίζεται με τους πίνακες και τις εντολές εκχωρήσεις, και θα μπορούσε να χαρακτηριστεί ως δομική ομάδα. Για να βελτιωθεί η επίδοση χρειάζεται να δοθεί ιδιαίτερη έμφαση στις δομικές συνιστώσες του προγραμματισμού για την επιτυχή μετάβαση από τις έννοιες στις δεξιότητες (Cabo, 2015). Η γνώση και η κατανόηση σε ανεξάρτητες έννοιες προγραμματισμού δείχνουν κρυφές συσχετίσεις και αλληλεξαρτήσεις και φανερώνουν ότι ορισμένες έννοιες είναι πιο σημαντικές από άλλες για τη συγγραφή προγραμμάτων. Επιπλέον, διαπιστώθηκε ότι οι μαθητές

είχαν καλύτερη κατανόηση των εννοιών που σχετίζονται με τον «αλγοριθμικό» παράγοντα παρά με τις έννοιες που σχετίζονται με τον «δομικό» παράγοντα.

Οι αρχάριοι περιορίζονται συνήθως στην επιφανειακή γνώση των προγραμμάτων. Συχνά ακολουθούν τον προγραμματισμό "γραμμή με γραμμή" αντί να χρησιμοποιούν σημαντικές δομές προγραμμάτων. Η γνώση των αρχαρίων τείνει να είναι συγκεκριμένη για το περιβάλλον και συχνά αποτυγχάνουν να εφαρμόσουν τις γνώσεις που έχουν αποκτήσει επαρκώς. Μπορεί να γνωρίζουν τη σύνταξη και τη σημασιολογία μεμονωμένων εντολών ή δομών, αλλά δεν ξέρουν πώς να τις συνδυάσουν. Ως εκ τούτου, είναι σημαντικό να συνδυαστούν τόσο η γνώση των εννοιών όσο και οι στρατηγικές για τη χρήση τους στη μαθησιακή διαδικασία. Ως πιο δύσκολες προγραμματικές έννοιες θεωρούνται η επανάληψη, οι πίνακες και οι δηλώσεις μεταβλητών, οι μη αριθμητικοί τύποι δεδομένων (χαρακτήρες, λογικοί). Υπάρχουν συχνά παρερμηνείες που σχετίζονται με τη αρχικοποίηση μιας μεταβλητής όπως το πλήθος ή το άθροισμα ή το μέγιστο, τις δομές επανάληψης, τις συνθήκες κ.α. Επίσης εντοπίζονται προβλήματα στην κατανόηση ότι κάθε εντολή εκτελείται στην κατάσταση που έχει δημιουργηθεί από τις προηγούμενες εντολές (Lahtinen et al., 2005).

Η δυσκολία που αντιμετωπίζουν οι αρχάριοι κατά τη σύνταξη των εντολών, αποσπά την προσοχή τους από την επίλυση του προβλήματος και την οργάνωση του προγράμματος. Αυτή η δυσκολία αναγνωρίστηκε νωρίς στην εκμάθηση του προγραμματισμού και αναπτύχθηκαν διαφορετικές στρατηγικές. Μια προσέγγιση είναι η μετάβαση από τη διδασκαλία του προγραμματισμού στις δεξιότητες επίλυσης προβλημάτων. Αυτή η προσέγγιση είναι επιτυχής και έτσι αποφεύγονται κάποια από τα προβλήματα που εμποδίζουν την πρόοδο στην ανάπτυξη των δεξιοτήτων σκέψης οι οποίες είναι σημαντικές στον προγραμματισμό. Ωστόσο, έχει επίσης επικριθεί επειδή η μετατροπή μιας λύσης ενός προβλήματος σε ένα πρόγραμμα δεν είναι προφανής. Ο στόχος του μαθήματος επίλυσης προβλημάτων είναι να διδάξει τις έννοιες προγραμματισμού χωρίς το βάρος της εκμάθησης πλήρως μιας γλώσσας προγραμματισμού (Cabo, 2015).

Οι γλώσσες προγραμματισμού εξελίσσονται συνεχώς και αποκτούν νέα τεχνολογικά χαρακτηριστικά. Η χρήση των βιβλιοθηκών της γλώσσας επίσης απαιτεί την ανεξάρτητη αναζήτηση των πληροφοριών, γεγονός που μπορεί να δυσκολεύει τους αρχάριους. Οι αντιλήψεις των εκπαιδευτικών για το δύσκολο περιεχόμενο του μαθήματος είναι σχεδόν ίδιες με τις αντιλήψεις των μαθητών. Επιπλέον, οι εκπαιδευτικοί αντιλαμβάνονται ότι η κατανόηση των δομών προγραμματισμού είναι δύσκολη καθώς και η διαχείριση σφαλμάτων απαιτεί πλήρη κατανόηση του προγράμματος (Lahtinen et al., 2005).

Οι αρχάριοι κάνουν πολλά λάθη και γράφουν συχνά έναν δυσανάγνωστο πρόγραμμα, το οποίο είναι δύσκολο να συντηρηθεί. Είναι πολύ πιο εύκολο να διδαχθεί ένα καλό στυλ προγραμματισμού

σε άτομα που δεν έχουν προηγούμενη εμπειρία προγραμματισμού, ενώ είναι δύσκολο να μπορέσουν οι μαθητές να υιοθετήσουν καλό στυλ προγραμματισμού, αφού είχαν υιοθετήσει ένα κακό προγενέστερα. Τα προβλήματα των κακών δεξιοτήτων προγραμματισμού εμφανίζονται σε μεγαλύτερα προγράμματα ή σε καταστάσεις όπου το πρόγραμμα χρειάζεται συντήρηση από άλλα άτομα ή μεταφέρθηκε σε άλλο περιβάλλον (Teodosiev & Nachev, 2015).

Τα κύρια λάθη που κάνουν οι φοιτητές/μαθητές κατά τη διάρκεια της μαθησιακής διαδικασίας είναι λάθη που σχετίζονται με τη σύνταξη, τη σημασιολογία, επίσης εντοπίζονται δυσκολίες στην κατανόηση των προβλημάτων, στο σχεδιασμό αλγορίθμων και στην ανίχνευση σφαλμάτων του προγράμματος. Οι δυσκολίες στη εκμάθηση προγραμματισμού σχετίζονται με την ανάπτυξη διαφόρων δεξιοτήτων (επίλυση προβλημάτων, υπολογιστικών εργαλείων για τη δημιουργία και αξιολόγηση της ποιότητας προγραμμάτων). Αυτοί οι παράγοντες οδηγούν τον εκπαιδευόμενο που έχει ελάχιστες ή μηδενικές προηγούμενες γνώσεις σχετικά με το αντικείμενο να αισθάνεται ανίκανος να συνεχίσει στα μαθήματα προγραμματισμού (Barbosa et al., 2013).

Τα σφάλματα αντικατοπτρίζουν διάφορες λεπτομέρειες κατανόησης εννοιών και δεξιοτήτων προγραμματισμού των μαθητών. Τα σφάλματα, τα οποία συνήθως είναι συντακτικά και σημασιολογικά αποτελούν τα βασικά θεμέλια για τη γνώση της δεξιότητας προγραμματισμού (Barbosa et al., 2013; Kiran & Moudgalaya, 2015). Μόνο μετά την επίλυση των σφαλμάτων και την επιτυχή μεταγλώττιση μπορεί ο μαθητής να βελτιωθεί τις δεξιότητες του στον προγραμματισμό. Αυτή το στάδιο της μάθησης είναι πολύ σημαντικό και οι άπειροι απογοητεύονται. Έτσι συχνά εμφανίζουν σημαντικές δυσκολίες στην κατανόηση μηνυμάτων σφάλματος. Αντίθετα οι πιο έμπειροι τείνουν να εφαρμόζουν συγκεκριμένες στρατηγικές για την υπέρβαση αυτών των σφαλμάτων (Kiran & Moudgalaya, 2015).

Ένα από τα πιο δύσκολα ζητήματα στην κατανόηση του προγραμματισμού αφορά, τη σχεδίαση ενός προγράμματος για την επίλυση ενός συγκεκριμένου προβλήματος, καθώς και τη διαίρεση του προγράμματος σε υποπρογράμματα. Για όλα αυτά τα θέματα ο μαθητής πρέπει να κατανοήσει τις μεγαλύτερες οντότητες του προγράμματος αντί για κάποιες λεπτομέρειες σχετικά με αυτό (Lahtinen et al., 2005). Η επίλυση προβλημάτων μέσω της αποτελεσματικής διάσπασης ενός προγράμματος και η σύνθεση είναι εξαιρετικά δύσκολες για αρχάριους. Αυτό επιδεινώνεται από το γεγονός ότι αυτές οι στρατηγικές είναι συνήθως σιωπηρές στη διδασκαλία: οι μαθητές αφήνονται να καταλάβουν τις δικές τους στρατηγικές επίλυσης προβλημάτων (Castro, 2015). Για να είναι σε θέση να γράψουν προγράμματα, οι μαθητές χρειάζονται δεξιότητες επίλυσης προβλημάτων και προγραμματισμού (Cabo, 2015).

Στην επίλυση προβλημάτων χρησιμοποιείται κάθε βήμα για την επίτευξη του επιθυμητού αποτελέσματος. Ο προγραμματισμός έχει πολλά κοινά με τις φυσικές επιστήμες και τα μαθηματικά,

επομένως είναι ένας πρακτικός και «οπτικός» τρόπος μάθησης για την επίλυση προβλημάτων. Οι τεχνικές διδασκαλίας επίλυσης προβλημάτων κατά τη σχολική ηλικία μπορούν να προετοιμάσουν τους μαθητές να μάθουν τις έννοιες του προγραμματισμού. Επίσης η διδασκαλία των γλωσσών προγραμματισμού σε παιδιά αποτελεί δύσκολο έργο αλλά σημαντικό. Ο προγραμματισμός ως υποχρεωτικό αντικείμενο ή προαιρετική επιλογή σε προγενέστερο στάδιο στο σχολικό σύστημα θεωρείται επωφελής (Taheri et al., 2015).

Η μη ακριβής κατανόηση του τρόπου με τον οποίο ο υπολογιστής εκτελεί ένα πρόγραμμα οδηγεί τους αρχάριους σε μεγάλες δυσκολίες στην εκμάθηση του προγραμματισμού (Ma, Ferguson, Roper, & Wood, 2011). Η διάσπαση ενός προβλήματος θεωρείται ως η πιο δύσκολη δεξιότητα προγραμματισμού για να κατακτηθεί. Αυτό οφείλεται: στην έλλειψη εμπειρίας, στην ελλιπή κατανόηση του προβλήματος και στη σειρά διδασκαλίας των εννοιών του προγραμματισμού. Οι εκπαιδευτικοί θεωρούν ότι οι εκπαιδευόμενοι παρότι κατανοούν την έννοια της διάσπασης ενός προβλήματος, δυσκολεύονται στην εφαρμογή της διαδικασίας αυτής. Είναι σε θέση να χρησιμοποιούν την δεξιότητα της διάσπασης πιο επιτυχημένα σε προβλήματα παρόμοια με εκείνα που έχουν λύσει ή όταν κατανοούν πολύ καλά το πρόβλημα. Συνήθως οποιαδήποτε δεξιότητα εισάγεται πρώτη, αποτελεί σημείο δυσκολίας για τους μαθητές. Ωστόσο, η κατανόηση της διάσπασης του προβλήματος, αποτελεί προϋπόθεση για την αφαίρεση, το σχεδιασμό αλγορίθμων και την αξιολόγηση. Ως εκ τούτου, πρέπει να κατακτηθεί, σε κάποιο βαθμό, πριν να καταστεί δυνατή η πρόσβαση στην πολυπλοκότητα των επιπέδων που ακολουθούν (Selby, 2015).

Ορισμένες από τις δυσκολίες που αντιμετωπίζουν οι αρχάριοι προγραμματιστές σχετίζονται με το γεγονός ότι η κατανόηση του προβλήματος που πρέπει να επιλυθεί με την εφαρμογή ενός προγράμματος θα πρέπει να αποτελεί προϋπόθεση για τη σύνταξη του ίδιου του προγράμματος. Η αδυναμία των εκπαιδευόμενων να δημιουργήσουν ένα νοητικό μοντέλο ενός δοσμένου προβλήματος εμποδίζει την ικανότητά τους να αναπτύξουν δεξιότητες επίλυσης προβλημάτων και να γράφουν προγράμματα υπολογιστών (Cabo, 2015).

Άλλο σημαντικό πρόβλημα στη διδασκαλία του προγραμματισμού φαίνεται να είναι το γεγονός ότι οι μαθητές υπερεκτιμούν την κατανόησή τους. Οι εκπαιδευτικοί πιστεύουν ότι το περιεχόμενο του μαθήματος είναι πιο δύσκολο για τους μαθητές από ότι πιστεύουν οι ίδιοι οι μαθητές. Η αιτία για τις διαφορετικές αντιλήψεις μπορεί να οφείλεται στο γεγονός ότι οι μαθητές δεν συνειδητοποιούν όλες τις δυσκολίες που έχουν όταν μαθαίνουν προγραμματισμό, αλλά οι εκπαιδευτικοί το κάνουν κατά την αξιολόγηση του προγραμματισμού. Επίσης, οι εκπαιδευτικοί γνωρίζουν τις έννοιες του προγραμματισμού σε βάθος και είναι σε θέση να παρατηρούν ότι οι μαθητές δεν έχουν πλήρη κατανόηση σε ζητήματα που οι ίδιοι πιστεύουν ότι έχουν (Lahtinen et al., 2005).

Ωστόσο, το μεγαλύτερο πρόβλημα των αρχάριων προγραμματιστών δεν φαίνεται να είναι η κατανόηση των βασικών εννοιών, αλλά μάλλον η εφαρμογή τους. Οι εκπαιδευτικοί πρέπει να επικεντρωθούν περισσότερο στον συνδυασμό και τη χρήση αυτών των χαρακτηριστικών, ειδικά στα βασικά ζητήματα σχεδιασμού των προγραμμάτων (Lahtinen et al., 2005). Επιπλέον, δεν υπάρχουν αρκετοί πόροι ώστε να εφαρμόζεται εξατομικευμένη διδασκαλία. Επίσης, τα τμήματα των μαθητών είναι μεγάλα και ανομοιογενή και επομένως είναι δύσκολο να σχεδιαστεί η διδασκαλία έτσι ώστε να είναι εποικοδομητική για όλους (Lahtinen et al., 2005).

Παρανοήσεις συμβαίνουν όταν οι μαθητές με ανεπαρκή κατανόηση μιας έννοιας, κατασκευάζουν τους δικούς τους κανόνες. Παρανοήσεις μπορεί επίσης να προκύψουν όταν γνωστοί όροι χρησιμοποιούνται σε ένα άγνωστο περιβάλλον (Sirkiä & Sorva, 2012).

Άλλες πτυχές όπως, το κίνητρο και η ενασχόληση είναι επίσης σημαντικοί καθοριστικοί παράγοντες της επιτυχίας του εκπαιδευόμενου στην επίλυση προβλημάτων. Οι παιδαγωγικές προσεγγίσεις που θα κινητοποιήσουν και θα προσελκύσουν τους μαθητές θα διευκολύνουν επίσης τη μετάβασή τους από τις έννοιες στις πρακτικές δεξιότητες, με ταυτόχρονη επίδραση στην επιτυχία των μαθητών (Cabo, 2015). Οι εκπαιδευτικοί χρειάζεται να ενθαρρύνουν τους μαθητές να δοκιμάζουν μια άλλη άποψη στα προβλήματα. Όταν δημιουργούν ένα πρόγραμμα μετατρέπουν το κείμενο σε αυτοματοποιημένο πρόγραμμα. Με αυτόν τον τρόπο, επιλύουν ένα πρόβλημα που είναι ουσιαστικά "τι πρέπει να κάνω για να καταστήσω κάτι αποδοτικό, παραγωγικό και χρήσιμο" (Taheri et al., 2015).

Πολύ σημαντικό στην εκπαίδευση του προγραμματισμού είναι ο συνδυασμός του περιεχομένου της γνώσης, καθώς τόσο η δηλωτική (στατική) γνώση όσο και η δυναμική κατανόηση πρέπει να απεικονίζονται και να εκπροσωπούνται στους μαθητές. Συνεπώς, υπάρχουν δύο εννοιολογικά και λειτουργικά εμπόδια που πρέπει να ξεπεραστούν στη διδασκαλία και την εκμάθηση του προγραμματισμού. Οι εκπαιδευτικοί αντιμετωπίζουν προβλήματα με τη μεταφορά θεωρητικών εννοιών καθώς και με την εξήγηση πιο πρακτικών τεχνικών προγραμματισμού. Η οπτικοποίηση του λογισμικού με τον συνδυασμό γραφικών, ακουστικών και κειμενικών αναπαραστάσεων στοχεύει στη διευκόλυνση της εκμάθησης και της κατανόησης του προγραμματισμού (Olsson & Mozelius, 2015).

Με την ανάπτυξη των μεθόδων διδασκαλίας και του διδακτικού υλικού, καθώς και με τη διδασκαλία των βασικών στοιχείων προγραμματισμού και της επίλυσης προβλημάτων από μικρή ηλικία, θα ήταν δυνατό να ενθαρρυνθεί ο μελλοντικός προγραμματιστής να γίνει πιο δημιουργικός και να γίνει πιο ενδιαφέρον το αντικείμενο του προγραμματισμού για αυτόν. Η αλληλεπίδραση με τους συμμαθητές, και τους εκπαιδευτικούς θα μπορούσε επίσης να βοηθήσει τους μαθητές να καλύψουν τις ανάγκες τους και θα ήταν ένας αποτελεσματικός τρόπος για να μάθουν επίσης. Αυτά

τα είδη αλληλεπιδράσεων θα βοηθούσαν στην καλύτερη κατανόηση των εννοιών και στην αποκατάσταση των σφαλμάτων στο πρόγραμμα (Taheri et al., 2015).

1.5 Η αξιολόγηση και οι δυσκολίες της

Ένα μεγάλο σύνολο ερευνών επιβεβαιώνουν ότι η αξιολόγηση των γνώσεων προγραμματισμού δεν είναι εύκολη ούτε ακριβής. Το χαμηλό ποσοστό επιτυχίας των μαθητών στα μαθήματα προγραμματισμού, μπορεί να εξηγηθεί εν μέρει και από την έλλειψη καλών εργαλείων αξιολόγησης για την ταξινόμηση της δυσκολίας των μαθησιακών δραστηριοτήτων που αφορούν τη σύνταξη ενός προγράμματος.

Παρότι χρησιμοποιούνται διαφορετικά μέσα αξιολόγησης εντοπίζονται ορισμένες κοινές πεποιθήσεις σχετικά με την αξιολόγηση. Αυτές σχετίζονται κυρίως με το γεγονός ότι οι γνώσεις των μαθητών δεν μπορούν και δεν πρέπει να αξιολογούνται μόνο με τελική εξέταση. Οι μαθητές πρέπει να αξιολογούνται κατά κάποιο τρόπο, κατά τη διάρκεια διδασκαλίας του μαθήματος ώστε να κινητοποιούνται αλλά και να παρακολουθούν την πρόοδό και τις δυσκολίες τους (Ivanonίc et al., 2015):

Οι εκπαιδευτικοί παρότι καθημερινά ενδιαφέρονται για την επίδοση των μαθητών, δίνουν όμως ιδιαίτερη βαρύτητα στις βαθμολογίες των εξετάσεων, έτσι η επίπτωση άλλων παραγόντων στην αξιολόγηση στην πραγματικότητα είναι αμελητέα. Για τους εκπαιδευτικούς, η διδασκαλία γίνεται εκπαίδευση βασισμένη σε εξετάσεις. Οι μαθητές που έχουν ισχυρές ερευνητικές δυνατότητες ενδέχεται να αγνοήσουν την ανάπτυξη της ικανότητάς τους για καινοτομία λόγω της εστίασης μόνο στις βαθμολογίες, ενώ άλλοι που έχουν ισχυρή πρακτική ικανότητα και ικανότητα για καινοτομία απογοητεύονται συχνά επειδή στο θεωρητικό τεστ είναι δύσκολο να πάρουν υψηλή βαθμολογία και να κερδίσουν την αναγνώριση. Για τους παραπάνω λόγους οι ερευνητές προτείνουν την ποικιλότητα στην αξιολόγηση. Αυτό το μοντέλο αξιολόγησης δίνει μια δίκαιη και λογική εκτίμηση όσον αφορά τη συνολική ποιότητα, αλλά δεν είναι ακριβές επειδή τα κριτήρια αξιολόγησης είναι αυθαίρετα. Ως εκ τούτου, το κλειδί ενός συστήματος αξιολόγησης είναι ότι ο καθορισμός μετρήσιμων δεικτών για κάθε μέρος της διαδικασίας αξιολόγησης, ιδίως για αστάθμητες παραμέτρους όπως τα αποτελέσματα εξετάσεων και τα τεστ στον υπολογιστή (Ling et al., 2012).

Τα μαθήματα προγραμματισμού αναμένουν από τους μαθητές/φοιτητές να λύνουν προβλήματα προγραμματισμού σε τακτική βάση. Λόγω του μεγάλου αριθμού των μαθητών στα μαθήματα καθίσταται εξαιρετικά δύσκολο έργο για τους εκπαιδευτικούς. Υπάρχει η ανάγκη να αυτοματοποιηθεί η διαδικασία της αξιολόγησης του προγραμματισμού. Οι Kaushal & Singh (2012) προτείνουν ένα σύστημα αξιολόγησης το οποίο αξιολογεί ένα πρόγραμμα από δύο κυρίως απόψεις: το κατά πόσο το πρόγραμμα είναι ένα αντίγραφο ενός ομότιμου (ανίχνευση λογοκλοπής) και αν είναι πρωτότυπο τότε επιβεβαιώνεται η ορθότητα του μέσω μιας σειράς δοκιμαστικών περιπτώσεων που παράγονται είτε χειρονακτικά είτε αυτόματα. Σε εμπειρική μελέτη η πρόοδος των μαθητών μετρήθηκε με τέσσερις παραμέτρους: την κανονικότητα, την ακεραιότητα, την αποτελεσματικότητα και την ακρίβεια. Επίσης οι μαθητές ενημερώνονται για την κατάσταση των

υποβληθέντων προγραμμάτων. Η ενσωμάτωση ενός τέτοιου συστήματος βελτίωσε την πρόοδο των μαθητών και ξεκούρασε τον εκπαιδευτικό από το κοπιαστικό έργο της χειρονακτικής αξιολόγησης των προγραμμάτων.

Οι González-Sacristán, Molins-Ruano, Díez, Rodríguez, & Sacha (2013) προτείνουν μια μέθοδο υποβοηθούμενη από υπολογιστή για τη βελτίωση της ποιότητας της διαδικασίας αξιολόγησης σε θέματα που σχετίζονται με τις βασικές έννοιες προγραμματισμού. Η μέθοδος περιλαμβάνει δύο μοντέλα. Στο πρώτο, όλοι οι μαθητές πρέπει να απαντήσουν στον ίδιο αριθμό ερωτήσεων διαφορετικού επιπέδου δυσκολίας. Ενώ στο δεύτερο, ο αριθμός των ερωτήσεων που απαντούν οι μαθητές δεν είναι πάντα ο ίδιος, είναι μικρότερος για τις πιο εύκολες και τις πιο δύσκολες ερωτήσεις. Έτσι είναι ότι είναι ευκολότερο να εντοπιστούν και να αξιολογηθούν οι φοιτητές που είναι πολύ καλοί (ή πολύ κακοί) από τους φοιτητές που βρίσκονται στο ενδιάμεσο επίπεδο. Μέσω προσομοιώσεων και πειραμάτων, τα μοντέλα που προτείνονται μπορούν να είναι πολύ χρήσιμα σε σύγκριση με τις παραδοσιακές διαδικασίες αξιολόγησης. Τα χαρακτηριστικά της μεθόδου βελτιώνουν την αντικειμενικότητα, την ασφάλεια και λαμβάνουν υπόψη τη συνάφεια των περιεχομένων του θέματος.

Η αξιολόγηση των δυνατοτήτων των μαθητών στον προγραμματισμό με χρήση του τεστ Dehnadi (2006) σε μια εκπαιδευτική γλώσσα, με online αξιολόγηση, περιγράφεται από τους Milkoná & Petránek (2016). Οι μαθητές εισάγουν τη λύση τους, και το πρόγραμμα τους δείχνει βήμα-βήμα πως ο αλγόριθμος τους λειτουργεί και αν είναι σωστός ή όχι. Το βασικό αποτέλεσμα από την εμπειρική μελέτη είναι ότι το τεστ αυτό δεν μπορεί να χρησιμοποιηθεί για όλους τους μαθητές, δίνει όμως καλά αποτελέσματα σε μαθητές με προηγούμενη εμπειρία προγραμματισμού.

Οι ικανότητες αυτοαξιολόγησης επίσης, μέσω διαδικτυακού εργαλείου, επιτρέπουν στον φοιτητή να έχει εικόνα για τη δική του διαδικασία μάθησης. Η αξιολόγηση περιλαμβάνει τόσο τον εντοπισμό λαθών, τα σημεία βελτίωσης όσο και γενική σύγκριση με τους υπόλοιπους συμμαθητές, από την άποψη της ποιότητας του λογισμικού. Με αυτόν τον τρόπο παρέχεται στον φοιτητή ένα πλαίσιο στο οποίο συγκρίνεται η ορθότητα του προγράμματος που έχει δημιουργήσει προτού παραδοθεί στον εκπαιδευτικό για αξιολόγηση. Ο εκπαιδευτικός είναι σε θέση να έχει πρόσβαση σε όλες αυτές τις πληροφορίες, έχοντας μια εικόνα για το πώς η τάξη, καθώς και μεμονωμένοι φοιτητές, προχωράνε. Αναμένεται ότι ο περιορισμός της επανάληψης της επίλυσης του προβλήματος από τον μαθητή και η ανατροφοδότηση των εκπαιδευτικών θα ενισχύσουν την αυτογνωσία των φοιτητών και θα βελτιώσουν τη συνολική τους επίδοση στα μαθήματα προγραμματισμού (Herrero & Baroque, 2015).

Η ικανότητα σε μια γλώσσα προγραμματισμού μπορεί να συγκριθεί με την επάρκεια μιας φυσικής γλώσσας (Poss, 2014). Έτσι, όπως αξιολογείται η επάρκεια στην φυσική γλώσσα με ορισμένα

κριτήρια θα μπορούσε να δημιουργηθεί ένα σύνολο μετρήσεων και τεχνικών που να επιτρέπουν τη δημιουργία προφίλ προγραμματιστών με βάση την ευχέρεια και τις ικανότητες σε μια γλώσσα προγραμματισμού. Διερευνώνται τεχνικές που αποσκοπούν στην αξιολόγηση των ικανοτήτων των προγραμματιστών μέσω της στατικής ανάλυσης του πηγαίου κώδικα τους, δηλαδή την ανάλυση του κώδικα χωρίς την πραγματική εκτέλεση του, σε αντίθεση με τη δυναμική ανάλυση που γίνεται κατά την εκτέλεση προγραμμάτων. Συνήθως εκτελείται με στόχο την εύρεση σφαλμάτων ή τη διασφάλιση ότι τηρούνται οι οδηγίες κωδικοποίησης (Ferreira Novais, Varanda Pereira, & Rangel Henriques, 2016).

Για την αξιολόγηση της δυσκολίας δραστηριοτήτων προγραμματισμού, συνήθως χρησιμοποιούνται δύο προσεγγίσεις, των ταξινομιών και των μετρήσεων. Η προσέγγιση των ταξινομιών, η οποία είναι υποκειμενική, απαιτεί οι εκπαιδευτικοί, ως δημιουργοί δραστηριοτήτων προγραμματισμού, να είναι έμπειροι σχετικά με την ταξινόμηση των δραστηριοτήτων στα επίπεδα των ταξινομιών. Οι ταξινομίες που χρησιμοποιούνται είναι η αρχική ταξινόμια Bloom, η αναθεωρημένη ταξινόμια Bloom και η ταξινόμια SOLO. Η άλλη προσέγγιση, οι μετρήσεις, είναι μια ποσοτική προσέγγιση όπου βασίζεται σε αναλυτικά εργαλεία που αναλύουν την απάντηση δείγματος για τις ερωτήσεις προγραμματισμού και επιστρέφουν μια αριθμητική τιμή που βοηθά στην ταξινόμηση της δυσκολίας του ερωτήματος που εξετάζεται. Χρησιμοποιούνται μετρήσεις που προέρχονται από τη λύση που δίνεται από τον εκπαιδευτή και που μπορεί να είναι πιο απλή από τις λύσεις που δίνουν οι μαθητές. Η λύση είναι ένας ενδεικτικός δείκτης δυσκολίας που βοηθά στην ταξινόμηση ερωτήσεων με βάση τις απαιτήσεις τους και πιθανόν να μην είναι κατάλληλη για ειδικά παραδείγματα προγραμματισμού (όπως η αναδρομή). Επίσης, ένα πρόγραμμα που δεν είναι ευανάγνωστο είναι δύσκολο να γραφτεί (Elnaffar, 2016).

Γενικά η αλγοριθμική είναι ένα δύσκολο πεδίο για να διδαχτεί από τον εκπαιδευτικό και πολύπλοκη η αφομοίωση από τον μαθητή. Ο αλγόριθμος, επιπλέον, χαρακτηρίζεται από το πλήθος των λύσεων για ένα πρόβλημα. Αυτό το χαρακτηριστικό αποτελεί την πρώτη δυσκολία, επειδή είναι δύσκολο ο εκπαιδευτικός να προβλέψει όλες τις πιθανές λύσεις ενός προβλήματος προκειμένου να τις ενσωματώσει στη βάση της λύσης του (Bey et al., 2010).

Η δεύτερη δυσκολία είναι άμεσα συνδεδεμένη με την ίδια την αξιολόγηση η οποία κατέχει πρωταρχική θέση σε μεγάλο αριθμό μαθησιακών δραστηριοτήτων. Οι ερευνητές θεωρούν ότι η αποτελεσματική διδασκαλία δεν αποδίδει ουσιαστικό ρόλο στην αξιολόγηση ως μέρος της μαθησιακής διαδικασίας. Η αξιολόγηση στην τάξη αποτελεί πηγή ασάφειας μεταξύ του αξιολογητή και του εκπαιδευόμενου, και μερικές φορές μεταξύ των ίδιων των αξιολογητών. Παρ'όλα αυτά, είναι μια διαδικασία ενσωμάτωσης στη διαδικασία μάθησης. Επιπλέον, η αξιολόγηση δεν περιορίζεται στην απόδοση ενός βαθμού, αν και είναι σημαντικό όταν θέλουμε να

ποσοτικοποιήσουμε τις δεξιότητες. Η αξιολόγηση αποτελεί οδηγό για την πρόοδο του εκπαιδευόμενου και παρεμβαίνει στο επίπεδο αλληλεπίδρασης μεταξύ εκπαιδευτικού και εκπαιδευόμενου για τη βελτιστοποίηση της μεταφοράς γνώσεων, δεξιοτήτων και πρακτικών. Έτσι, η σημασία της είναι πρωτεύουσα. Με αυτή την έννοια, έχουν χρησιμοποιηθεί πολλές μέθοδοι και εργαλεία στην αξιολόγηση, τα οποία είναι ανεπαρκή. Αυτή η ανεπάρκεια χαρακτηρίζεται είτε από αναποτελεσματικότητα, από αμφίβολο αποτέλεσμα είτε από μοναδικότητα, δηλαδή δεν μπορεί να εφαρμοστεί σε όλα τα πεδία (Bey et al., 2010).

Η επίδοση στις έννοιες προγραμματισμού συσχετίζεται σε μεγάλο βαθμό με την απόδοση στις πρακτικές δεξιότητες προγραμματισμού. Η κατανόηση εννοιών είναι ένας καλός παράγοντας πρόβλεψης της ικανότητας εφαρμογής της γνώσης στη σύνταξη προγραμμάτων. Χωρίς καλό υπόβαθρο στην κατανόηση των εννοιών, είναι απίθανο να αναπτυχθούν αποδεκτές πρακτικές δεξιότητες προγραμματισμού. Ένα ελάχιστο επίπεδο κατανόησης κάθε μεμονωμένης έννοιας είναι σημαντικό για την ανάπτυξη της ικανότητας συγγραφής προγράμματος. Απαιτείται επαρκής επίδοση σε έννοιες προγραμματισμού ώστε να μπορούν οι μαθητές να γράψουν προγράμματα (Cabo, 2015).

Υπάρχουν και άλλοι παράγοντες που μπορούν επίσης να συμβάλουν στην ικανότητα συγγραφής προγραμμάτων. Όπως, η ικανότητα κατανόησης ενός προβλήματος, με τη μεταφορά ενός λεκτικού προβλήματος σε αλγόριθμο, είναι πιθανό να συμβάλει στην επίδοση των φοιτητών στη σύνταξη προγραμμάτων. Άλλοι ερευνητές έχουν δείξει ότι η κατανόηση της επίλυσης προβλήματος είναι απαραίτητη προϋπόθεση για τη σύνταξη ενός προγράμματος. Οι μαθητές θα πρέπει να είναι ικανοί σε αρκετές, αν όχι σε όλες τις εννοιολογικές κατηγορίες ενώ κάποιιοι από αυτούς δεν διαθέτουν μεταγνωστικές δεξιότητες και ως εκ τούτου δεν γνωρίζουν τη δική τους γνώση (Cabo, 2015).

Οι αρχάριοι έχουν ποικίλες γνωστικές δεξιότητες. Όσοι διαθέτουν καλές γνωστικές δεξιότητες μαθαίνουν εύκολα, ενώ οι υπόλοιποι μπορεί να απαιτούν τακτική ανάδραση και παρέμβαση από τους εκπαιδευτικούς. Η αποτυχία των εκπαιδευτικών να διορθώσουν έγκαιρα τις παρερμηνείες των μαθητών επιδεινώνει τις μαθησιακές δυσκολίες και οδηγεί σε κακές επιδόσεις στις αξιολογήσεις του προγραμματισμού. Η αποφυγή αυτών των προβλημάτων απαιτεί μια νέα προσέγγιση στο σχεδιασμό της αξιολόγησης. Οι δραστηριότητες χρειάζεται να σχεδιάζονται με βάση τις μαθησιακές ανάγκες των διαφορετικών ομάδων εκπαιδευόμενων, βαθμιδωτά σε επίπεδο δυσκολίας και σε περιβάλλον οπτικού προγραμματισμού. Η αξιολόγηση των μαθησιακών αποτελεσμάτων έδειξε ότι οι σταδιακές οπτικές εποικοδομητικές εργασίες μπορούν να βοηθήσουν στη μείωση του μαθησιακού χάσματος διαφορετικών ομάδων σπουδαστών. Επίσης οι βελτιωμένες επιδόσεις και η εμπλοκή μπορούν να αποδοθούν εν μέρει σε οπτικές δραστηριότητες που αντανακλούν τα αναπτυσσόμενα νοητικά μοντέλα. Το κύριο όφελος αυτής της προσέγγισης είναι ότι οι μαθητές

είναι ελεύθεροι να επιλέξουν τις δικές τους διαδρομές προς τον τελικό στόχο προχωρώντας με το δικό τους ρυθμό. Οι μαθητές που χρειάζονται περισσότερη γνωστική υποστήριξη μπορούν να επιλέξουν εργασίες που παρέχουν μεγαλύτερο επίπεδο οπτικής ανάδρασης και γνωστικής υποστήριξης. Ανάλογα με τη σύνθεση της ομάδας των σπουδαστών, ο εκπαιδευτικός μπορεί να μεταβάλει τον αριθμό και τον τύπο των πρόσθετων εργασιών. Βέβαια η προσέγγισή αυτή περιορίζει τη σειρά με την οποία μπορούν να μάθουν τις έννοιες (Thevathayan & Hamilton, 2015).

Η αξιολόγηση διαφορετικών πτυχών της ιεραρχίας ανάγνωσης του προγράμματος, του λεξιλογίου, της κατανόησης και της ευχέρειας μπορεί να εφαρμοστεί σε ένα ευρύ φάσμα γλωσσών προγραμματισμού. Ωστόσο, αυτές οι ικανότητες πρέπει να μαθαίνονται σταδιακά και να αναπτύσσονται μέσω της πρακτικής άσκησης. Οι προγραμματιστές εξετάζουν την ιεραρχική δομή που δημιουργείται από τα εμφωλευμένα τμήματα προγραμμάτων. Αυτή η δομική άποψη συνδέεται τόσο με τη λογική που υποστηρίζει την οργάνωση του προγράμματος, όσο και με τη διαμόρφωση του με εσοχές. Διαπιστώθηκε εξοικείωση με το λεξιλόγιο των ιεραρχιών και ακόμα μεγαλύτερη στη κατανόηση των όρων στο πλαίσιο του προγράμματος παρά στη επίσημη διατύπωση των ορισμών τους (Park, Kim, Chhabra, Lee, & Forte, 2016).

Σε ορισμένες χώρες αλλά και στην Ελλάδα εφαρμόζεται η πρωτοβουλία "Ωρα του Κώδικα", έτσι δίνεται ευκαιρία για διδασκαλία και αξιολόγηση των δεξιοτήτων προγραμματισμού σε μαθητές Γυμνασίου. Η αξιολόγηση της μαθησιακής συμπεριφοράς των μαθητών και το ενδιαφέρον που εξέφρασαν, οδήγησαν στην ιδέα να προσαρμοστεί η μεθοδολογία και να διδαχθεί ο προγραμματισμός σύμφωνα με τις συγκεκριμένες ανάγκες τους. Ως εκ τούτου, ένα απλό εξατομικευμένο πλαίσιο σχεδιάστηκε και υλοποιήθηκε με σκοπό την παροχή εξατομικευμένου μαθησιακού περιβάλλοντος για την εκπαίδευση στον προγραμματισμό (Mallios & Vassilakoroulos, 2015).

2. Ταξινόμηση εκπαιδευτικών στόχων

2.1 Η αρχική ταξινόμια Bloom

Η ταξινόμια μαθησιακών στόχων αποτελεί ένα πλαίσιο για την ταξινόμηση των απαιτήσεων, τι αναμένεται ή προτίθεται να μάθουν οι μαθητές ως αποτέλεσμα της διδασκαλίας. Ο Bloom, (1956), πρότεινε ένα πλαίσιο ταξινόμιας για τη συγγραφή μαθησιακών στόχων, το οποίο υιοθετήθηκε σε πολλούς εκπαιδευτικούς τομείς.

Η ταξινόμια του Bloom περιγράφηκε για πρώτη φορά ως ιεραρχικό μοντέλο για τον γνωστικό τομέα. Η αρχική ταξινόμια παρείχε προσεκτικά ορισμούς με μορφή ουσιαστικού για κάθε μία από τις έξι μεγάλες κατηγορίες στον γνωστικό τομέα (Krathwohl, 2010). Οι κατηγορίες αυτές είναι: Γνώση, Κατανόηση, Εφαρμογή, Ανάλυση, Σύνθεση και Αξιολόγηση. Με εξαίρεση την κατηγορία Εφαρμογή, καθεμιά από τις υπόλοιπες χωρίζεται σε υποκατηγορίες. Οι κατηγορίες ταξινομήθηκαν από την απλούστερη στην περίπλοκη και από την πιο συγκεκριμένη στην πιο αφηρημένη. Περαιτέρω, θεωρείται ότι η αρχική ταξινόμια αντιπροσωπεύει συσσωρευτική ιεραρχία, δηλαδή, η κατάκτηση της κάθε απλούστερης κατηγορίας είναι προϋπόθεση για την κυριαρχία της επόμενης πιο περίπλοκης. Τα επίπεδα της αρχικής ταξινόμιας είναι τα εξής (Bloom, 1956):

- 1) Γνώση: Η γνώση ορίζεται ως η ανάκληση προηγούμενων δεδομένων ή πληροφοριών. Η γνώση αντιπροσωπεύει το χαμηλότερο επίπεδο μαθησιακών αποτελεσμάτων στον γνωστικό τομέα.
 - 2) Κατανόηση: Η κατανόηση ορίζεται ως η πραγματική κατανόηση μιας έννοιας. Σημαίνει ότι ο μαθητής μπορεί να ερμηνεύσει την έννοια με τα δικά του λόγια.
 - 3) Εφαρμογή: Η εφαρμογή ορίζεται ως η ικανότητα χρήσης μιας έννοιας που έχει μάθει σε μια νέα κατάσταση.
 - 4) Ανάλυση: Η ανάλυση ορίζεται ως η ικανότητα διαχωρισμού των στοιχείων ή των εννοιών σε συστατικά μέρη ώστε να κατανοηθεί η οργανωτική του δομή.
 - 5) Σύνθεση: Η σύνθεση ορίζεται ως η ικανότητα να τοποθετηθούν τα συστατικά μαζί για να σχηματίσουν ένα νέο σύνολο.
 - 6) Αξιολόγηση: Η αξιολόγηση ορίζεται ως η ικανότητα να κρίνει την αξία του υλικού για δεδομένο σκοπό.
-
- ✓ Στο επίπεδο γνώσης ο μαθητής είναι σε θέση να αναγνωρίσει ή να ανακαλέσει πληροφορίες χωρίς να είναι απαραίτητη οποιαδήποτε κατανόηση γι 'αυτό.
 - ✓ Στο επίπεδο κατανόησης ο μαθητής είναι σε θέση να κατανοήσει και να εξηγήσει την έννοια των πληροφοριών που έλαβε.

- ✓ Στο επίπεδο εφαρμογής ο μαθητής είναι σε θέση να επιλέξει και να χρησιμοποιήσει δεδομένα και μεθόδους για την επίλυση μιας συγκεκριμένης εργασίας ή ενός προβλήματος.
- ✓ Στο επίπεδο ανάλυσης ο μαθητής μπορεί να διακρίνει, να ταξινομεί και να συσχετίζει τις πληροφορίες που δίνονται και να αποσυνθέτει ένα πρόβλημα στα μέρη του.
- ✓ Στο επίπεδο σύνθεσης ο μαθητής είναι σε θέση να γενικεύσει ιδέες και να τις εφαρμόσει για να λύσει ένα νέο πρόβλημα.
- ✓ Στο επίπεδο αξιολόγησης ο μαθητής είναι σε θέση να συγκρίνει και να αξιολογήσει μεθόδους ή λύσεις για την επίλυση ενός προβλήματος ή την επιλογή της καλύτερης λύσης.

Η αρχική ταξινομία Bloom είναι μια ευρέως αποδεκτή ταξινόμηση για την εκτίμηση του γνωστικού επιπέδου που επιτεύχθηκε για έναν μαθητή σε ένα συγκεκριμένο θέμα. Σύμφωνα με τον Bloom, (1956), κάθε επίπεδο γνωστικής ανάπτυξης εξαρτάται από τις συμπεριφορές και τις γνώσεις που αποκτώνται στα προηγούμενα επίπεδα. Η γνωστική ανάπτυξη των μαθητών στον προγραμματισμό θα πρέπει να λαμβάνεται υπόψη κατά το σχεδιασμό των μαθησιακών δραστηριοτήτων προγραμματισμού. Η ταξινομία Bloom έχει χρησιμοποιηθεί στον προγραμματισμό ως πλαίσιο για διαφορετικούς σκοπούς: τον σχεδιασμό των μαθημάτων, την αξιολόγηση των φοιτητών/μαθητών ή τον σχεδιασμό εκπαιδευτικών εφαρμογών.

2.1. Η αναθεωρημένη ταξινόμια Bloom

Αρχικά ο όρος ταξινόμια ήταν άγνωστος ως εκπαιδευτικός όρος και πολλοί χρήστες δεν κατάλαβαν τι σήμαινε, γι' αυτό δόθηκε ελάχιστη προσοχή στην αρχική ταξινόμηση. Όμως, καθώς οι ερευνητές και οι εκπαιδευτικοί είδαν το δυναμικό της, το πλαίσιο έγινε ευρέως γνωστό και έγιναν αναφορές, τελικά μεταφράστηκε σε 22 γλώσσες. Το μοντέλο επανεξετάστηκε από τον Anderson (2001) και μια ομάδα γνωστικών ψυχολόγων. Ως αποτέλεσμα, έγιναν ορισμένες σημαντικές αλλαγές στην ορολογία και τη δομή της ταξινόμιας. Ο αρχικός αριθμός των έξι κατηγοριών, διατηρήθηκε, με αλλαγές. Τρεις κατηγορίες μετονομάστηκαν, η σειρά δύο κατηγοριών αντικαταστάθηκε και τα ονόματα των κατηγοριών άλλαξαν σε μορφή ρήματος για να ταιριάζουν με τον τρόπο που χρησιμοποιούνται στους μαθησιακούς στόχους.

Όπως η αρχική ταξινόμια, έτσι και η αναθεωρημένη αποτελεί ιεραρχία με την έννοια ότι οι έξι κύριες κατηγορίες διαφέρουν ως προς την πολυπλοκότητά τους. Επειδή η αναθεώρηση δίνει μεγαλύτερη βαρύτητα στη χρήση των μαθησιακών στόχων, η απαίτηση μιας αυστηρής ιεραρχίας έχει χαλαρώσει έτσι οι κατηγορίες επικαλύπτονται η μια από την άλλη (Krathwohl, 2010).

Οι Anderson et al. (2001) παρέχουν παραδείγματα για το πώς οι γνώσεις και οι γνωστικές κατηγορίες εφαρμόζονται σε διάφορους τομείς θεματικών περιοχών. Η πληροφορική και ο προγραμματισμός δεν συγκαταλέγονται στους τομείς που καλύπτονται. Σε πολλές έρευνες έγινε προσπάθεια να περιγραφούν οι κατηγορίες της ταξινόμιας Bloom χρησιμοποιώντας παραδείγματα για τον προγραμματισμό.

Οι Thompson, Grove, Luxton-reilly, Whalley, & Robbins, (2008) αναλύουν κάθε μία από τις κατηγορίες της ταξινόμιας Bloom και παρέχουν μια συνεπή ερμηνεία με συγκεκριμένα παραδείγματα που θα επιτρέψουν στους εκπαιδευτικούς της Πληροφορικής να χρησιμοποιούν την ταξινόμια του Bloom για την αξιολόγηση των γνώσεων προγραμματισμού. Θεωρούν σημαντικό να αναπτυχτεί μια κοινή αντίληψη για το πώς ερμηνεύεται η αναθεωρημένη ταξινόμια Bloom στον τομέα της επιστήμης των υπολογιστών. Επίσης οι Hernán-Losada, Pareja-Flores, & Velázquez-Iturbide, (2008) προτείνουν κατάλληλες δραστηριότητες προγραμματισμού για αξιολόγηση στα επίπεδα κατανόησης, εφαρμογής, ανάλυσης και σύνθεσης στην αρχική ταξινόμια Bloom. Η πρόταση βασίζεται σε είδη δραστηριοτήτων και προτεινόμενα παραδείγματα καθώς και ένα σύνολο χαρακτηριστικών που οι δραστηριότητες πρέπει να περιλαμβάνουν ως μεταδεδομένα για να διευκολύνουν τη χρήση τους με παιδαγωγικά κριτήρια.

Οι κατηγορίες και η ερμηνεία τους στον προγραμματισμό περιγράφονται ως εξής:

- Το επίπεδο «Θυμάμαι» περιγράφεται ως ανάκτηση σχετικής γνώσης από μακροχρόνια μνήμη. Η κατηγορία αυτή περιλαμβάνει την αναγνώριση και την ανάκληση. Σε αυτές τις περιπτώσεις, οι μαθητές καλούνται να εκτελούν εργασίες που απαιτούν γνώσεις που έχουν μάθει.

Ο καθορισμός αν μια δραστηριότητα ανήκει σε αυτή την κατηγορία συχνά απαιτεί λεπτομερή γνώση της διδακτέας ύλης του μαθήματος, καθώς ο σημαντικότερος παράγοντας για αυτή την κατηγορία είναι αν ο μαθητής γνωρίζει από προηγούμενο μάθημα τη λύση της δραστηριότητας. Αν η δραστηριότητα μπορεί να ολοκληρωθεί απλώς ανακαλώντας κάτι, η εργασία αξιολόγησης ανήκει σε αυτήν την κατηγορία.

- Το επίπεδο «Κατανόω» περιγράφεται ως κατασκευή προγράμματος ή αλγορίθμου από δοσμένα δεδομένα (τμήματα προγραμμάτων, δομές επιλογής, δομές επανάληψης, διαγράμματα ροής) τα οποία χρειάζεται να τροποποιηθούν. Αυτή η κατηγορία περιλαμβάνει ερμηνεία, διευκρίνιση, ταξινόμηση, ανακεφαλαίωση, συμπέρασμα. Τη μετατροπή ενός αλγορίθμου από μια μορφή αναπαράστασης σε άλλη μορφή ή μιας δομής σε άλλη ισοδύναμη, εξήγηση μιας έννοιας ή ενός αλγορίθμου ή ενός προγράμματος.

- Το επίπεδο «Εφαρμόζω» περιγράφεται ως εκτέλεση ή χρήση μιας διαδικασίας σε μια δεδομένη κατάσταση. Η κατηγορία αυτή περιλαμβάνει εκτέλεση και εφαρμογή. Η διαδικασία και ο αλγόριθμος ή το πρόγραμμα είναι γνωστά στον μαθητή, και τα δύο εφαρμόζονται σε ένα πρόβλημα που είναι γνωστό, αλλά που δεν έχει λυθεί προηγουμένως στο ίδιο πλαίσιο ή με τα ίδια δεδομένα ή με τα ίδια εργαλεία, ή εφαρμόζονται σε άγνωστο πρόγραμμα.

- Το επίπεδο «Αναλύω» περιγράφεται ως η διάσπαση του υλικού στα συστατικά μέρη του και ο καθορισμός του τρόπου με τον οποίο τα μέρη συνδέονται μεταξύ τους και με μια γενική δομή ή σκοπό. Αυτή η κατηγορία περιλαμβάνει τη διαφοροποίηση, την οργάνωση και τον προσδιορισμό. Να διασπαστεί μια δραστηριότητα προγραμματισμού στα συστατικά μέρη της (εντολές εκχώρησης, δομές επιλογής, δομές επανάληψης). Να οργανωθούν τα συστατικά μέρη για την επίτευξη ενός γενικού στόχου, να προσδιοριστούν τα σημαντικά ή τα μη σημαντικά στοιχεία του προβλήματος.

- Το επίπεδο «Αξιολογώ» περιγράφεται με το να γίνονται κρίσεις με βάση κριτήρια και πρότυπα. Στην αναθεωρημένη ταξινομία, αυτή η κατηγορία περιλαμβάνει τον έλεγχο και την κριτική. Καθορίζει σε ποιο βαθμό ένα τμήμα προγράμματος ικανοποιεί τις απαιτήσεις του προβλήματος μέσω δοκιμών, κρίνει την απόδοση ενός προγράμματος βασισμένου σε πρότυπα κωδικοποίησης ή κριτήρια απόδοσης.

- Το επίπεδο «Δημιουργώ» περιγράφεται ως ο συνδυασμός στοιχείων μαζί για να σχηματίσουν ένα συνεκτικό ή λειτουργικό σύνολο, αναδιοργανώνοντας τα στοιχεία σε ένα νέο σχέδιο ή δομή. Στην αναθεωρημένη ταξινομία, η κατηγορία αυτή περιλαμβάνει τη δημιουργία, το σχεδιασμό και την παραγωγή. Επιδιώκεται με ένα νέο εναλλακτικό αλγόριθμο ή υποθέτοντας ότι ένας νέος συνδυασμός αλγορίθμων θα λύσει ένα πρόβλημα. Γίνεται με επινόηση εναλλακτικής διαδικασίας ή στρατηγικής για την επίλυση ενός προβλήματος. Επίσης σύνθετες εργασίες

προγραμματισμού, που μπορεί να περιλαμβάνουν τη διαίρεση της εργασίας σε μικρότερα τμήματα στα οποία μπορούν να εφαρμοστούν γνωστοί αλγόριθμοι και διαδικασίες.

2.2 Η εφαρμογή των ταξινομιών Bloom στον προγραμματισμό

Οι ταξινομίες Bloom (αρχική και αναθεωρημένη) έχουν χρησιμοποιηθεί ως σημείο αναφοράς ή πλαίσιο για έρευνα σε πολλές μελέτες που αφορούν τον προγραμματισμό. Παρακάτω αναφέρονται έρευνες στις οποίες χρησιμοποιούνται οι ταξινομίες αυτές για την εκμάθηση του προγραμματισμού.

Οι Hwang, Wang, Hwang, Huang, & Huang (2008) προτείνουν μια καινοτόμο προσέγγιση για τη γνωστική ανάπτυξη της εκμάθησης προγραμματισμού, με τη χρήση ενός διαδικτυακού συστήματος που υποστηρίζει μαθησιακές δραστηριότητες προγραμματισμού με διαφορετικά επίπεδα δυσκολίας, από απλές σε πολύπλοκες, βασισμένο στην αρχική ταξινόμια Bloom. Το κίνητρο ήταν να παρέχει κατάλληλες δραστηριότητες και επαρκή πρακτική στους μαθητές ώστε να τους βοηθήσει να αναπτύξουν τις γνώσεις τους στον προγραμματισμό.

Επίσης οι Hernán-Lozada, Lázaro-Carrascosa, & Velázquez-Iturbide (2008) εξετάζουν τη συστηματική σχεδίαση και ανάπτυξη εκπαιδευτικών εφαρμογών μέσα στο παιδαγωγικό πλαίσιο που παρέχεται από την ταξινόμια Bloom. Ως τομέας εφαρμογής, επιλέχθηκε ο αντικειμενοστραφής προγραμματισμός. Συγκεκριμένα περιγράφεται μια εκπαιδευτική εφαρμογή για εκμάθηση της έννοιας της κληρονομιάς. Η εφαρμογή περιλαμβάνει διάφορες ενότητες που στοχεύουν στα τρία χαμηλότερα επίπεδα της αρχικής ταξινόμιας Bloom.

Οι Ortiz, Frances, & Martin (2008) περιγράφουν μια πρόταση διδασκαλίας που εισάγει μια διαδικασία, η οποία βασίζεται στο πέρασμα των διαφόρων κατηγοριοποιημένων τύπων προβλημάτων, στους μαθησιακούς στόχους και στους διάφορους τύπους γνώσης σύμφωνα με την ταξινόμια Bloom.

Η διόρθωση σφαλμάτων είναι πάντα δύσκολη για τους αρχαίους προγραμματιστές και οι Xu & Rajlich (2004) μελετώντας τη γνωστική διαδικασία κατά τη διάρκεια εντοπισμού σφαλμάτων, υποστηρίζουν ότι οι αρχάριοι πρέπει να αφιερώνουν χρόνο αποκτώντας και κατανοώντας τη γνώση, αφού η επιτυχής διόρθωση απαιτεί τη χρήση όλων των γνωστικών επιπέδων της ταξινόμιας Bloom.

Οι Gluga, Kay, Lister, Kleitman, & Lever (2012b) κατέληξαν στο συμπέρασμα ότι οι εκπαιδευτικοί που επιδιώκουν να χρησιμοποιήσουν την ταξινόμια του Bloom χρειάζεται να εκπαιδευτούν. Για το σκοπό αυτό, δημιούργησαν ένα διαδικτυακό σεμινάριο ταξινόμια Bloom με διαδραστικά παραδείγματα, αυτο-εξήγηση και αυτο-ανάδραση των χρηστών. Τα αποτελέσματα από την αξιολόγηση του σεμιναρίου είναι χρήσιμα για να εντοπιστεί πού χρησιμοποιείται η ταξινόμια ασυνεπώς λόγω διαφορετικών υποθέσεων σχετικά με το περιβάλλον μάθησης, τις διαφορετικές ερμηνείες των κατηγοριών Bloom ή την παρερμηνεία των κατηγοριών. Συνολικά, το σεμινάριο ήταν επιτυχές ως προς τον τρόπο εφαρμογής της ταξινόμιας Bloom στις ερωτήσεις προγραμματισμού. Τα αποτελέσματα επιπροσθέτως όμως υπογραμμίζουν σημαντικούς τομείς στην εφαρμογή της

ταξινομίας Bloom που δημιουργούν προβλήματα στην αξιολόγηση γνώσεων προγραμματισμού (Gluga, Kay, Lister, Kleitman, & Lever, 2012a).

Οι ίδιοι ερευνητές σε επόμενη έρευνα τους (Gluga, et al., 2013) μελετούν κατά πόσο θα συμβάλει το σεμινάριο στο να αναπτύξουν οι εκπαιδευτικοί γρήγορα και αποτελεσματικά την κατανόηση στην αρχική ταξινομία Bloom έτσι ώστε να μπορούν να την εφαρμόζουν με συνέπεια στις αξιολογήσεις τους, όπως είναι οι εργασίες των εξετάσεων. Κύριος σκοπός ήταν η δυνατότητα αναθεώρησης των εξετάσεων και ο μελλοντικός σχεδιασμός τους με μεγαλύτερη συνειδητοποίηση των γνωστικών επιπέδων.

Ένα μοντέλο ενοποίησης δύο προσεγγίσεων για μαθήματα μεγάλης κλίμακας λογισμικού περιγράφει ο Biggs (2014) χρησιμοποιώντας την εποικοδομητική ευθυγράμμιση σε συνδυασμό με την αναθεωρημένη ταξινομία Bloom. Οι φοιτητές είχαν περισσότερα κίνητρα να επικεντρωθούν σε μαθησιακά αντικείμενα και να αποδεχθούν μαθησιακές δραστηριότητες. Δημιουργήθηκε επίσης αποθετήριο για τον εκπαιδευτικό λαμβάνοντας υπόψη όλα τα επίπεδα της διάστασης της γνωστικής διαδικασίας και της κατηγορίας γνώσης. Το αποθετήριο παρέχει τη δυνατότητα αναθεώρησης των δραστηριοτήτων αξιολόγησης.

Η διδασκαλία ασφαλών εννοιών κωδικοποίησης παράλληλα με τις θεμελιώδεις έννοιες προγραμματισμού εξασφαλίζει ότι οι φοιτητές δημιουργούν ισχυρές δεξιότητες προγραμματισμού. Στον τομέα της ασφαλούς κωδικοποίησης ως μέρος του προγράμματος σπουδών, αναφέρεται η έρευνα των Taylor & Kaza (2011) με την παρουσίαση ενός μοντέλου που χρησιμοποιεί διδακτικές ενότητες ασφάλειας βασισμένες σε λίστες ελέγχου για να αυξηθεί η ευαισθητοποίηση των φοιτητών αλλά κυρίως η ικανότητα εφαρμογής ασφαλών αρχών κωδικοποίησης. Το μοντέλο αξιολογήθηκε με τη χαρτογράφηση ερωτήσεων αξιολόγησης σύμφωνα με αναθεωρημένη ταξινομία Bloom. Οι φοιτητές που χρησιμοποίησαν αυτές τις διδακτικές ενότητες είχαν καλύτερη απόδοση στην απομνημόνευση, την κατανόηση και την εφαρμογή ασφαλών εννοιών κωδικοποίησης αλλά και περισσότερες δεξιότητες να γράφουν προγράμματα για την αντιμετώπιση συγκεκριμένων ζητημάτων ασφάλειας.

Οι Teodosiev & Nachev (2015) διερευνούν το ρόλο του αλγορίθμου στα εισαγωγικά μαθήματα προγραμματισμού. Λόγω της φύσης των δραστηριοτήτων, είναι δύσκολο να πεισθούν οι αρχάριοι για την ιδιαίτερη σημασία ανάπτυξης του αλγορίθμου. Ως εκ τούτου, ένα σημαντικό μεθοδολογικό ζήτημα είναι η προσεκτική επιλογή δραστηριοτήτων και παραδειγμάτων, που θα παρακινούν τους μαθητές να προσεγγίσουν τα προβλήματα που εξετάζουν και να εκτιμήσουν διαφορετικές εναλλακτικές λύσεις αλγορίθμων πριν τη δημιουργία προγράμματος. Για να το σκοπό αυτό, παρέχονται παραδείγματα διδασκαλίας αναλύοντας τα μαθησιακά αποτελέσματα. Γίνεται αναφορά μεταξύ των επιπέδων της αναθεωρημένης ταξινομίας Bloom και των δεξιοτήτων των

μαθητών στον αλγόριθμο και στον προγραμματισμό. Συζητούνται επίσης ξεχωριστά τα επίπεδα της ταξινόμιας στο πλαίσιο του αλγορίθμου και της εφαρμογής του. Γίνεται σαφές ότι το κύριο επίκεντρο ενός εισαγωγικού μαθήματος προγραμματισμού θα πρέπει να κατευθύνεται προς τον αλγόριθμο, ο οποίος απαιτεί λογική και μαθηματική κουλτούρα.

Οι Shargabi, Aljunid, Annamalai, Shuhidan, & Zin (2016) προσδιορίζουν δραστηριότητες οι οποίες ταξινομούνται σε ομοιογενείς κατηγορίες που βασίζονται σε παρόμοιες γνωστικές δεξιότητες. Η ταξινόμηση γίνεται με την αντιστοίχιση των δραστηριοτήτων με τις κατηγορίες της αναθεωρημένης ταξινόμιας Bloom. Πραγματοποιήθηκε έρευνα για την ταξινόμηση αυτών των δραστηριοτήτων σε κάθε κατηγορία Bloom με βάση την πιθανή αποτελεσματικότητά τους στην ανάπτυξη της κατανόησης του προγράμματος. Τα αποτελέσματα έδειξαν ότι οι δραστηριότητες που εφαρμόζονται συχνότερα στη διδασκαλία, έλαβαν υψηλότερες βαθμολογίες. Από τα συμπεράσματα διαφαίνεται ότι οι δεξιότητες κατανόησης του προγράμματος μπορούν να βελτιωθούν μέσω δραστηριοτήτων για διάφορους τύπους εργασιών. Κατόπιν γίνεται επικύρωση της ταξινόμησης των δραστηριοτήτων κατά την οποία προκύπτει συμφωνία μεταξύ των εκπαιδευτικών που έκαναν την ταξινόμηση στις περισσότερες κατηγορίες (Saroni, Aljunid, Shuhidan, & Shargabi, 2016).

Η ταξινόμια Bloom έχει χρησιμοποιηθεί τόσο στην εκμάθηση του προγραμματισμού όσο και στην αξιολόγηση του. Ο Alhazbi (2016) παρουσιάζει ένα διαδικτυακό σύστημα διδασκαλίας το οποίο χρησιμοποιεί τα γνωστικά επίπεδα της ταξινόμιας τόσο στη διδασκαλία όσο και στην αξιολόγηση, για την υποστήριξη φοιτητών που μαθαίνουν προγραμματισμό. Κάθε μάθημα σχετίζεται με συγκεκριμένους μαθησιακούς στόχους. Καινοτομία του συστήματός αποτελεί η χρήση των γνωστικών επιπέδων της αναθεωρημένης ταξινόμιας ως παράγοντα προσαρμογής στην πρόοδο των μαθητών, όπου το σύστημα μετακινεί το φοιτητή από απλά επίπεδα σε πιο σύνθετα. Σε κάθε μάθημα, το σύστημα στοχεύει να μετακινήσει τον εκπαιδευόμενο από τα χαμηλότερα επίπεδα στα υψηλότερα εφόσον περάσει όλες τις αξιολογήσεις σε αυτό ενώ σε περίπτωση αποτυχίας δίνει ανατροφοδότηση. Μόνο όταν ο εκπαιδευόμενος αξιολογηθεί επιτυχώς σε ένα επίπεδο προχωράει στην αξιολόγηση του επόμενου επιπέδου.

Σε πολλές άλλες έρευνες η ταξινόμια χρησιμοποιείται για αξιολόγηση γνώσεων και δεξιοτήτων. Σύμφωνα με τον Alaoutinen (2010) μια κλίμακα αυτοαξιολόγηση βασισμένη στην αναθεωρημένη ταξινόμια, εξετάζει παράγοντες που επηρεάζουν την ακρίβεια αξιολόγησης και την επίδοση στα μαθήματα. Η κλίμακα αξιολογεί συγκρίνοντας τα αποτελέσματα της αυτοαξιολόγησης των μαθητών με τις επιδόσεις τους σε ένα μάθημα προγραμματισμού. Ουσιαστικά ερευνάται η δυνατότητα χρήσης μια ταξινόμιας ως βάση μιας κλίμακας αξιολόγησης προκειμένου να επιτευχθεί μεγαλύτερη αντικειμενικότητα. Η ταξινόμια παρέχει ακριβείς περιγραφές των κατηγοριών της και δεν δίνει

περαιτέρω περιθώρια για ερμηνεία. Η κλίμακα αξιολόγησης γενικεύει ερωτήσεις και δραστηριότητες που προτείνονται σε προηγούμενες μελέτες (Thompson et. al.,2008, Hernán-Losada et al.,2008).

Επίσης οι Alaoutinen & Nikula (2009), περιγράφουν ένα πείραμα, που έχει ως στόχο να αναπτύξει ένα αξιόπιστο διαδικτυακό σύστημα αυτοαξιολόγησης για το μάθημα του προγραμματισμού. Συγκρίνεται μια απλή κλίμακα τεσσάρων βαθμίδων με μια κλίμακα έξι βαθμίδων που τροποποιήθηκε σύμφωνα με την ταξινόμια Bloom. Και οι δύο κλίμακες έδωσαν παρόμοια αποτελέσματα, αλλά η κλίμακα που βασιζόταν στην αναθεωρημένη ταξινόμια έδειξε να υπερεκτιμά τις δεξιότητες των μαθητών.

Την αξιολόγηση γνώσεων προγραμματισμού στην ανάπτυξη εφαρμογών για κινητά με το προγραμματιστικό περιβάλλον App Inventor, βασισμένη στην αναθεωρημένη ταξινόμια Bloom, η οποία περιελάμβανε την αυτοαξιολόγηση των μαθητών, τις αξιολογήσεις από τους εκπαιδευτές καθώς και τις συγκρίσεις των δύο αποτελεσμάτων περιγράφει ο Honig (2013). Τα αποτελέσματα της αξιολόγησης έδειξαν ότι ήταν ιδιαίτερα επιτυχής, στις τρεις χαμηλότερες κατηγορίες αξιολόγησης: θυμάμαι, κατανοώ και εφαρμόζω αλλά και αποδεκτή στις ανώτερες κατηγορίες: αναλύω, αξιολογώ και δημιουργώ. Προτείνεται βελτίωση του περιεχομένου του μαθήματος για να προστεθεί ακρίβεια σε κάθε κατηγορία αξιολόγησης με επιπρόσθετα παραδείγματα των συστατικών της εφαρμογής, καθώς η αξιολόγηση δεν περιελάμβανε καμία αξιολόγηση ορισμένων βασικών στοιχείων προγραμματισμού (όπως δομές ελέγχου, λογικές εκφράσεις, υποπρογράμματα και παράμετρους). Η προσθήκη αυτών των στοιχείων στην αξιολόγηση, πιθανόν στις κατηγορίες αναλύω και αξιολογώ, θα μπορούσε να διευρύνει τη διαδικασία αξιολόγησης.

Η έρευνα των Shuhidan, Hamilton, & D'Souza (2009) περιλαμβάνει ερωτήσεις που προέρχονται από τελικές εξετάσεις προγραμματισμού αρχαρίων. Οι ερωτήσεις κατηγοριοποιούνται και ταξινομούνται στο πλαίσιο της αρχικής ταξινόμιας Bloom. Επιδιώκεται η κατανόηση των αιτιών των σφαλμάτων που κάνουν οι μαθητές. Αρχικά, η διαδικασία αφορά τις ερωτήσεις πολλαπλής επιλογής και τις απαντήσεις των συμμετεχόντων και την κατάταξη τους σύμφωνα με την αρχική ταξινόμια Bloom. Κατόπιν γίνεται επέκταση ώστε να διερευνηθεί η ικανότητα κατανόησης των αρχαρίων σε γραπτά τμήματα προγράμματος. Ωστόσο, αν και εφαρμόστηκε η αρχική ταξινόμια του Bloom, αναγνωρίστηκε ότι υπάρχουν επιπλέον εξηγήσεις των γνωστικών απαιτήσεων που αναφέρονται στην αναθεωρημένη έκδοση. Εφαρμόστηκαν οι κατευθυντήριες γραμμές της γνωστικής ταξινόμιας στις ερωτήσεις που αποτελούσαν την τελική εξέταση. Από τα ευρήματα της έρευνας φαίνεται ότι το περιεχόμενο των ερωτήσεων πολλαπλής επιλογής μπορεί να ταξινομηθεί κυρίως στα τρία χαμηλότερα επίπεδα: θυμάμαι, κατανοώ και εφαρμόζω.

Οι μαθητές αποδίδουν καλύτερα σε αξιολογήσεις εννοιών παρά σε αξιολογήσεις εφαρμογής των εννοιών αυτών στη σύνταξη προγραμμάτων. Οι επιδόσεις σε έννοιες μπορούν να καταταχθούν στα πρώτα δύο επίπεδα της ταξινομίας Bloom (επίπεδο γνώσης και κατανόησης) και οι επιδόσεις στις δεξιότητες μπορούν να χαρτογραφηθούν στα επίπεδα τρία και τέσσερα (επίπεδο εφαρμογής και ανάλυσης)(Cabo, 2015).

2.3 Δυσκολίες στη χρήση της ταξινομίας Bloom

Η χρήση της ταξινομίας Bloom στο πεδίο της πληροφορικής θεωρήθηκε αντιφατική. Η αρχική ταξινόμηση συμβάλλει στην ταξινόμηση της δυσκολίας του προγράμματος στο επίπεδο σκέψης, ενώ η αναθεωρημένη συμβάλλει στην ταξινόμηση με βάση το επίπεδο γνώσεων ή δεξιοτήτων που απαιτούνται για την επίλυση μιας δραστηριότητας. Όταν εφαρμόζεται παρουσιάζονται κάποιες δυσκολίες, όπως είναι η ερμηνεία των επιπέδων της ταξινομίας και η αναγκαιότητα βαθιάς κατανόησης του περιεχομένου του προβλήματος. Επιπλέον, στην περίπτωση που η ταξινόμηση γίνεται από πολλούς εκπαιδευτικούς, η ταύτιση στην κατηγοριοποίηση είναι δύσκολη, επειδή ο κάθε εκπαιδευτικός μπορεί να έχει προϋπάρχουσες αντιλήψεις σχετικά με την πολυπλοκότητα του προβλήματος και του επιπέδου των δεξιοτήτων που απαιτούνται για την επίλυσή του από τους μαθητές (Elnaffar, 2016).

Σύμφωνα με τους Thompson et. al., (2008), διαπιστώθηκαν σημαντικές διαφοροποιήσεις κατά την κατηγοριοποίηση των δραστηριοτήτων αξιολόγησης των γνώσεων προγραμματισμού σε κάθε κατηγορία της ταξινομίας. Αυτό οφείλονταν κυρίως στη δυσκολία χαρτογράφησης των μαθησιακών δραστηριοτήτων στον τομέα του προγραμματισμού. Σε ορισμένες περιπτώσεις, που κάποιοι από τους εκπαιδευτικούς δεν εμπλέκονταν στη διδασκαλία του συγκεκριμένου μαθήματος, δεν ήταν σε θέση να γνωρίζουν το πλαίσιο διδασκαλίας για τη συγκεκριμένη δραστηριότητα αξιολόγησης. Όταν καθορίζονταν το πλαίσιο διδασκαλίας, ήταν δυνατό να υπάρξει συμφωνία στην κατάλληλη γνωστική κατηγορία για τη συγκεκριμένη δραστηριότητα αξιολόγησης.

Ενδεχομένως είναι δύσκολο να γίνει διάκριση μεταξύ των επιπέδων στις κατηγορίες κατά την εφαρμογή της ταξινομίας Bloom σε κάθε δραστηριότητα λόγω της φύσης των εξετάσεων προγραμματισμού (Taylor & Kaza, 2011). Σπάνια υπάρχουν λέξεις-κλειδιά για την ταξινόμηση τα οποία να χρησιμοποιούνται από τον εξεταστή στις ερωτήσεις εξετάσεων. Αναφέρονται δυσκολίες από τους ερευνητές στη διάκριση της κατηγοριοποίησης στα επίπεδα κατανόησης και εφαρμογής στα πρώτα στάδια της μάθησης, επειδή οι αρχάριοι αναμένεται τόσο να κατανοούν όσο και να είναι σε θέση να αποδείξουν τις βασικές γνώσεις. Ως εκ τούτου, διαπιστώθηκε ότι η κατηγοριοποίηση της απάντησης ενός αρχάριου ήταν ένα δύσκολο έργο που δημιουργούσε ενδιαφέρουσες προκλήσεις.

Η αρχική ταξινόμηση Bloom δημιουργήθηκε ώστε να ταιριάζει γενικά στο πεδίο της εκπαίδευσης έτσι ήταν πολύ δύσκολο να ταξινομηθούν οι ερωτήσεις στις τελικές εξετάσεις στον προγραμματισμό με τη χρήση της. Επίσης δυσκολίες εντοπίζονται κατά τη διάκριση μεταξύ των κατηγοριών, καθώς στα θέματα προγραμματισμού οι ερωτήσεις βασίζονται κυρίως σε τμήματα προγραμμάτων. Ειδικότερα στα εισαγωγικά μαθήματα προγραμματισμού, παρότι οι ερωτήσεις ανήκαν στα χαμηλότερα επίπεδα (της γνώσης και της κατανόησης) της ταξινομίας, δημιουργούσαν σύγχυση και ήταν δύσκολο για τους αρχάριους να απαντήσουν σωστά (Shuhidan et al., 2009).

Οι Fuller et al. (2007) επίσης αμφισβητούν εάν τα επίπεδα της ταξινομίας Bloom είναι καλά διατεταγμένα για αξιολόγηση στον προγραμματισμό. Προτείνουν μια νέα ταξινόμηση που επιχειρεί να απεικονίσει καλύτερα την κατανόηση εννοιών. Η ταξινόμηση περιλαμβάνει έναν δισδιάστατο πίνακα που αντιπροσωπεύει δύο ξεχωριστές περιοχές των δεξιοτήτων: τις δεξιότητες κατανόησης και ερμηνείας ενός δοσμένου προγράμματος και τις δεξιότητες σχεδιασμού και δημιουργίας ενός νέου προγράμματος. Η ταξινομία που χρησιμοποιήθηκε είναι μια δισδιάστατη προσαρμογή της αναθεωρημένης ταξινομίας. Στον κατακόρυφο άξονα τοποθετούνται τα επίπεδα που αφορούν την παραγωγή προγράμματος και είναι τα Δημιουργώ, εφαρμόζω, ενώ στον οριζόντιο άξονα είναι τα επίπεδα που αφορούν ερμηνεία και είναι τα θυμάμαι, κατανοώ, αναλύω, αξιολογώ. Η ταξινομία αυτή διαχωρίζει την ικανότητα κατανόησης ενός ήδη υπάρχοντος προγράμματος από την ικανότητα δημιουργίας ενός νέου τμήματος προγράμματος.

Η αξιολόγηση της ταξινόμησης ερωτήσεων προγραμματισμού (Gluga et al., 2012a, 2012b) επιβεβαίωσε τις προηγούμενες τεκμηριωμένες ασάφειες στην εφαρμογή της ταξινομίας Bloom σε περιπτώσεις όπου απαιτείται γνώση σχετικά με το μαθησιακό πλαίσιο για ακριβή ταξινόμηση. Επίσης διαπιστώνεται ότι διαφορετικοί ερευνητές κάνουν διαφορετικές ταξινομήσεις, και αυτό οφείλεται σε προϋπάρχουσες παρανοήσεις των κατηγοριών ή διαφορετικές ερμηνείες σχετικά με την πολυπλοκότητα των δραστηριοτήτων και την πολυπλοκότητα που απαιτείται για την επίλυσή τους.

Οι Gluga et al. (2013) επισημαίνουν ότι η εφαρμογή της ταξινομίας είναι δύσκολη σε περιπτώσεις που δεν είναι γνωστό με ακρίβεια το εκπαιδευτικό πλαίσιο. Δηλαδή, εφ' όσον ο εκπαιδευτικός που κάνει την ταξινόμηση των ερωτημάτων έχει πλήρη επίγνωση του πλαισίου διδασκαλίας (διότι αυτός διδάσκει το γνωστικό αντικείμενο) είναι σε θέση να εφαρμόσει ορθά την ταξινομία, τότε η ταξινόμηση είναι έγκυρη, γιατί δεν χρειάζεται να γίνουν υποθέσεις. Όταν, όμως, η ταξινόμηση ερμηνεύεται από έναν διαφορετικό εκπαιδευτικό ή από έναν συντονιστή προγράμματος που παρακολουθεί την πρόοδο σε όλο το πρόγραμμα σπουδών, δείχνει τον τρόπο με τον οποίο διδάχθηκε ένα γνωστικό αντικείμενο. Επομένως είναι δύσκολο, ή ίσως αδύνατο, να ταξινομηθεί ή να ερμηνευτεί με νόημα η γνωστική δυσκολία των ερωτημάτων αξιολόγησης χωρίς να ληφθεί υπόψη το προβλεπόμενο πλαίσιο διδασκαλίας και μάθησης.

Οι αρχάριοι αντιμετωπίζουν δύο εμπόδια στην πορεία τους για να γίνουν έμπειροι προγραμματιστές: την γνωστική κατανόηση των εννοιών προγραμματισμού, (οι δύο πρώτες κατηγορίες στην ταξινομία Bloom) και την δεξιότητα εφαρμογής των εννοιών αυτών (τρίτη κατηγορία στην ταξινομία του Bloom) για τη σύνταξη σωστών προγραμμάτων. Οι αξιολογήσεις των εννοιών προγραμματισμού ομαδοποιήθηκαν σε επτά διαφορετικές κατηγορίες: ανάθεση, δομές επανάληψης (για, όσο, μέχριςότου), δομές επιλογής (Αν / αλλιώς), μέθοδοι, πίνακες, κλάσεις και

γενική σύνταξη. Σκοπός ήταν να αξιολογηθεί το γνωστικό επίπεδο των σπουδαστών στις δύο πρώτες κατηγορίες της ταξινόμησης του Bloom: γνώση και κατανόηση, σε όλες τις κατηγορίες εννοιών, εκτός από τις δομές επανάληψης και τη σύνταξη, η επίδοση ήταν επαρκής (Cabo, 2015).

Η εφαρμογή της αναθεωρημένης ταξινομίας στους μαθησιακούς στόχους και στα στοιχεία αξιολόγησης στο αντικείμενο του προγραμματισμού δεν είναι απλό έργο. Διαφορετικοί εκπαιδευτικοί θα μπορούσαν να ταξινομήσουν την ίδια ερώτηση ή δραστηριότητα σε διαφορετική γνωστική κατηγορία, αυτό εξαρτάται κατά πόσο ο εκπαιδευτικός έχει κατανοήσει την ταξινόμηση και από τον τρόπο εφαρμογής αυτής της ταξινόμησης στο αντικείμενο του προγραμματισμού (Alhazbi, 2016).

Συνοψίζοντας οδηγούμαστε στα εξής συμπεράσματα:

Η ταξινομία Bloom δημιουργήθηκε για να ταξινομήσει τους μαθησιακούς στόχους γενικότερα στην εκπαίδευση χωρίς να προτείνονται από τους δημιουργούς συγκεκριμένα παραδείγματα στον προγραμματισμό. Παρότι προτάθηκαν σε πολλές έρευνες πολλά παραδείγματα στο πεδίο αυτό είναι δύσκολη η εφαρμογή της επειδή στον προγραμματισμό δεν υπάρχουν λέξεις χαρακτηριστικές που να υποδηλώνουν το επίπεδο μιας δραστηριότητας. Επιπλέον οι ερωτήσεις βασίζονται κυρίως σε τμήματα προγραμμάτων και αυτό καθιστά δύσκολη τη διάκριση μεταξύ των επιπέδων. Ένας άλλος σημαντικός παράγοντας είναι ότι οι εκπαιδευτικοί που υλοποιούν την ταξινόμηση οφείλουν να γνωρίζουν το πλαίσιο διδασκαλίας του μαθήματος δηλαδή να διδάσκουν οι ίδιοι τους που αξιολογούν ώστε μπορούν να κατηγοριοποιούν τις δραστηριότητες στα κατάλληλα επίπεδα της ταξινομίας και να μην υπάρχουν παρανοήσεις ή διαφορετικές ερμηνείες των επιπέδων.

2.4 Ταξινομία SOLO

Η δομή των παρατηρούμενων μαθησιακών αποτελεσμάτων, δηλαδή η γνωστική ταξινομία SOLO, η οποία αναπτύχθηκε από τους Biggs & Collis (1982), κατηγοριοποιεί την νοητική δραστηριότητα μέσω των ποιοτικών και ποσοτικών χαρακτηριστικών των δραστηριοτήτων που απαιτείται να υλοποιήσουν οι μαθητές.

Σύμφωνα με τους Hattie & Brown (2004), η SOLO περιγράφει τις διαδικασίες που σχετίζονται με την ερώτηση και την απάντηση σε αυτή σε μια κλίμακα αυξανόμενης δυσκολίας ή πολυπλοκότητας. Η πιο γνωστή ταξινόμηση στην εκπαίδευση είναι η ταξινομία Bloom η οποία αναφέρεται στον τύπο σκέψης ή επεξεργασίας που απαιτείται για την ολοκλήρωση εργασιών ή τις απαντήσεις σε ερωτήσεις. Τα προβλήματα που εντοπίστηκαν στην ταξινομία Bloom αντιμετωπίζονται καλύτερα με τη χρήση της ταξινομίας SOLO. Όταν χρησιμοποιείται η ταξινομία SOLO, οι ερωτήσεις ή οι δραστηριότητες θα δημιουργούνται με διαφορετικό τρόπο και ο εκπαιδευτικός θα επικεντρωθεί στην ταξινόμηση μόνο των απαντήσεων.

Η ταξινομία αυτή αναπτύχθηκε με την ανάλυση της δομής των απαντήσεων των εκπαιδευόμενων σε δραστηριότητες ως απάντηση σε δεδομένες πληροφορίες ή γνώσεις και τον προσδιορισμό του τύπου σκέψης που παρουσιάζεται με εκτεταμένες γραπτές απαντήσεις. Η ταξινομία συμβάλλει στη μάθηση, στη διδασκαλία ή στην αξιολόγηση. Αποτελεί και βασίζεται στην ανάλυση των δομικών χαρακτηριστικών των ερωτήσεων και των απαντήσεων. Προσδιορίζει χαρακτηριστικά αύξησης της ποσότητας και της ποιότητας της σκέψης.

Μπορεί να βοηθήσει τους εκπαιδευτικούς να εφαρμόσουν τις κατάλληλες εκπαιδευτικές διαδικασίες, αξιολογώντας την ποιότητα της μάθησης που λαμβάνει χώρα στην τάξη και να επιφέρουν κατάλληλες τροποποιήσεις στη διδασκαλία τους, όποτε χρειάζεται. Η ισχύς της ταξινομίας SOLO βασίζεται στο να αξιολογηθεί η ποιότητα και η ποσότητα της μάθησης, τα οποία είναι χαρακτηριστικά που απουσιάζουν από πολλές διαδικασίες αξιολόγησης της μάθησης των μαθητών. Οι Biggs και Collis (1982) ωστόσο, παρέχουν πολλά παραδείγματα για το πώς οι εκπαιδευτικοί μπορούν να χρησιμοποιήσουν την ταξινομία όχι μόνο για να αξιολογήσουν τη γνώση του περιεχομένου, αλλά και για να βοηθήσουν τους μαθητές να διερευνήσουν το πλήρες φάσμα των δεξιοτήτων που απαιτούνται σε μια συγκεκριμένη εργασία (Hattie & Brown, 2004)

Η ποιοτική αξιολόγηση των απαντήσεων των μαθητών μπορεί να ταξινομηθεί σε κατηγορίες - επίπεδα, τα οποία εκφράζουν την εξελικτική πορεία της διαδικασίας οικοδόμησης της γνώσης. Κάθε απάντηση των μαθητών για να αξιολογηθεί και να ταξινομηθεί σε ένα επίπεδο διερευνήθηκε και αναλύθηκε ως προς τις ακόλουθες τρεις συνιστώσες (Μπέλλου, 2003):

- a) Τον εντοπισμό και την αναφορά των στοιχείων σε μια περιγραφή, και των παραγόντων σε μια αιτιολόγηση, που διαδραματίζουν ενεργό ρόλο στο υπό μελέτη φαινόμενο.
- b) Τους συσχετισμούς των δεδομένων (στοιχείων ή παραγόντων).

c) Την εξαγωγή τεκμηριωμένου συγκεράσματος, σύμφωνα με τα προηγούμενα.

Η ταξινόμια SOLO περιλαμβάνει 5 επίπεδα:

1. Πρώτο επίπεδο προδομικό. Ο μαθητής:

- είτε αποφεύγει την ερώτηση (άρνηση, απόκρουση)
- είτε μεταφέρει την ερώτηση καταφατικά (ταυτολογία)
- είτε κάνει έναν άσχετο συνειρμό βασισμένο σε προσωπικά δεδομένα
- δεν εξετάζει διάφορους παράγοντες και έννοιες που έχουν σχέση με το θέμα και δεν προβαίνει σε κανένα συσχετισμό τους
- χρησιμοποιεί λίγες από τις διαθέσιμες πληροφορίες και σπάνια καταλήγει σε συμπέρασμα.

2. Δεύτερο επίπεδο, μονοδομικό. Ο μαθητής:

- επιλέγει ένα από τα σχετικά στοιχεία της παρουσίασης και επικεντρώνεται σ' αυτό
- ανταποκρίνεται με περιορισμένο τρόπο, σπάνια συνδέει τμήματα πληροφοριών και δεν δίνει εξηγήσεις
- δεν καταλήγει σε συμπέρασμα ή το συμπέρασμα προκύπτει βιαστικά από λίγα στοιχεία.

3. Τρίτο επίπεδο πολυδομικό. Ο μαθητής:

- επιλέγει δύο ή περισσότερα στοιχεία από την παρουσίαση και τα παραθέτει, αναφέροντάς τα απλώς με μία σειρά, αγνοώντας τις σχέσεις τους
- δε χρησιμοποιεί όλες τις διαθέσιμες πληροφορίες και δεν επιχειρεί να κάνει συσχετισμούς
- συνήθως δεν καταλήγει σε συμπέρασμα ή οδηγείται σε εναλλακτικό συμπέρασμα ή ακόμα μπορεί να αναφέρει το αναμενόμενο συμπέρασμα αποσπασματικά, χωρίς αυτό να προκύπτει από τα δεδομένα και τη λογική που παρουσίασε πριν

4. Τέταρτο επίπεδο συσχετιστικό. Ο μαθητής:

- χρησιμοποιεί τις περισσότερες ή όλες τις διαθέσιμες πληροφορίες, τις συνδέει με συνεπή τρόπο και τις ενσωματώνει σε ένα εννοιολογικό σχήμα, με το οποίο αντιμετωπίζει και τις αντικρουόμενες καταστάσεις
- αρχίζει να δίνει εξηγήσεις συνδυάζοντας τα στοιχεία και αναζητώντας σχέσεις αιτίου – αποτελέσματος. Έτσι παράγει ένα επιχείρημα και δεν αρκείται απλώς σε μια παράθεση σχετικών στοιχείων
- η διαδικασία της επαγωγής οδηγεί σε ένα επιστημονικά αποδεκτό συμπέρασμα. Ως εκ τούτου δεν αναφέρεται καμιά εναλλακτική λύση

5. Πέμπτο επίπεδο εκτεταμένης θεώρησης. Ο μαθητής:

- χρησιμοποιεί πληροφορίες που δεν περιλαμβάνονται στα δεδομένα, γενικευμένες επιστημονικές αρχές που δείχνουν ότι το παράδειγμα είναι μια μόνο πιθανή περίπτωση ενός φαινομένου μεγάλης εμβέλειας, που επεκτείνεται σε άλλες περιπτώσεις
- κάνει νέες υποθέσεις που βασίζονται σε προηγούμενα λογικά αιτιολογημένα συμπεράσματα
- δεν επιδιώκει να προσδιορίσει ένα ορισμένο ή περιορισμένο συμπέρασμα, αλλά δοκιμάζει, υποθέτει και κρίνει άλλες πιθανές απαντήσεις που θα ταίριαζαν στο ερώτημα.

Ο Biggs (1999) παρείχε ένα παράδειγμα εφαρμογής της ταξινομίας κυρίως στην τριτοβάθμια εκπαίδευση σε δραστηριότητες προγραμματισμού στο πλαίσιο της ανάπτυξης λογισμικού οπτικού προγραμματισμού. Το πρόγραμμα έπρεπε να είναι συντακτικά σωστό και να εκτελεί ένα σύνολο από απαιτούμενες λειτουργίες. Αν και τα κριτήρια βαθμολόγησης περιελάμβαναν τη δυνατότητα χρήσης υποπρογραμμάτων αυτό δεν οδήγησε τους φοιτητές να ενσωματώσουν υποπρογράμματα ή να μειώσουν την αλληλοεπικάλυψη υποπρογραμμάτων. Τα υποπρογράμματα έτειναν να είναι μεγάλα και να εκτελούν πολλαπλές εργασίες. Οι φοιτητές έκαναν συχνά λάθη και έγραφαν ελλιπή προγράμματα για όλες τις λειτουργίες που δίνονταν για βαθμολόγηση. Ο προγραμματισμός αποτελούσε μια άσκηση για την ολοκλήρωση όσο το δυνατόν περισσότερων λειτουργιών χωρίς να ενδιαφέρει η τελική ορθότητα ή η ποιότητα του προγράμματος.

2.5 Εφαρμογή της ταξινομίας SOLO

Πλήθος δραστηριοτήτων χρησιμοποιεί την ταξινομία SOLO στην αξιολόγηση απαντήσεων σε ερωτήσεις προγραμματισμού. Από τις έρευνες διαπιστώνεται ότι οι εκπαιδευτικοί επικεντρώνονται σε πτυχές του προγραμματισμού που συνδέονται με τα κατώτερα επίπεδα της ταξινομίας SOLO (Lister, Simon, Thompson, Whalley, & Prasad, 2006; Thompson, Whalley, Lister, & Simon, 2006; Whalley et al., 2006).

Στην έρευνα του Thompson (2007) δίνονται σε φοιτητές ένα σύνολο από κριτήρια, βασισμένα στην ταξινομία SOLO, για να επηρεάσουν την προσέγγισή δραστηριοτήτων προγραμματισμού και να αξιολογήσουν τις εργασίες τους. Οι φοιτητές δεν ήταν εξοικειωμένοι με αυτή την προσέγγιση αξιολόγησης. Ενώ γνώριζαν τα κριτήρια, δυσκολεύονταν να κατανοήσουν ή να αποδεχθούν τις προσδοκίες των κριτηρίων. Παρότι χρησιμοποιήθηκαν σε δραστηριότητες προγραμματισμού που αφορούσαν την εφαρμογή ενός προγράμματος υπάρχει η δυνατότητα να χρησιμοποιηθούν για αξιολόγηση απαντήσεων των μαθητών στην ανάγνωση και τη σύγκριση των τμημάτων κώδικα καθώς και στην αξιολόγηση των στρατηγικών που χρησιμοποιούνται για την επίλυση προβλημάτων ανάγνωσης κώδικα (Lister et al., 2006; Thompson et al., 2006; Whalley et al., 2006).

Η έρευνα των Sheard et al. (2008) βασίζεται στην προηγούμενη εργασία των Lister et al. (2006) και ασχολείται με ορισμένες κριτικές της έρευνας αυτής. Οι φοιτητές είναι σχετικά συνεπείς στο

επίπεδο SOLO για δραστηριότητες του ίδιου γνωστικού επιπέδου και οι απαντήσεις τους στις δραστηριότητες ανάγνωσης προγράμματος συσχετίζονται θετικά με τις αντίστοιχες στη δημιουργία προγράμματος. Η βαθμολογία εξετάσεων μειώνεται καθώς το επίπεδο της κατηγορίας SOLO μειώνεται, από το συσχετιστικό στο προδομικό. Η ταξινόμια SOLO δεν χρησιμοποιήθηκε για βαθμολογία στις εξετάσεις, αλλά εφαρμόστηκε κατόπιν, ανεξάρτητα από τη βαθμολογία των εξετάσεων. Διαπιστώθηκε συνοχή στα επίπεδα SOLO των απαντήσεων των φοιτητών σε ερωτήσεις πολλαπλής επιλογής προγραμματισμού, καθώς και συνέπεια μεταξύ δραστηριοτήτων ανάγνωσης και δημιουργίας προγράμματος (Sheard et al., 2008).

Η έρευνα των Philpott, Robbins, & Whalley (2007) είχε ως στόχο να βελτιώσει τη διδασκαλία του προγραμματισμού. Η έρευνα αποσκοπούσε στην διερεύνηση την ιεραρχία των δεξιοτήτων προγραμματισμού, διαπιστώθηκε ότι οι βασικές έννοιες προγραμματισμού αποτελούν το κατώτερο μέρος της ιεραρχίας. Επίσης παρατηρήθηκε ισχυρή συσχέτιση μεταξύ των δεξιοτήτων εκτέλεσης και δημιουργίας προγράμματος, ειδικά όταν η εκτέλεση περιελάμβανε επαναλήψεις. Στην έρευνα αναφέρονται αποτελέσματα που δείχνουν ότι η δεξιότητα εκτέλεσης προγράμματος είναι χαμηλότερη στην ιεραρχία από την δεξιότητα ερμηνείας προγράμματος (Philpott et al., 2007).

Ο Thompson (2008) εξηγεί ότι η ταξινόμια SOLO αντιπροσωπεύει τη δομή του παρατηρούμενου μαθησιακού αποτελέσματος και αποτελεί ποσοτικό και ποιοτικό μέτρο. Τα χαμηλότερα επίπεδα επικεντρώνονται στην ποσότητα της γνώσης που γνωρίζει ο φοιτητής ενώ τα υψηλότερα επίπεδα επικεντρώνονται στο βαθμό ολοκλήρωσης των δραστηριοτήτων που αναθέτονται στους φοιτητές (Lister, Clear, et al., 2009; Sheard et al., 2008; Whalley, Clear, Robbins, & Thompson, 2011). Με τη χρήση της ταξινόμιας οι απαντήσεις των φοιτητών ταξινομούνται όχι τόσο ως προς την ορθότητά, αλλά σύμφωνα με το βαθμό ολοκλήρωσης τους, εφόσον όσο πιο ολοκληρωμένη είναι μια απάντηση είναι τόσο πιο πειστική απόδειξη αποτελεί ότι ο φοιτητής έχει κατανοήσει το πρόγραμμα.

Οι Lister, Fidge, & Teague (2009) διαχώρισαν τους φοιτητές με βάση τις σωστές απαντήσεις τους σε ερωτήσεις εκτέλεσης, επεξήγησης, και γραφής προγράμματος, έτσι διεξήγαγαν ζεύγη συγκρίσεων των τριών κατηγοριών ερωτήσεων. Κατέληξαν στο συμπέρασμα ότι οι φοιτητές που μπορούν να εξηγήσουν τον κώδικα μπορούν γενικά να εκτελούν τον κώδικα ενώ οι φοιτητές που μπορούν να γράψουν κώδικα μπορούν και τα δύο, και να εκτελούν αλλά και να εξηγήσουν κώδικα. Η ανάλυση αυτή υποστηρίζει την έννοια της ιεραρχίας των δεξιοτήτων προγραμματισμού, με την εκτέλεση ως την πιο στοιχειώδη ικανότητα, την εξήγηση ως μια ενδιάμεση δεξιότητα και τη συγγραφή ως κορυφή της ιεραρχίας.

Η έρευνα των Lister, Clear, et al. (2009) περιλαμβάνει την αναπαραγωγή προηγούμενης ανάλυσης χρησιμοποιώντας μια πολύ ευρύτερη συλλογή δεδομένων, την τελειοποίηση της ταξινόμιας SOLO

σε ερωτήσεις ερμηνείας κώδικα, την επέκταση της ταξινόμησης στις ερωτήσεις συγγραφής προγράμματος, την επέκταση μερικών παλαιότερων μελετών σχετικά με το «χάσιμο χρόνου» των φοιτητών απαντώντας σε ερωτήσεις εξετάσεων και τη διερεύνηση μιας περαιτέρω θεωρητικής βάσης για εργασία που μέχρι τώρα ήταν πρωτίστως εμπειρική.

Οι Lister et al. (2009) εφάρμοσαν τις ταξινομίες Bloom και SOLO ως συστήματα ταξινόμησης για ερωτήσεις και απαντήσεις εξετάσεων αρχαρίων προγραμματιστών. Η έρευνα επεκτάθηκε από ερωτήσεις κατανόησης σε ερωτήσεις που αφορούσαν τη σύνταξη προγράμματος. Αυτές οι προσπάθειες υιοθέτησαν μια στρατηγική από πάνω προς τα κάτω για να κατατάξουν άμεσα κάθε απάντηση φοιτητή στα επίπεδα SOLO. Παρότι πέτυχαν μια ταξινόμηση για κάποια ερωτήματα και κατέληξαν σε συμφωνία μεταξύ τους οι ερευνητές, η διαδικασία φάνηκε πολύ συνδεδεμένη με το πλαίσιο και τα συγκεκριμένα ερωτήματα.

Εκφράστηκε η άποψη ότι αν η ταξινομία SOLO δεν μπορεί να εφαρμοστεί με συνέπεια από πολλούς εκπαιδευτικούς οι οποίοι είναι εκπαιδευμένοι στη χρήση της, η αξία της πρέπει να αμφισβητηθεί. Πράγματι, ακόμη και όταν έμπειροι κριτές ταξινομούσαν απαντήσεις παρά τις αρχικές διαφωνίες οι οποίες αποτελούσαν μειοψηφία, στο σύνολο η συμφωνία μεταξύ τους ήταν αρκετά υψηλή (Clear et al., 2008). Σε αντίθεση με την ανησυχία για τη συνεπή εφαρμογή των αξιολογήσεων SOLO, υπήρξε μεγάλη τάση για γενναιοδωρία στη βαθμολογία των φοιτητών με αποτέλεσμα να γίνουν με ασάφεια οι κατηγοριοποιήσεις. Οι κατηγορίες πρέπει να είναι απλές και κατανοητές. Προτάθηκε ως λύση η συγχώνευση των τριών χαμηλότερων επιπέδων της ταξινόμησης σε μία μόνο κατηγορία. Συνοπτικά, προτείνεται να εφαρμοστεί ξανά η ταξινομία SOLO σε ερωτήσεις ανάγνωσης. Γενικές παρατηρήσεις ανέφεραν υψηλότερα επίπεδα συναίνεσης σε αυτό το σύνολο κατηγοριών (Lister, Clear, et al., 2009).

Από την άλλη πλευρά, λίγες έρευνες αφορούν την κατηγοριοποίηση απαντήσεων των φοιτητών σε ερωτήσεις σχετικές με τη δημιουργία προγράμματος. Υποστηρίζεται ότι η ταξινομία SOLO στην περίπτωση αυτή αντιπροσωπεύει ιεραρχική κλίμακα, υποδηλώνοντας ότι οι ορισμοί που χρησιμοποιούνται για την κατηγοριοποίηση είναι αποδεκτοί. Η ίδια έρευνα καταλήγει σε αυτό το συμπέρασμα και για δραστηριότητες ανάγνωσης προγράμματος. Ως εκ τούτου πρόκειται για έναν πολλά υποσχόμενο τρόπο κατηγοριοποίησης των επιδόσεων σε δραστηριότητες που περιλαμβάνουν τη δημιουργία προγράμματος (Lister, Clear, et al., 2009).

Η έρευνα της Selby (2015) δεν επικεντρώνεται στο πόσο καλά ο μαθητής κατέκτησε μια δεξιότητα, αλλά στον εντοπισμό του υψηλότερου επιπέδου γνωστικής ικανότητας που απαιτείται για την δεξιότητα αυτή. Οι εκπαιδευτικές ταξινομίες Bloom και SOLO αποτελούν το πλαίσιο αυτής της έρευνας. Από τα συμπεράσματα διαφαίνεται ότι τα ανώτερα επίπεδα της ταξινομίας Bloom είναι

εφαρμόσιμα στον προγραμματισμό. Μπορεί να είναι δυνατή η ταξινόμηση διαφορετικών δραστηριοτήτων και γνωστικών διεργασιών σε μια ιεραρχία.

Στην έρευνα των Whalley, Clear, Robbins, & Thompson (2011) αναπτύχθηκε ένα πλαίσιο που περιελάμβανε τα σημαντικότερα στοιχεία, ή τμήματα προγραμμάτων, που χρησιμοποιούνται σε προβλήματα δημιουργίας προγράμματος. Διαπιστώθηκε ότι ο συνδυασμός του πλαισίου με την ταξινόμια SOLO βοήθησε να οριστούν οι κατηγορίες SOLO και παρείχε μια βελτιωμένη προσέγγιση στην εφαρμογή των αρχών της ταξινόμιας σε προβλήματα δημιουργίας προγράμματος. Σε προηγούμενες έρευνες, η κατάταξη των προγραμμάτων στην ταξινόμια του SOLO είχε αποδειχθεί δύσκολη (Clear et al., 2008, Lister et al., 2009), δεδομένου ότι η διαδικασία κατάταξης φαίνεται πολύ δεσμευτικό πλαίσιο. Ως εκ τούτου η επίτευξη συνεπών αξιολογήσεων είναι πρόκληση, ειδικά για προβλήματα γραφής προγράμματος.

Μια προσέγγιση υιοθετήθηκε για να επεξεργαστεί τις απαντήσεις μαθητών από κάτω προς τα πάνω μέσω μιας διεργασίας κωδικοποίησης και σχεδιασμού δύο επιπέδων. Με βάση το καθορισμένο πλαίσιο ποιότητας εντοπίστηκαν σημαντικά στοιχεία που μπορούν να συμβάλλουν σε πιο συνεπείς κατηγοριοποιήσεις SOLO στις απαντήσεις σε γραπτές ερωτήσεις. Αυτά τα στοιχεία αντιπροσωπεύουν αφαιρέσεις από τον ίδιο τον κώδικα, με βάση ποιοτικές κρίσεις, οι οποίες μπορούν να προσαρμοστούν ανάλογα δραστηριότητες γραφής κώδικα. Οι εκπαιδευτικοί πρέπει να είναι προσεκτικοί κατά την εφαρμογή της ταξινόμιας. Απαιτείται συνεπής και κατάλληλος σχεδιασμός για ερωτήσεις γραφής προγράμματος από τους εκπαιδευτικούς, βελτίωση της πρακτικής για τον αρχάριο και αύξηση στην ποιότητα της διδασκαλίας και της αξιολόγησης (Whalley, Clear, Robbins, & Thompson, 2011).

Η αξιοποίηση της ταξινόμιας SOLO για την αξιολόγηση των αλγοριθμικών χαρακτηριστικών του εισαγωγικού προγραμματισμού παρουσιάζεται από τους Ginat & Menashe, (2015). Δίνεται ένα πλαίσιο χρήσης της SOLO για την αξιολόγηση των βασικών χαρακτηριστικών αλγοριθμικού σχεδιασμού του γραπτού κώδικα. Η καινοτομία προέρχεται από την εστίαση στην επιλογή, στον ευέλικτο χειρισμό και στη σύνθεση των βασικών μοτίβων, καθώς και στην αξιοποίηση των χαρακτηριστικών των εργασιών. Καθορίζονται τα επίπεδα SOLO σε σχέση με τον αλγοριθμικό σχεδιασμό και τα συνήθη σφάλματα. Στόχος της εργασίας είναι η δημιουργία ενός πλαισίου για την αλγοριθμική εξέταση και αξιολόγηση των δραστηριοτήτων προγραμματισμού και των λύσεων συμπεριλαμβανομένων των δυσκολιών των μαθητών.

Οι εκπαιδευτικοί προσαρμόζουν την ταξινόμια καθώς θεωρούν τα τρία χαμηλότερα επίπεδα ως πιο ποσοτικά και τα δύο ανώτερα επίπεδα ως πιο ποιοτικά. Η εστίαση στη χρήση της ταξινόμιας SOLO αφορά την επιλογή και την εφαρμογή των γνωστών διαδικασιών όπως είναι ο υπολογισμός πλήθους και αθροίσματος σε μια επαναληπτική διαδικασία, η σειριακή αναζήτηση, ο υπολογισμός

μεγίστου-ελαχίστου, η χρήση πίνακα, οι μαθηματικοί υπολογισμοί και οι δομές επανάληψης. Η βασική αρχή είναι η επαναχρησιμοποίηση των διαδικασιών. Έτσι, κατά τη διδασκαλία του προγραμματισμού, είναι πολύ ωφέλιμο να διδάσκονται οι μαθητές πως να αναγνωρίζουν τα πρότυπα που σχετίζονται με τις δραστηριότητες, να τα χρησιμοποιούν και να συνθέτουν προγράμματα με ευελιξία. Το πλαίσιο αυτό προσφέρει ταξινομημένη εξέταση ποικίλων λύσεων και περιλαμβάνει αναφορά σε σφάλματα. Ωστόσο, τα σημαντικά σφάλματα, συνεπάγονται έλλειψη γνώσης εννοιών ή έλλειψη ικανής ολοκλήρωσης, αντικατοπτρίζουν μια λύση χαμηλότερου επιπέδου ενώ τα μη σημαντικά δεν επηρεάζουν το επίπεδο ταξινόμησης μιας λύσης (Ginat & Menashe, 2015).

Η επέκταση της ταξινομίας των Ginat & Menashe (2015) για τον σχεδιασμό αλγορίθμου από τους Izu, Weerasinghe, & Pore (2016) παρουσιάζει ένα πλαίσιο αξιολόγησης που χρησιμοποιεί τη ταξινομία SOLO για την αξιολόγηση της πολυπλοκότητας των ερωτημάτων προγραμματισμού. Το πλαίσιο επεκτείνει τη SOLO χρησιμοποιώντας μια προσαρμόσιμη παράμετρο που καθορίζει ρητά την ικανότητα του φοιτητή να γράφει ολοένα και πιο σύνθετα τμήματα προγράμματος. Η ανάλυση των απαντήσεων από εξετάσεις επιτρέπει να ποσοτικοποιηθεί η διαδικασία και να μελετηθούν οι συσχετίσεις μεταξύ της γνώσης των δεξιοτήτων προγραμματισμού και της συνολικής επίδοσης των φοιτητών. Επιπλέον, εντοπίζονται συνήθη σφάλματα που καταδεικνύουν τις δυσκολίες που αντιμετωπίζουν οι φοιτητές όταν προσπαθούν να συνδυάσουν δομές προγραμματισμού με μη συνηθισμένους τρόπους όπως τη διάσπαση του προβλήματος με συνδυασμό δομών επανάληψης, συνθηκών και εντολών εκχώρησης, την επαναχρησιμοποίηση και επέκταση των διαδικασιών προγραμματισμού και των προτύπων που έχουν μάθει και έχουν εξασκηθεί.

Στην έρευνα των Izu et al. (2016) παραλείπεται η υψηλότερη κατηγορία (εκτεταμένη θεώρηση) για δύο λόγους. Πρώτος, επειδή δεν είναι δυνατό να προσδιοριστεί από τις απαντήσεις στις εξετάσεις εάν μπορούν να επεκτείνουν τις τρέχουσες γνώσεις τους σε άλλα θέματα ή πεδία. Δεύτερος, επειδή στα περισσότερα μαθήματα προγραμματισμού είναι δύσκολο το περιεχόμενο, δεν υπάρχει περιθώριο για αξιολογήσεις σε ερωτήσεις στο επίπεδο αυτό. Η ταξινομία SOLO μπορεί να εφαρμοστεί σε ερωτήσεις και σε απαντήσεις εξετάσεων. Αν μια ερώτηση είναι γραμμένη σε δεδομένο επίπεδο, μπορεί να απαντηθεί μόνο σε αυτό το επίπεδο ή πιο κάτω, παρόλο που τις περισσότερες φορές μια απάντηση χαμηλότερου επιπέδου είναι ελλιπής ή λανθασμένη.

Η ταξινομία SOLO αποτελεί ένα μέσο ταξινόμησης των μαθησιακών αποτελεσμάτων με βάση την πολυπλοκότητά τους, επιτρέποντάς να αξιολογηθεί το έργο των φοιτητών από την άποψη της ποιότητάς του. Οι Izu et al. (2016) συμφωνούν με την αντίληψη των Petersen et al. (2011) ότι κάθε ταξινόμηση των ερωτήσεων εξαρτάται σε μεγάλο βαθμό από το περιεχόμενο του μαθήματος και το βάθος της κάλυψής του. Στην έρευνα τους χωρίζουν το δείγμα σε δύο ομάδες: στην πρώτη οι

απαντήσεις έχουν το ίδιο επίπεδο SOLO (οι δύο ερωτήσεις έχουν το ίδιο επίπεδο ή μία ερώτηση ήταν κενή) και στην δεύτερη με δύο διαφορετικές βαθμολογίες SOLO. Όπως αναμενόταν, οι βαθμολογίες των εξετάσεων μειώνονται καθώς το επίπεδο στην ταξινόμια SOLO μειώνεται, από το συσχετιστικό στο προδομικό. Η περιγραφή των επιπέδων SOLO σε συνδυασμό με τον κατάλογο των δομικών στοιχείων για το μάθημα καθιστούν ένα καλό σημείο αναφοράς για τους εκπαιδευτικούς ώστε να καθορίσουν τόσο την αθροιστική όσο και τη διαμορφωτική αξιολόγηση των φοιτητών στο σωστό επίπεδο δυσκολίας.

Με άλλα λόγια, οι φοιτητές που απάντησαν σε προδομικό επίπεδο χρειάζονται ενισχυτική στήριξη. Οι φοιτητές παρουσίασαν ένα ευρύ φάσμα επιδόσεων, η συνολική απόδοση της τάξης ήταν παρόμοια με άλλων δημοσιευμένων μελετών. Όπως ήταν αναμενόμενο, οι αρχάριοι έκαναν συντακτικά και λογικά σφάλματα. Στο μονοδομικό επίπεδο, οι σπουδαστές έδειξαν μια κακή αντίληψη της σημασιολογίας του βρόχου, τόσο για απλούς όσο και για εμφωλευμένους βρόχους. Στο πολυδομικό επίπεδο, οι σπουδαστές απέτυχαν να εξετάσουν προσεκτικά την ενσωμάτωση τμημάτων βρόχου και την πρόσβαση στη δομή δεδομένων (Izu et al., 2016).

Οι Mühlhling et al. (2015) εξετάζουν τη δομική γνώση των αρχαρίων σε περιβάλλον οπτικού προγραμματισμού. Προτείνονται επίπεδα δυσκολίας για τις δύο διαστάσεις (τα αντικείμενα και την αλγοριθμική δομή) βάσει της ταξινόμιας SOLO. Φυσικά, η ιεραρχία αυτών των επιπέδων πρέπει να επικυρωθεί εμπειρικά. Επιπλέον, για κάποιες άλλες έννοιες προγραμματισμού το αντίστοιχο επίπεδο φαίνεται αμφισβητήσιμο. Από την έρευνα φαίνεται ότι ο ορισμός και η μέτρηση των δεξιοτήτων προγραμματισμού είναι ένα πολύ περίπλοκο και δύσκολο έργο, που απαιτεί πολλές διαφορετικές δραστηριότητες.

Ενδιαφέρον παρουσιάζει η αξιολόγηση που σχεδιάστηκε για να μετρήσει τις δεξιότητες υπολογιστικής σκέψης μαθητών πρωτοβάθμιας εκπαίδευσης, με χρήση της ταξινόμιας SOLO, σε περιβάλλον οπτικού προγραμματισμού. Οι απαντήσεις αξιολογήθηκαν ως προς την ορθότητα αλλά και ως προς επίπεδο SOLO. Η βαθμολογία ως προς την ταξινόμια SOLO αντικατοπτρίζει το βαθμό κατανόησης της δομής του προβλήματος. Οι μαθητές με βαθμολογίες ανάγνωσης και κατανόησης μαθηματικών κάτω από τις ελάχιστες βασικές απαιτήσεις, δεν ήταν σε θέση να δείξουν βασική κατανόηση των δραστηριοτήτων προγραμματισμού. Τα αποτελέσματα, επίσης, επιβεβαιώνουν προηγούμενη μελέτη που συνδέει την ικανότητα υπολογισμού με την επίδοση των μαθητών, με την πλειονότητα των μαθητών να μην επιχειρούν ή να μην είναι σε θέση να δείξουν καμιά βασική κατανόηση των δραστηριοτήτων αξιολόγησης (Seiter, 2015).

Οι Shi, Min, & Zhang (2017) εφαρμόζουν οπτικοποίηση των ρόλων των μεταβλητών στη διδασκαλία προγραμματισμού. Για τους σκοπούς της έρευνας το επίπεδο SOLO βαθμολογήθηκε σύμφωνα με τις κατηγορίες SOLO για την κατασκευή προγράμματος. Οι αναλύσεις δεδομένων έδειξαν

σημαντική βελτίωση του επιπέδου SOLO της κατασκευής του προγράμματος και μεγαλύτερη αποδοχή για τους ρόλους των μεταβλητών. Συμπερασματικά, η απεικόνιση των ρόλων των μεταβλητών μπορεί να δώσει στους αρχάριους ένα νέο εννοιολογικό πλαίσιο που να τους επιτρέπει να σχεδιάσουν συσχετιστικό πρόγραμμα από μια ολιστική άποψη και τους βοηθά να μάθουν την έννοια των ρόλων των μεταβλητών. Ωστόσο, διαπιστώθηκε βελτίωση στο επίπεδο SOLO των φοιτητών, και όχι στις βαθμολογίες τους για την κατασκευή του προγράμματος. Θα πρέπει να υιοθετηθούν πρόσθετες μέθοδοι διδασκαλίας για τη βελτίωση της ικανότητας των μαθητών όσον αφορά την κατασκευή του προγράμματος.

Σε ορισμένες έρευνες συνδυάζεται η ταξινομία Bloom με την ταξινομία SOLO. Η έρευνα των Petersen, Craig, & Zingaro (2011) εξέτασε τις ταξινομίες Bloom και SOLO για την αξιολόγηση ερωτήσεων εξετάσεων από πολλά εκπαιδευτικά ιδρύματα. Διαπιστώθηκε ότι στις δύο ταξινομίες το επίπεδο εξαρτάται από συγκεκριμένα παραδείγματα και ασκήσεις που καλύπτονται κατά τη διάρκεια της διδακτικής πράξης. Για παράδειγμα, εάν η απάντηση ή το πρόγραμμα έχει παρουσιαστεί ή συζητηθεί στην τάξη, το επίπεδο στην ταξινομία του Bloom θα είναι γνώση ενώ το επίπεδο στην ταξινομία του SOLO θα είναι μονοδομικό. Διαφορετικά ο φοιτητής πρέπει να δημιουργήσει τη λύση είτε από την αρχή είτε επεκτείνοντας ένα άλλο απλούστερο μοτίβο με το οποίο είναι ήδη εξοικειωμένος, τότε το επίπεδο των ερωτήσεων θα μεταβληθεί στην ταξινομία Bloom σε σύνθεση ενώ στην ταξινομία SOLO σε πολυδομικό ή συσχετιστικό.

Επίσης διερευνάται ο συνδυασμός των ταξινομιών Bloom και SOLO στον Nondeterminism προγραμματισμό. Η έρευνα επικεντρώνεται στις κατηγορίες της αναθεωρημένης ταξινόμιας Bloom, δημιουργώ και εφαρμόζω. Οι κύριοι λόγοι είναι ότι η ερμηνεία αυτών των κατηγοριών στο πλαίσιο του προγραμματισμού συνδέεται ευθέως με τις δραστηριότητες στις οποίες επικεντρώθηκε το μάθημα, την κατανόηση και τη δημιουργία προγραμμάτων. Η απομάκρυνση των σφαλμάτων, η οποία σχετίζεται με την ανάλυση και την αξιολόγηση στην αναθεωρημένη ταξινόμηση του Bloom, τονίστηκε λιγότερο. Επιπλέον επειδή οι κατηγορίες Bloom μπορούν να αλληλεπικαλύπτονται, η ταξινόμηση μπορεί να είναι διφορούμενη. Έτσι, η έρευνα εστίασε σε μία ενδιάμεση κατηγορία (εφαρμόζω) και μία υψηλότερη κατηγορία (δημιουργώ). Στη συνδυασμένη ταξινομία δεν δίνεται υπεροχή σε καμία από τις ταξινομίες. Σύμφωνα με τα ευρήματά όλοι οι φοιτητές κατέληξαν σε επαρκές επίπεδο κατανόησης στις κατώτερες έως ενδιάμεσες κατηγορίες και όσοι δημιούργησαν εργασίες σε επίπεδο που εμπίπτει στις ανώτερες κατηγορίες έφθασαν επίσης σε επαρκές επίπεδο (Alexandron, Armoni, Gordon, & Harel, 2016).

Οι Meerbaum-Salant et al. (2010) θεωρούν ότι η ερμηνεία της αναθεωρημένης ταξινόμησης Bloom για δραστηριότητες προγραμματισμού δεν είναι απλή, εφόσον ακόμα και εμπειρογνώμονες δυσκολεύονται να συμφωνήσουν σε μια ερμηνεία. Επίσης υποστηρίζουν ότι η γραφή ενός

προγράμματος μπορεί να ανήκει σε διαφορετικά γνωστικά επίπεδα όπως κατανόω, εφαρμόζω ή δημιουργώ και, ομοίως, η εκτέλεση προγράμματος μπορεί να ανήκει στα επίπεδα θυμάμαι, κατανόω ή αναλύω. Θεωρήθηκε ότι η κατηγοριοποίηση δραστηριοτήτων του ίδιου είδους σε διαφορετικές κατηγορίες εξαιτίας διαφορών μεγέθους αποτελεί σωστή λύση και ότι το ζήτημα του μεγέθους (και κατά συνέπεια της πολυπλοκότητας) πρέπει να αντιμετωπιστεί ρητά. Για τους παραπάνω λόγους δημιούργησαν μια ταξινόμηση η οποία είναι συνδυασμός των ταξινομιών Bloom και SOLO.

Η νέα ταξινομία περιλαμβάνει τρεις υπερ-κατηγορίες, μονοδομική, πολυδομική και συσχετιστική καθεμία από τις οποίες περιέχει τρεις υποκατηγορίες, που αντιστοιχούν στην κατανόηση, την εφαρμογή και τη δημιουργία. Τελικά κατέληξαν σε μια ταξινόμηση εννέα επιπέδων με υψηλότερο το επίπεδο συσχετιστικής δημιουργίας και χαμηλότερο το επίπεδο μονοδομικής κατανόησης. Η συνδυασμένη ταξινόμηση χρησιμοποιήθηκε για σχεδιασμό εργαλείων καθώς και για ανάλυση. Από τα αποτελέσματα της έρευνας αναιρείται η παραδοσιακή άποψη ότι η κατανόηση κώδικα που σχετίζεται με την κατηγορία εφαρμογής είναι ευκολότερη από τη δημιουργία κώδικα. Οι αφηρημένες έννοιες όπως η αρχικοποίηση, οι μεταβλητές κ.α. απαιτούν πολυδομική και ενδεχομένως συσχετιστική ικανότητα (Meerbaum-Salant et al., 2010).

Για την καταλληλότητα της ταξινομίας SOLO ως εργαλείο αξιολόγησης γνώσεων προγραμματισμού είναι ανάγκη να αξιοποιήσουμε τα συμπεράσματα της βιβλιογραφικής επισκόπησης. Αρχικά επισημαίνουμε ότι η εφαρμογή της ταξινομίας SOLO καθώς και της ταξινομίας Bloom παρουσιάζει δυσκολίες στην διδακτική πράξη. Και στις δύο ταξινομίες η ταξινόμηση των δραστηριοτήτων σε επίπεδα εξαρτάται από συγκεκριμένα παραδείγματα και ασκήσεις που διδάσκονται στην τάξη. Η διαδικασία κατάταξης των δραστηριοτήτων σε επίπεδα είναι άμεσα συνδεδεμένη με το πλαίσιο του μαθήματος και τις συγκεκριμένες δραστηριότητες. Μέσα από τη βιβλιογραφική επισκόπηση διαπιστώθηκε ότι η χρήση της ταξινομίας SOLO εξαρτάται σε μεγάλο βαθμό από το περιεχόμενο του μαθήματος και το βάθος κάλυψης της διδακτέας ύλης.

Σε πολλές έρευνες εκφράζεται η άποψη ότι αν η ταξινομία SOLO δεν μπορεί να εφαρμοστεί με συνέπεια από πολλούς εκπαιδευτικούς οι οποίοι είναι εκπαιδευμένοι στη χρήση της, η αξία της πρέπει να αμφισβητηθεί. Αυτό δημιουργεί ερωτηματικά και κρίνεται αναγκαίο οι εκπαιδευτικοί να είναι προσεκτικοί κατά την εφαρμογή της ταξινομίας.

Ένα άλλο συμπέρασμα που εντοπίστηκε στις έρευνες και αξίζει ιδιαίτερη προσοχή είναι το γεγονός ότι η ταξινομία SOLO δεν εφαρμόζεται αυτούσια όπως καθορίστηκε από τον δημιουργό της. Οι ερευνητές προσαρμόζουν την ταξινομία σε πολλές έρευνες. Συνήθως παραλείπουν κάποια επίπεδα (προδρομικό, Εκτεταμένης θεώρησης) ή δημιουργούν επιμέρους κατηγορίες σε κάποια επίπεδα, ή συνενώνουν τα χαμηλότερα επίπεδα σε ένα επίπεδο. Σε ορισμένες περιπτώσεις δημιουργούν μια

νέα ταξινομία συνδυάζοντας την ταξινομία SOLO με την ταξινομία Bloom. Επομένως η εφαρμογή της ταξινομίας SOLO στην αξιολόγηση γνώσεων προγραμματισμού δημιουργεί ασάφειες και αμφισβήτηση για τους παραπάνω λόγους.

3. Το εκπαιδευτικό πλαίσιο του Προγραμματισμού στο Γενικό Λύκειο στην Ελλάδα

Η διδασκαλία του προγραμματισμού και της αλγοριθμικής στο Γενικό Λύκειο αποτελεί βασική συνιστώσα του Ελληνικού Προγράμματος Σπουδών (Π.Σ) για τη διδασκαλία της Επιστήμης των Υπολογιστών. Στη Γ λυκείου διδάσκεται το μάθημα «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον» στη Ομάδα Θετικών Σπουδών και στην Ομάδα Οικονομίας και Πληροφορικής δύο (2) ώρες τη βδομάδα. Εισήχθηκε στο Γενικό Λύκειο από το σχολικό έτος 1999-2000 ως μάθημα υποχρεωτικό στην Τεχνολογική Κατεύθυνση.

Ο γενικός σκοπός του μαθήματος είναι η ανάπτυξη δεξιοτήτων και ικανοτήτων σχετικών με την αλγοριθμική και την ορθολογική χρήση τους στην καθημερινή ζωή. Πολλές βασικές έννοιες αλγοριθμικής, αλλά και προγραμματισμού, συνιστούν αναπόσπαστο τμήμα των γενικών γνώσεων και δεξιοτήτων που πρέπει να αποκτήσει ο μαθητής, οι οποίες στην πλειονότητά τους δεν προσεγγίζονται από άλλα μαθήματα (Βακάλη et al., 1998).

Το μάθημα Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον δεν έχει ως στόχο τη διδασκαλία και την εκμάθηση κάποιου συγκεκριμένου προγραμματιστικού περιβάλλοντος ούτε αποσκοπεί στη λεπτομερειακή εξέταση της δομής, και των συντακτικών κανόνων κάποιας γλώσσας προγραμματισμού. Ως εργαλείο δόμησης της σκέψης εστιάζει στις προσεγγίσεις και στις τεχνικές επίλυσης προβλημάτων. Έχει σαν γενικό σκοπό οι μαθητές:

- ✓ να καλλιεργήσουν αναλυτική σκέψη και συνθετική ικανότητα,
- ✓ να αναπτύξουν τη δημιουργικότητα, τη φαντασία στο σχεδιασμό,
- ✓ να αναπτύξουν ικανότητες μεθοδολογικού χαρακτήρα και δεξιότητες αλγοριθμικής προσέγγισης,
- ✓ να καταστούν ικανοί να υλοποιούν τις λύσεις απλών προβλημάτων με χρήση βασικών προγραμματιστικών γνώσεων

Μια εκτεταμένη βιβλιογραφική επισκόπηση στο διεθνή καθώς και στον ελληνικό χώρο έδειξε ότι υπάρχουν προτάσεις για την ανάπτυξη της γνώσης αλλά και αποτελέσματα από εμπειρικές μελέτες στον προγραμματισμό, παρόλα αυτά το πλαίσιο αξιολόγησης της γνώσης και των μαθησιακών αποτελεσμάτων, δεν είναι καθορισμένο με σαφή κριτήρια και συνέπεια ως προς τις θεωρητικές προσεγγίσεις και τα συμπεράσματα των μελετών.

Αυτός ο παράγοντας είναι ιδιαίτερα σημαντικός στον προγραμματισμό αφού συνδυάζει αφενός την αλγοριθμική σκέψη και αφετέρου την γνώση μιας γλώσσας προγραμματισμού. Ο σωστός και ολοκληρωμένος συνδυασμός τους μπορεί να οδηγήσει στην επίλυση ενός προβλήματος με προγραμματισμό (Μπέλλου & Μικρόπουλος, 2008).

B. Εμπειρική μελέτη

4. Μεθοδολογία

4.1 Ερευνητικά ερωτήματα

Για τους εκπαιδευτικούς που διδάσκουν προγραμματισμό είναι θέμα ιδιαίτερης σημασίας να μπορούν να παρακολουθούν την πρόοδο των μαθητών τους κατά τη διδασκαλία του μαθήματος. Η βιβλιογραφική επισκόπηση για την αξιολόγηση των γνώσεων προγραμματισμού εστιάζει στην τριτοβάθμια εκπαίδευση και ελάχιστες έρευνες αφορούν άλλες βαθμίδες εκπαίδευσης. Επίσης διαπιστώνεται αμφισβήτηση για τη χρήση των ταξινομιών Bloom και SOLO στην αξιολόγηση γνώσεων προγραμματισμού.

Λαμβάνοντας υπόψη όλα τα παραπάνω η παρούσα μελέτη διερευνά:

1. το επίπεδο γνώσεων στον προγραμματισμό τελειόφοιτων μαθητών Λυκείου,
2. την εφαρμογή του μοντέλου της Ιεραρχικής Αξιολόγησης Προγραμματισμού

4.2 Το πλαίσιο της έρευνας

Στην παρούσα έρευνα εφαρμόστηκε το μοντέλο της Ιεραρχικής Αξιολόγησης Προγραμματισμού (ΙΑΠ) που προτείνουν οι Μπέλλου & Μικρόπουλος (2008) το οποίο αποτελεί ένα μοντέλο για την ποιοτική αξιολόγηση των γνώσεων στον προγραμματισμό. Βασίζεται στη λογική της ταξινομίας SOLO καθώς υιοθετεί ορισμένες από τις αρχές της και προτείνει πέντε ιεραρχικά επίπεδα αξιολόγησης. Στο μοντέλο αυτό οι γνώσεις που αφορούν τον προγραμματισμό διακρίνονται σε δύο κατηγορίες. Η πρώτη κατηγορία περιλαμβάνει τον συνδυασμό συντακτικής – γραμματικής και εννοιολογικής γνώσης σχετικά με τη γλώσσα προγραμματισμού ενώ η δεύτερη κατηγορία περιλαμβάνει τον συνδυασμό εννοιολογικής και στρατηγικής γνώσης του προγραμματισμού, δηλαδή την αλγοριθμική σκέψη.

Η ανάλυση κάθε μιας από τις δύο κατηγορίες της προγραμματιστικής γνώσης σε ιεραρχικά επίπεδα παρουσιάζεται στον πίνακα 4.1:

Πίνακας 4.1: Ιεραρχικά επίπεδα γνώσεων προγραμματισμού

Γνώση στη γλώσσα προγραμματισμού	Αλγοριθμική σκέψη
1. Λανθασμένη χρήση εντολών.	1. Κατανόηση του προβλήματος εν μέρει.
2. Ορθή χρήση απλών εντολών, όπως εντολές εισόδου – εξόδου, εντολές εκχώρησης, δηλώσεις σταθερών και μεταβλητών	2. Κατανόηση του προβλήματος, αλλά ελλιπής αλγόριθμος.
3. Ορθή χρήση λογικών δομών, όπως δομές επιλογής και δομές επανάληψης, κατανόηση λειτουργίας των εντολών, των μεταβλητών και της μνήμης.	3. Σχεδιασμός της λύσης σε γενικό πλαίσιο, χωρίς να λαμβάνονται υπόψη οριακές περιπτώσεις και περιορισμοί.
4. Δεξιότητες προηγούμενου επιπέδου και επιλογή κατάλληλων δομών επανάληψης, συναρτήσεων, διαδικασιών.	4. Οργάνωση και απόδοση της λύσης του προβλήματος με πλήρη και ικανοποιητικά δομημένη προσέγγιση.
5. Βαθιά γνώση και φιλοσοφία της γλώσσας προγραμματισμού.	5. Επιλογή και υλοποίηση του βέλτιστου αλγόριθμου.

Ο συνδυασμός των παραπάνω περιπτώσεων των ιεραρχικών επιπέδων που περιλαμβάνονται στις δύο κατηγορίες γνώσεων προγραμματισμού οδήγησε τους Μπέλλου & Μικρόπουλο (2008) να δημιουργήσουν πέντε ιεραρχικά επίπεδα μαθησιακών αποτελεσμάτων, τα οποία συγκροτούν το μοντέλο της «Ιεραρχικής Αξιολόγησης Προγραμματισμού».

Τα επίπεδα αυτά είναι:

1ο επίπεδο, προδομικό

Ο μαθητής δεν έχει κατανοήσει το πρόβλημα, ούτε γνωρίζει τη σύνταξη των εντολών. Εναλλακτικά, δεν απαντά. Ουσιαστικά δεν έχει δομήσει ακόμα επιστημονικά αποδεκτή γνώση σχετική με το ζητούμενο θέμα.

2ο επίπεδο, επιμέρους κατανόησης

Ο μαθητής έχει αντιληφθεί εν μέρει το πρόβλημα και έχει αποδώσει με ορθό τρόπο, σύμφωνα με τη γραμματική και το συντακτικό της χρησιμοποιούμενης γλώσσας, απλές εντολές (όπως εισόδου – εξόδου, εκχώρησης και δήλωσης μεταβλητών) που απαιτούνται για τη διατύπωση της λύσης σε μορφή προγράμματος.

3ο επίπεδο, προσεγγιστικής κατανόησης

Ο μαθητής έχει αντιληφθεί σε γενικές γραμμές το πρόβλημα και έχει αποδώσει μια αποδεκτή λύση με ορθό τρόπο (γραμματικό – συντακτικό), τόσο όσον αφορά τις απλές εντολές, όσο και τις λογικές δομές επανάληψης και επιλογής, που απαιτεί το πρόγραμμα. Φαίνεται ότι γνωρίζει τον τρόπο λειτουργίας της κάθε εντολής καθώς και τη λειτουργία των μεταβλητών.

4ο επίπεδο, συνδυαστικό

Ο μαθητής έχει κατανοήσει ακόμα και τα επιμέρους προβλήματα και έχει σχεδιάσει ικανοποιητικά δομημένη λύση του όλου προβλήματος. Χρησιμοποιεί εντολές και δομές, όπως στο προηγούμενο επίπεδο, αλλά επιπλέον επιλέγει την κατάλληλη δομή επανάληψης και τις κατάλληλες συναρτήσεις και διαδικασίες. Συνδυάζει με επιτυχία και τις δύο συνιστώσες του προγραμματισμού, την αλγοριθμική σκέψη και τη γνώση της γλώσσας.

5ο επίπεδο, εκτεταμένης θεώρησης

Ο μαθητής έχει επιλέξει να υλοποιήσει το βέλτιστο αλγόριθμο για κάθε περίπτωση και γνωρίζει τη φιλοσοφία της συγκεκριμένης γλώσσας προγραμματισμού που πιθανά ο ίδιος έχει επιλέξει να χρησιμοποιήσει.

4.3 Δείγμα

Η έρευνα διεξήχθη σε ένα Γενικό Λύκειο κατά το σχολικό έτος 2017-18 στο πλαίσιο του μαθήματος «Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον» της Γ' Λυκείου. Το Πρόγραμμα Σπουδών υποδεικνύει τη χρήση ενός περιβάλλοντος ψευδοκώδικα, με το όνομα ΓΛΩΣΣΑ για την διδασκαλία των βασικών αλγοριθμικών και προγραμματικών δομών (τύποι δεδομένων, μεταβλητές, δομές ελέγχου και επανάληψης, πίνακες, διαδικασίες και συναρτήσεις). Οι μαθητές διερευνούν προβλήματα και αναπτύσσουν και αναλύουν τις λύσεις τους στο περιβάλλον ΓΛΩΣΣΑ. Στην έρευνα συμμετείχαν 33 μαθητές ομάδας προσανατολισμού Θετικών Σπουδών και Οικονομίας – Πληροφορικής. Όλοι οι μαθητές διδάχθηκαν το αντίστοιχο μάθημα περίπου 12 βδομάδες. Η γραπτή εξέταση πραγματοποιήθηκε στο τέλος του πρώτου τετραμήνου και είχε διάρκεια τρεις διδακτικές ώρες. Επίσης όλοι οι μαθητές συναίνεσαν να χρησιμοποιηθούν τα γραπτά τους στην παρούσα έρευνα.

Τα θέματα της εξέτασης περιελάμβαναν 10 δραστηριότητες με υποερωτήματα, ταξινομημένες στα γνωστικά επίπεδα της αναθεωρημένης ταξινομίας Bloom. Όλα τα επίπεδα περιλαμβάνουν δύο δραστηριότητες εκτός από τα επίπεδα θυμάμαι και δημιουργώ, τα οποία περιλαμβάνουν μία δραστηριότητα. Αυτό έγινε επειδή το πρώτο επίπεδο θεωρείται απλό ενώ το δεύτερο ιδιαίτερα δύσκολο για τους μαθητές. Ο πίνακας 4.2 δείχνει αναλυτικά τις απαντήσεις των μαθητών ανά επίπεδο και δραστηριότητα.

Πίνακας 4.2: Απαντήσεις μαθητών ανά επίπεδο και δραστηριότητα

Σύνολα	Επίπεδο Θυμάμαι	Επίπεδο Κατανούώ		Επίπεδο Εφαρμόζω		Επίπεδο Αναλύω		Επίπεδο Αξιολογώ		Επίπεδο Δημιουργώ
		Δρ. 1η	Δρ. 2η	Δρ. 1η	Δρ. 2η	Δρ. 1η	Δρ. 2η	Δρ. 1η	Δρ. 2η	
		Απαντήσεις	33	33	32	32	32	33	29	
Δεν απάντησαν	0	0	1	1	1	0	4	0	4	5

Αρχικά οι απαντήσεις των μαθητών αξιολογήθηκαν με το μέθοδο Ιεραρχικής Αξιολόγησης γνώσεων Προγραμματισμού ενώ παράλληλα δόθηκαν για αξιολόγηση με τον παραδοσιακό τρόπο βαθμολόγησης σε έξι ανεξάρτητους βαθμολογητές, με εμπειρία από 12 έως 18 έτη στη διδασκαλία και την αξιολόγηση προγραμματισμού. Ο καθένας από αυτούς βαθμολόγησε τις απαντήσεις γραπτών πέντε ή έξι μαθητών. Η βαθμολογία δόθηκε σε 20βάθμια κλίμακα είτε για κάθε ερώτημα της δραστηριότητας είτε για ολόκληρη τη δραστηριότητα. Τα αποτελέσματα από τη βαθμολόγηση αυτή παρουσιάζονται στο παράρτημα μετά το τέλος της έρευνας. Στη συνέχεια οι βαθμολογίες των μαθητών κατηγοριοποιήθηκαν στα επίπεδα της ΙΑΠ ώστε να μπορούν να συγκριθούν οι δύο τρόποι αξιολόγησης και να εξαχθούν συμπεράσματα. Για το σκοπό αυτό χρησιμοποιήθηκε η παρακάτω κλίμακα αντιστοίχισης:

- 1^ο επίπεδο ΙΑΠ περιλαμβάνει βαθμολογίες από 0 έως και 7
- 2^ο επίπεδο ΙΑΠ περιλαμβάνει βαθμολογίες από 8 έως και 11
- 3^ο επίπεδο ΙΑΠ περιλαμβάνει βαθμολογίες από 12 έως και 16
- 4^ο επίπεδο ΙΑΠ περιλαμβάνει βαθμολογίες από 17 έως και 19
- 5^ο επίπεδο ΙΑΠ περιλαμβάνει τον βαθμό 20

4.4 Δραστηριότητες αξιολόγησης και ανάλυση δεδομένων

Οι δραστηριότητες που επιλέγει ο εκπαιδευτικός να χρησιμοποιήσει κατά τη διδασκαλία του έχουν σημαντικό αντίκτυπο στην κατανόηση του προγράμματος. Ως εκ τούτου η επιλογή των κατάλληλων δραστηριοτήτων έχει ιδιαίτερη σημασία. Σε πολλές έρευνες εφαρμόζεται η οργάνωση των μαθησιακών δραστηριοτήτων σύμφωνα με το αυξανόμενο επίπεδο δυσκολίας της ταξινόμιας Bloom κατά τη διδασκαλία. Επίσης οι τελικές εξετάσεις στα εισαγωγικά μαθήματα προγραμματισμού ακολουθούν την ίδια προσέγγιση (Gomes & Mendes, 2009; Saroni et al., 2016).

Η ταξινόμια Bloom ασχολείται με την αναγνώριση του υψηλότερου επιπέδου γνωστικής ζήτησης που απαιτείται από μια δραστηριότητα ενώ η ταξινόμια SOLO είναι ένα μοντέλο ικανότητας που απεικονίζει το βαθμό στον οποίο κατακτιέται η κατανόηση. Ο συνδυασμός αναγνωρίζει την πολυπλοκότητα που συνδέεται με την απόκτηση αλλά και την απόδειξη της κατανόησης (Meerbaum-Salant et al., 2010).

Οι δραστηριότητες που δόθηκαν στους μαθητές για να αξιολογηθούν οι γνώσεις τους στον προγραμματισμό είναι οι παρακάτω:

Επίπεδο Θυμάμαι

- a. Ποιους τύπους δεδομένων υποστηρίζει το πρόγραμμα; Δώστε ένα παράδειγμα εντολής εκχώρησης για κάθε τύπο δεδομένων.
- b. Να γράψετε τη σύνταξη για κάθε δομή επανάληψης που χρησιμοποιείται στο πρόγραμμα.
- c. Ποια γεωμετρικά σχήματα χρησιμοποιούνται στο διάγραμμα ροής και σε τι χρησιμεύει το καθένα.

Επίπεδο Κατανόω

Η πρώτη δραστηριότητα (ΔΡ1):

$X \leftarrow 1$

$A \leftarrow 10$

ΟΣΟ $X < A$ ΕΠΑΝΑΛΑΒΕ
ΓΡΑΨΕ X

$X \leftarrow X + 3$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Δίνεται το διπλανό τμήμα προγράμματος σε ΓΛΩΣΣΑ.

α) Ποιοι αριθμοί εμφανίζονται κατά την εκτέλεση του προγράμματος.

β) Ποια είναι η μικρότερη τιμή που μπορεί να πάρει η μεταβλητή A ώστε να εμφανίζονται πάλι οι ίδιοι αριθμοί.

γ) Αντικαταστήστε την ΟΣΟ ... ΕΠΑΝΑΛΑΒΕ με άλλη δομή επανάληψης ώστε το πρόγραμμα να δίνει τα ίδια αποτελέσματα.

Η δεύτερη δραστηριότητα (ΔΡ2):

Δίνεται το παρακάτω τμήμα προγράμματος

$S \leftarrow 0$

$I \leftarrow 1$

ΟΣΟ $I \leq 10$ ΕΠΑΝΑΛΑΒΕ

ΔΙΑΒΑΣΕ α

$S \leftarrow S + \alpha$

$I \leftarrow I + 1$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ S

α) Αν δοθεί σε κάθε επανάληψη ως είσοδος το 10 τι θα εμφανιστεί στην οθόνη του υπολογιστή; Με ποια συνθήκη θα εμφανιστεί το 10 μετά το τέλος επανάληψης; β) Αν η συνθήκη γίνει $I \leq 0$ τότε τι θα εμφανιστεί στην οθόνη του υπολογιστή γ) Τι υπολογίζει το παραπάνω πρόγραμμα;

Επίπεδο Εφαρμοζώ

Η πρώτη δραστηριότητα (ΔΡ1):

Δίνεται το παρακάτω πρόβλημα «Να γραφεί πρόγραμμα που θα διαβάζει τα ονόματα και τους χρόνους που έκαναν 20 αθλητές στο αγώνισμα των 400 μέτρων και να εμφανίζει το όνομα του πρωταθλητή». Συμπληρώστε τα κενά στο παρακάτω πρόγραμμα ώστε να αποτελεί λύση του παραπάνω προβλήματος.

ΠΡΟΓΡΑΜΜΑ Α2

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΜΙΝ

ΧΑΡΑΚΤΗΡΕΣ : Ο, ΟΜΙΝ

ΑΡΧΗ

ΜΙΝ ← 9999999999

ΟΜΙΝ ← ' '

(α) -----

ΔΙΑΒΑΣΕ Χ,Ο

ΑΝ Χ < ΜΙΝ ΤΟΤΕ

(β) -----

(γ) -----

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ΟΜΙΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Η δεύτερη δραστηριότητα (ΔΡ2):

Δίνεται το παρακάτω πρόβλημα «Να γραφεί πρόγραμμα που θα διαβάσει τα ποσά που ξόδεψε κάποιος για τις εξόδους του σε ένα μήνα (θετικοί αριθμοί) και θα υπολογίζει και θα εμφανίζει το άθροισμα και το πλήθος τους». Συμπληρώστε τα κενά στο παρακάτω πρόγραμμα ώστε να αποτελεί λύση του παραπάνω προβλήματος.

ΠΡΟΓΡΑΜΜΑ Α3

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ : Χ, Σ

ΑΚΕΡΑΙΕΣ : ΠΛ

ΑΡΧΗ

ΠΛ ← 0

(α) -----

ΔΙΑΒΑΣΕ Χ

(β) -----

Σ ← Σ + Χ

(γ) -----

ΔΙΑΒΑΣΕ Χ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ Σ, ΠΛ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Επίπεδο Αναλύω

Η πρώτη δραστηριότητα (ΔΡ1):

Να γραφεί πρόγραμμα που θα διαβάσει θετικούς ακέραιους αριθμούς και θα εμφανίζει το τελευταίο ψηφίο τους, το πλήθος των άρτιων και το ποσοστό των περιττών.

Η δεύτερη δραστηριότητα (ΔΡ2):

Μια εταιρεία δημοσκοπήσεων θέτει κάποιο ερώτημα σε ένα δείγμα 2000 πολιτών. Για την επεξεργασία των δεδομένων να δημιουργήσετε πρόγραμμα που θα κάνει τα εξής: Διαβάζει το φύλο του πολίτη ελέγχοντας ότι είναι Άνδρας ή Γυναίκα (Α=Άνδρας, Γ=Γυναίκα) να διαβάζει την απάντηση που έδωσε κάθε πολίτης, η οποία μπορεί να είναι «ΝΑΙ», «ΟΧΙ», «ΔΕΝ ΞΕΡΩ» ελέγχοντας ώστε η απάντηση να παίρνει αποκλειστικά μόνο αυτές τις τιμές. Κατόπιν να υπολογίζει και να εμφανίζει το πλήθος των ατόμων που απάντησαν «ΝΑΙ» καθώς και το ποσοστό των γυναικών και των ανδρών που απάντησαν «ΟΧΙ».

Επίπεδο Αξιολογώ

Η πρώτη δραστηριότητα (ΔΡ1):

Δίνεται το πρόγραμμα:

ΠΡΟΓΡΑΜΜΑ Α1

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΑΤ

ΑΡΧΗ

ΔΙΑΒΑΣΕ Χ

ΟΣΟ Χ<>0 ΕΠΑΝΑΛΑΒΕ

ΑΝ Χ>0 ΤΟΤΕ

 ΑΤ ← Χ

ΑΛΛΙΩΣ

 ΑΤ ← Χ*(-1)

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ΑΤ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

α)Τι υπολογίζει η μεταβλητή ΑΤ;

β)Εντοπίστε και διορθώστε το λάθος στο παραπάνω πρόγραμμα

γ)πως μπορεί να τροποποιηθεί το παραπάνω πρόγραμμα ώστε με λιγότερες εντολές να δίνει το ίδιο αποτέλεσμα;

Η δεύτερη δραστηριότητα (ΔΡ2):

Δίνεται το πρόγραμμα:

ΠΡΟΓΡΑΜΜΑ Α2

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ :B, M

ΑΚΕΡΑΙΕΣ : K

ΑΡΧΗ

K ← 1

ΟΣΟ K<=10 ΕΠΑΝΑΛΑΒΕ

ΔΙΑΒΑΣΕ B

ΑΝ B>M ΤΟΤΕ

 M ← B

ΤΕΛΟΣ_ΑΝ

K ← K +1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ M

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

α)Τι υπολογίζει η μεταβλητή M;

β)Εντοπίστε και διορθώστε το λάθος στο παραπάνω πρόγραμμα

γ)πως μπορεί να τροποποιηθεί το παραπάνω πρόγραμμα ώστε με λιγότερες εντολές να δίνει το ίδιο αποτέλεσμα;

Επίπεδο Δημιουργώ

Σε ένα κινητό τηλέφωνο ο χρήστης πρέπει να εισάγει το PIN στην αρχή της λειτουργίας του. Το κινητό ανοίγει όταν δοθεί το σωστό PIN. Ο χρήστης μπορεί να κάνει το πολύ 3 προσπάθειες. Μετά από τις 3 αποτυχημένες προσπάθειες το τηλέφωνο κλειδώνει. Να γραφεί πρόγραμμα που θα διαβάζει το αποθηκευμένο PIN του κινητού. Επίσης θα διαβάζει τους αριθμούς που θα εισάγει ο χρήστης. Θα εμφανίζει το μήνυμα «Καλώς ήρθατε» όταν το κινητό ανοίγει ή το μήνυμα «κλείδωσε» σε περίπτωση αποτυχίας. Επίσης θα εμφανίζει τον αριθμό προσπαθειών εισαγωγής PIN στο κινητό.

Ανάλυση Δεδομένων

Αρχικά περιγράφεται η ανάλυση δεδομένων της ταξινόμησης των απαντήσεων των μαθητών στα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού.

Ανάλυση δεδομένων στο επίπεδο Θυμάμαι

Η δραστηριότητα που δόθηκε στους μαθητές ήταν:

- Ποιους τύπους δεδομένων υποστηρίζει το πρόγραμμα; Δώστε ένα παράδειγμα εντολής εκχώρησης για κάθε τύπο δεδομένων.
- Να γράψετε τη σύνταξη για κάθε δομή επανάληψης που χρησιμοποιείται στο πρόγραμμα.
- Ποια γεωμετρικά σχήματα χρησιμοποιούνται στο διάγραμμα ροής και σε τι χρησιμεύει το καθένα.

Για το α ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- Όταν η απάντηση του μαθητή αναφέρει τους 4 τύπους μεταβλητών γράφοντας τους αντίστοιχους ορισμούς και για κάθε τύπο δίνεται μια εντολή εκχώρησης τότε η απάντηση τοποθετείται στο 5ο επίπεδο (εκτεταμένης θεώρησης).
- Όταν η απάντηση του μαθητή αναφέρει τους 4 τύπους μεταβλητών και για κάθε τύπο δίνεται μια εντολή εκχώρησης τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- Όταν η απάντηση του μαθητή αναφέρει τους 4 τύπους μεταβλητών και για κάθε τύπο δίνεται ένα παράδειγμα τιμής της κάθε μεταβλητής ή λείπει κάποιος τύπος η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- Όταν η απάντηση του μαθητή αναφέρει τους 4 τύπους μεταβλητών και για κάθε τύπο δίνεται ένα παράδειγμα τιμής της κάθε μεταβλητής όμως να υπάρχει λάθος (τύπος ή παράδειγμα) η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης). Επίσης στο ίδιο επίπεδο κατατάσσονται απαντήσεις που αναφέρουν μόνο τους 4 τύπους μεταβλητών.
- Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

5 ^ο επίπεδο	Τύποι	Εντολές εκχώρησης
	ΑΚΕΡΑΙΟΣ	B ← 10
	ΠΡΑΓΜΑΤΙΚΟΣ	X ← -36.5
	ΛΟΓΙΚΟΣ	K ← ΨΕΥΔΗΣ
	ΧΑΡΑΚΤΗΡΑΣ	A ← 'Άννα'

Ακέραιος τύπος: περιλαμβάνει τους ακέραιους αριθμούς που είναι γνωστοί από τα μαθηματικά.

Πραγματικός τύπος: περιλαμβάνει τους πραγματικούς αριθμούς που είναι γνωστοί από τα μαθηματικά.

Χαρακτήρας: αναφέρεται τόσο σε ένα χαρακτήρα όσο και σε ένα σύνολο χαρακτήρων και βρίσκονται υποχρεωτικά μέσα σε απλά εισαγωγικά (').

Λογικός: δέχεται μόνο 2 τιμές, ΑΛΗΘΗΣ και ΨΕΥΔΗΣ.

4 ^ο επίπεδο	Τύποι	Εντολές εκχώρησης
	ΑΚΕΡΑΙΟΣ	X←2
	ΠΡΑΓΜΑΤΙΚΟΣ	A←3.2
	ΛΟΓΙΚΟΣ	K←ΑΛΗΘΗΣ
	ΧΑΡΑΚΤΗΡΑΣ	ON←'καλημέρα'
3 ^ο επίπεδο	ΠΡΑΓΜΑΤΙΚΟΣ:1.2	ΠΡΑΓΜΑΤΙΚΟΣ: 2/3
	ΑΚΕΡΑΙΟΣ: 8 ή	ΑΚΕΡΑΙΟΣ: 1
	ΛΟΓΙΚΟΣ: ΑΛΗΘΗΣ	ΧΑΡΑΚΤΗΡΑΣ: όνομα
	ΧΑΡΑΚΤΗΡΑΣ: "ΜΑΡΙΑ"	

2^ο επίπεδο – 1^ο επίπεδο παραλείπονται οι ενδεικτικές απαντήσεις επειδή είναι ευνόητες.

Για το β ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού (ΙΑΠ) ορίζονται ως εξής:

- i. Όταν στην απάντηση του ο μαθητής έχει γράψει τη σύνταξη των τριών δομών επανάληψης στη γενική μορφή και τη λειτουργία κάθε δομής τότε η απάντηση τοποθετείται στο 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν στην απάντηση του ο μαθητής έχει γράψει τη σύνταξη των τριών δομών επανάληψης στη γενική μορφή ή με δικό του παράδειγμα τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν στην απάντηση του ο μαθητής έχει γράψει τη σύνταξη των τριών δομών επανάληψης αλλά υπάρχουν λάθη στη σύνταξη ή δεν έχει τη σύνταξη από μία δομή τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν στην απάντηση του ο μαθητής έχει γράψει τη σύνταξη μιας μόνο δομής σωστά ή λείπει εντελώς η συνθήκη τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

5ο επίπεδο

Η ΓΛΩΣΣΑ υποστηρίζει τρεις εντολές επανάληψης, την εντολή ΟΣΟ όπου η επανάληψη ελέγχεται από μία λογική έκφραση στην αρχή και εκτελείται συνεχώς όσο η συνθήκη είναι Αληθής, την εντολή ΜΕΧΡΙΣ_ΟΤΟΥ όπου η συνθήκη βρίσκεται στο τέλος του βρόχου και εκτελείται συνεχώς μέχρις ότου η συνθήκη αυτή γίνει Αληθής και τέλος την εντολή ΓΙΑ, με την οποία ο βρόχος επαναλαμβάνεται για προκαθορισμένο αριθμό φορών.

Η γενική μορφή σύνταξης της επανάληψης «ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ...» είναι η εξής:

ΓΙΑ Μεταβλητή ΑΠΟ Τιμή1 ΜΕΧΡΙ Τιμή2 ΜΕ_ΒΗΜΑ Τιμή3

Ομάδα εντολών

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Λειτουργία:

Η εντολή αυτή χειρίζεται μια μεταβλητή, στην οποία αρχικά εκχωρείται η αρχική τιμή. Η τιμή της μεταβλητής συγκρίνεται με την τελική τιμή και εφόσον είναι μικρότερη (ή μεγαλύτερη) από αυτή (θετικό βήμα ή αρνητικό βήμα), τότε εκτελούνται οι εντολές που βρίσκονται στο βρόχο (ανάμεσα στις εντολές ΓΙΑ και ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ). Στη συνέχεια η μεταβλητή ελέγχου αυξάνεται κατά την τιμή που ορίζει το ΒΗΜΑ. Αν η νέα τιμή είναι μικρότερη (ή μεγαλύτερη) της τελικής, τότε ο βρόχος εκτελείται ξανά. Η διαδικασία αυτή επαναλαμβάνεται συνεχώς, έως ότου η τιμή ελέγχου γίνει μεγαλύτερη (ή μικρότερη) της τελικής τιμής, οπότε τερματίζεται η επανάληψη και το πρόγραμμα συνεχίζει με την εντολή που ακολουθεί το ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ.

Η γενική μορφή σύνταξης της επανάληψης «ΟΣΟ... ΕΠΑΝΑΛΑΒΕ» είναι η εξής:

ΟΣΟ συνθήκη ΕΠΑΝΑΛΑΒΕ

ΟΜΑΔΑ ΕΝΤΟΛΩΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Λειτουργία:

Σε αυτή, η συνθήκη που ελέγχει την επανάληψη βρίσκεται στην αρχή της επανάληψης και ο βρόχος επαναλαμβάνεται συνεχώς, όσο η συνθήκη αυτή ισχύει. Όταν η συνθήκη γίνει Ψευδής τότε σταματάει η επανάληψη και εκτελείται η εντολή μετά από το ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ. Χαρακτηριστικό της επανάληψης αυτής είναι ότι ο αριθμός των επαναλήψεων δεν είναι γνωστός, ούτε μπορεί να υπολογιστεί πριν από την εκτέλεση του προγράμματος.

Η γενική μορφή σύνταξης της επανάληψης «ΜΕΧΡΙΣ_ΟΤΟΥ ...» είναι η εξής:

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΟΜΑΔΑ ΕΝΤΟΛΩΝ

ΜΕΧΡΙΣ_ΟΤΟΥ Συνθήκη

Λειτουργία:

Σε αυτή, η συνθήκη που ελέγχει την επανάληψη βρίσκεται στο τέλος της επανάληψης και ο βρόχος επαναλαμβάνεται συνεχώς, όσο η συνθήκη αυτή δεν ισχύει. Όταν η συνθήκη γίνει Αληθής τότε σταματάει η επανάληψη και εκτελείται η εντολή μετά από την ΜΕΧΡΙΣ_ΟΤΟΥ.

4ο επίπεδο

Η γενική μορφή σύνταξης της επανάληψης «ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ...» είναι η εξής:

ΓΙΑ Μεταβλητή ΑΠΟ Τιμή1 ΜΕΧΡΙ Τιμή2 ΜΕ_ΒΗΜΑ Τιμή3

Ομάδα εντολών

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Η γενική μορφή σύνταξης της επανάληψης «ΟΣΟ... ΕΠΑΝΑΛΑΒΕ» είναι η εξής:

ΟΣΟ συνθήκη ΕΠΑΝΑΛΑΒΕ

ΟΜΑΔΑ ΕΝΤΟΛΩΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Η γενική μορφή σύνταξης της επανάληψης «ΜΕΧΡΙΣ_ΟΤΟΥ ...» είναι η εξής:

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΟΜΑΔΑ ΕΝΤΟΛΩΝ

ΜΕΧΡΙΣ_ΟΤΟΥ Συνθήκη

3ο επίπεδο

ΓΙΑ ΑΠΟ «.....» ΜΕΧΡΙ «.....» ΜΕ_ΒΗΜΑ α

εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ Ι ΑΠΟ ----- ΜΕΧΡΙ ----- ΜΕ_ΒΗΜΑ ΕΠΑΝΑΛΑΒΕ

εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΟΣΟ «συνθήκη»

εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΟΣΟ Χ>0 ΤΟΤΕ ΕΠΑΝΑΛΑΒΕ

Ή

εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

Ή

εντολές

ΜΕΧΡΙΣ_ΟΤΟΥ Συνθήκη

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

2ο επίπεδο

ΓΙΑ Μ ΑΠΟ Τιμή1 ΜΕΧΡΙ Τιμή2 ΜΕ_ΒΗΜΑ χ

ΟΣΟ συνθήκη1 ΕΠΑΝΑΛΑΒΕ

ΜΕΧΡΙΣ_ΟΤΟΥ συνθήκη

1ο επίπεδο

ΓΙΑ, ΑΠΉ, ΜΕΧΡΙ, ΟΣΟ

ΓΙΑ...ΑΠΟ ...ΜΕΧΡΙ, ΟΣΟ, ΜΕΧΡΙΣ_ΟΤΟΥ

ΓΙΑ συνθήκη ΕΠΑΝΑΛΑΒΕ, ΕΠΑΝΕΛΑΒΕ ΜΕΧΡΙΣ_ΟΤΟΥ

Για το γ ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού (ΙΑΠ) ορίζονται ως εξής:

- i. Όταν στην απάντηση του ο μαθητής έχει γράψει όλα τα σχήματα και εξηγεί ποια είναι λειτουργία κάθε σχήματος τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- ii. Όταν στην απάντηση του ο μαθητής δεν έχει γράψει όλα τα σχήματα ή δεν δίνει εξήγηση για όλα τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iii. Όταν στην απάντηση του ο μαθητής έχει γράψει κάποιο σχήμα λάθος ή δίνει μια εξήγηση λάθος ή γράφει σωστά μόνο τα σχήματα τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- iv. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).
- v. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

Το διάγραμμα ροής αποτελείται από ένα σύνολο γεωμετρικών σχημάτων, όπου το καθένα δηλώνει μία συγκεκριμένη ενέργεια ή λειτουργία. Τα σχήματα ενώνονται μεταξύ τους με βέλη που δηλώνουν την σειρά εκτέλεσης των ενεργειών αυτών. Τα κυριότερα σχήματα είναι τα εξής:

Έλλειψη: Η αρχή και το τέλος του αλγορίθμου.

Ρόμβος: Μία συνθήκη με δύο εξόδους ανάλογα με την τιμή της (αληθής – ψευδής).

Πλάγιο παρ/μο: Η είσοδος και η έξοδος στοιχείων του αλγορίθμου.

Ή εναλλακτικά ο μαθητής σχεδιάζει το κάθε σχήμα και εξηγεί την λειτουργία του.

3ο επίπεδο

Έλλειψη: Η αρχή και το τέλος του αλγορίθμου. Ρόμβος, Πλάγιο παρ/μο

Ορθογώνιο παρ/μο: Η εκτέλεση μίας ή περισσότερων πράξεων.

2ο επίπεδο

Πλάγιο παρ/μο: χρησιμοποιείται για την εντολή εκχώρησης

πλάγιο ορθογώνιο τρίγωνο αντί για πλάγιο παρ/μο

κύκλος αντί για έλλειψη

πλάγιο παρ/μο, ρόμβος, έλλειψη, ορθογώνιο παρ/μο.

1ο επίπεδο

Παραλείπονται οι ενδεικτικές απαντήσεις επειδή είναι ευνόητες

Ανάλυση δεδομένων στο επίπεδο Κατανόω

Η πρώτη δραστηριότητα (ΔΡ1) που δόθηκε στους μαθητές ήταν:

X ← 1

A ← 10

ΟΣΟ X < A ΕΠΑΝΑΛΑΒΕ

ΓΡΑΨΕ X

X ← X + 3

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Δίνεται το διπλανό τμήμα προγράμματος σε ΓΛΩΣΣΑ.

α) Ποιοι αριθμοί εμφανίζονται κατά την εκτέλεση του προγράμματος.

β) Ποια είναι η μικρότερη τιμή που μπορεί να πάρει η μεταβλητή A ώστε να εμφανίζονται πάλι οι ίδιοι αριθμοί.

γ) Αντικαταστήστε την ΟΣΟ ... ΕΠΑΝΑΛΑΒΕ με άλλη δομή επανάληψης ώστε το πρόγραμμα να δίνει τα ίδια αποτελέσματα.

Για το α ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης)
- ii. Όταν η απάντηση του μαθητή αναφέρει ότι εμφανίζονται μόνο οι τιμές 1,4,7 τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό)
- iii. Όταν η απάντηση του μαθητή αναφέρει ότι εμφανίζονται οι τιμές 1,4,7, 10 ή οι τιμές 4,7,10 τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης)
- iv. Όταν ο μαθητής υπολογίζει κάθε τιμή του Χ με βήμα το 1 τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

Η σωστή απάντηση παρουσιάζεται στον παρακάτω πίνακα 4.3:

Πίνακας 4.3: Πίνακας τιμών ΔΡ1 (α ερώτημα)

Χ	1	4	7	10
Α	10			
Έξοδος	1	4	7	

Ή εναλλακτικά οι τιμές:1,4,7

3ο επίπεδο

Οι μαθητές έδωσαν ως απάντηση τις τιμές 1,4,7,10. Δηλαδή θεωρούν ότι η τιμή με την οποία τερματίζει η επανάληψη εμφανίζεται και αυτή. Αυτό θα μπορούσε να συμβεί στην περίπτωση που η συνθήκη περιελάμβανε το ίσον (=).

Επίσης μαθητές έδωσαν ως απάντηση τις τιμές 4,7,10. Αυτό θα μπορούσε να συμβεί μόνο στην περίπτωση που η εντολή ΓΡΑΨΕ Χ ήταν μετά την εντολή εκχώρησης $X \leftarrow X + 3$

2ο επίπεδο

Πίνακας 4.4: Πίνακας τιμών ΔΡ1 (α ερώτημα)

X	1	2	3	4	5	6	7	8	9	10
A	10	10	10	10	10	10	10	10	10	10
Έξοδος	1	2	3	4	5	6	7	8	9	10

Δεν λαμβάνει ο μαθητής υπόψη του ότι το βήμα της επανάληψης είναι το 3 αλλά θεωρεί ότι είναι το 1 και έτσι κάνει επιπλέον επαναλήψεις με αποτέλεσμα να εμφανίσει 10 τιμές της μεταβλητής X.

1ο επίπεδο

Δόθηκε ως απάντηση στο ερώτημα τυχαίες τιμές ή ονόματα μεταβλητών όπως:

X,A ή 1 ή 1,10,3 ή 10,3,A,1

Για το β ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης)
- Όταν η απάντηση του μαθητή αναφέρει ότι είναι η τιμή 8 τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό)
- Όταν η απάντηση του μαθητή αναφέρει ότι είναι η τιμή 10 τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης)
- Όταν η απάντηση του μαθητή αναφέρει έναν αριθμό με τον οποίο εκτελεί λιγότερες ή περισσότερες επαναλήψεις τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- Όταν ο μαθητής δεν απαντά ή δίνει άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

Οι μαθητές έδωσαν ως απάντηση την σωστή τιμή δηλαδή το 8

3ο επίπεδο

Οι μαθητές έδωσαν ως απάντηση την τιμή 10 ή 9.9999 ή 9

2ο επίπεδο, παράδειγμα οι τιμές 7, 4

1ο επίπεδο

Οι μαθητές έδωσαν ως απάντηση μια άσχετη τιμή ή δεν απάντησαν όπως: A←1, X

Για το γ ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Όταν στην απάντηση ο μαθητής χρησιμοποιεί τη δομή επανάληψης Για ...από ...μέχρι με τελική τιμή το 7 τότε η απάντηση τοποθετείται στο 5ο επίπεδο (εκτεταμένης θεώρησης)
- ii. Όταν στην απάντηση ο μαθητής χρησιμοποιεί τη δομή επανάληψης Για ...από ...μέχρι με τελική τιμή το 10 ή το 9 ή το 8 τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό). Καθώς επίσης όταν αντικαθιστά την ΟΣΟ με την Μέχρις_ότου με σωστή συνθήκη.
- iii. Όταν στην απάντηση ο μαθητής χρησιμοποιεί τη δομή επανάληψης Για ...από ...μέχρι με λανθασμένη τελική τιμή ή την Μέχρις_ότου με λάθος συνθήκη τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης)
- iv. Όταν στην απάντηση ο μαθητής αντικαθιστά τη δομή επανάληψης με δομή επιλογής Αν τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

5ο επίπεδο

ΓΙΑ Χ ΑΠΟ 1 ΜΕΧΡΙ 7 ΜΕ_ΒΗΜΑ 3
ΓΡΑΨΕ Χ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

4ο επίπεδο

ΓΙΑ Χ ΑΠΟ 1 ΜΕΧΡΙ 10 ΜΕ_ΒΗΜΑ 3
ΓΡΑΨΕ Χ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

$X \leftarrow 1$

$A \leftarrow 10$

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ Χ

$X \leftarrow X + 3$

ΜΕΧΡΙΣ_ΟΤΟΥ $X \geq A$

3ο επίπεδο

ΓΙΑ Χ ΑΠΟ 1 ΜΕΧΡΙ 4 ΜΕ_ΒΗΜΑ 3
ΓΡΑΨΕ Χ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

2ο επίπεδο

$X \leftarrow 1$

$A \leftarrow 10$

ΑΝ $X \geq A$ ΤΟΤΕ

ΓΡΑΨΕ Χ

$X \leftarrow X + 3$

ΤΕΛΟΣ_ΑΝ

1ο επίπεδο

ΜΕΧΡΙΣ_ΟΤΟΥ ...ΕΠΑΝΑΛΑΒΕ, ΓΙΑ $X < A$

ΕΠΑΝΑΛΑΒΕ

Η δεύτερη δραστηριότητα (ΔΡ2) που δόθηκε στους μαθητές ήταν:

Δίνεται το παρακάτω τμήμα προγράμματος

$S \leftarrow 0$

$I \leftarrow 1$

ΟΣΟ $I \leq 10$ ΕΠΑΝΑΛΑΒΕ

ΔΙΑΒΑΣΕ α

$S \leftarrow S + \alpha$

$I \leftarrow I + 1$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ S

α) Αν δοθεί σε κάθε επανάληψη ως είσοδος το 10 τι θα εμφανιστεί στην οθόνη του υπολογιστή; Με ποια συνθήκη θα εμφανιστεί το 10 μετά το τέλος επανάληψης; β) Αν η συνθήκη γίνει $I \leq 0$ τότε τι θα εμφανιστεί στην οθόνη του υπολογιστή γ) Τι υπολογίζει το παραπάνω πρόγραμμα;

Για το α ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης)
- ii. Όταν στην απάντηση του ο μαθητής αναφέρει ότι εμφανίζεται το 100 και γράφει τη σωστή συνθήκη ώστε να εμφανίζεται το 10 τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό)
- iii. Όταν στην απάντηση του ο μαθητής αναφέρει ότι εμφανίζεται το 100 και δεν απαντά ή απαντά γράφοντας λανθασμένη συνθήκη ώστε να εμφανίζεται το 10 τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης)
- iv. Όταν στην απάντηση του ο μαθητής αναφέρει ότι εμφανίζεται το 10 τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

4ο επίπεδο

Οι μαθητές έδωσαν ως απάντηση ότι θα εμφανιστεί το 100 και επίσης το 10 θα εμφανιστεί με τη συνθήκη $I \leq 1$

Πίνακας 4.5: Πίνακας τιμών ΔΡ1 (α ερώτημα)

S	0	10	20	30	4050	60	70	80	90	100
I	1	2	3	4	5	6	78	9	10	11
α	10	10	10	10	10	10	10	10	10	10
Έξοδος										100

3ο επίπεδο

Πίνακας 4.6: Πίνακας τιμών ΔΡ1 (α ερώτημα)

S	0	10	20	30	4050	60	70	80	90	100
I	1	2	3	4	5	6	78	9	10	11
α	10	10	10	10	10	10	10	10	10	10
Έξοδος										100

2ο επίπεδο

Θα εμφανιστεί το 10, δηλαδή εκτελεί 1 φορά την επανάληψη

1ο επίπεδο

$S=10, I=2$ ή $S=10, I=11$ ή $S \leftarrow \alpha$

Για το β ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν στην απάντηση του ο μαθητής αναφέρει ότι εμφανίζεται το μηδέν τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό)
- iii. Όταν στην απάντηση του ο μαθητής αναφέρει ότι η τιμή που έχει το S τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης)
- iv. Όταν στην απάντηση του ο μαθητής αναφέρει ότι δεν εμφανίζει τίποτα γιατί δεν κάνει επανάληψη τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά τότε τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

Οι μαθητές απάντησαν ότι θα εμφανιστεί το μηδέν (0).

3ο επίπεδο

Η τιμή που έχει το S.

2ο επίπεδο

Οι μαθητές απάντησαν ότι δεν θα εμφανιστεί τίποτα γιατί δεν θα γίνουν επαναλήψεις.

1ο επίπεδο

Οι μαθητές δεν απάντησαν.

Για το γ ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν στην απάντηση του ο μαθητής αναφέρει ότι εμφανίζεται το άθροισμα και το πλήθος ή μόνο το άθροισμα τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν στην απάντηση του ο μαθητής αναφέρει ότι εμφανίζεται το άθροισμα των αριθμών που είναι μικρότεροι ή ίσοι από το 10 τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν στην απάντηση του ο μαθητής αναφέρει ότι εμφανίζεται το πλήθος τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά τότε τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

Οι μαθητές απάντησαν ότι θα εμφανιστεί το άθροισμα και το πλήθος (S,I) ή μόνο το άθροισμα.

3ο επίπεδο

Το άθροισμα των αριθμών ≤ 10

2ο επίπεδο

Οι μαθητές απάντησαν ότι θα εμφανιστεί το πλήθος.

1ο επίπεδο

Οι μαθητές δεν απάντησαν.

Ανάλυση δεδομένων στο επίπεδο Εφαρμόζω

Η πρώτη δραστηριότητα (ΔΡ1) που δόθηκε στους μαθητές ήταν:

Δίνεται το παρακάτω πρόβλημα «Να γραφεί πρόγραμμα που θα διαβάζει τα ονόματα και τους χρόνους που έκαναν 20 αθλητές στο αγώνισμα των 400 μέτρων και να εμφανίζει το όνομα του πρωταθλητή». Συμπληρώστε τα κενά στο παρακάτω πρόγραμμα ώστε να αποτελεί λύση του παραπάνω προβλήματος.

ΠΡΟΓΡΑΜΜΑ Α2

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΜΙΝ

ΧΑΡΑΚΤΗΡΕΣ : Ο, ΟΜΙΝ

ΑΡΧΗ

ΜΙΝ ← 9999999999

ΟΜΙΝ ← ' '

(α) -----

ΔΙΑΒΑΣΕ Χ,Ο

ΑΝ Χ < ΜΙΝ ΤΟΤΕ

(β) -----

(γ) -----

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ΟΜΙΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Για το α ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης)
- ii. Όταν στην απάντηση του ο μαθητής υλοποιεί σωστά 20 επαναλήψεις τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό)
- iii. Όταν στην απάντηση του ο μαθητής δεν υλοποιεί σωστά την επανάληψη με λάθος μεταβλητές ή οριακές τιμές, τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης)
- iv. Όταν στην απάντηση του ο μαθητής δεν υλοποιεί σωστά την επανάληψη με λάθος μεταβλητές και τις αντίστοιχες οριακές τιμές η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

ΓΙΑ α ΑΠΟ 1 ΜΕΧΡΙ 20

3ο επίπεδο

ΟΣΟ $\alpha < 20$ ΕΠΑΝΕΛΑΒΕ, ΟΣΟ $\alpha > 20$ ΚΑΙ $\alpha \geq 20$ ' ΕΠΑΝΕΛΑΒΕ, ΓΙΑ α ΑΠΟ 1 ΜΕΧΡΙ 20

2ο επίπεδο

ΟΣΟ $\alpha > 20$ ΚΑΙ $\alpha \leq 20$ ΕΠΑΝΕΛΑΒΕ

ΟΣΟ $\alpha < 0$ ΕΠΑΝΕΛΑΒΕ

1ο επίπεδο

ΓΡΑΨΕ ΟΜΙΝ

Για το β ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης)
- ii. Όταν ο μαθητής απαντά δίνοντας τη σωστή εντολή εκχώρησης τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό)
- iii. Όταν ο μαθητής απαντά δίνοντας τη σωστή εντολή εκχώρησης αλλά κάνει συντακτικά λάθη τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης)
- iv. Όταν ο μαθητής εκχωρεί σε λάθος μεταβλητή τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

$\text{MIN} \leftarrow \alpha$

3ο επίπεδο

$\alpha \leftarrow \text{MIN}$

2ο επίπεδο

$\alpha \leftarrow \alpha + \text{MIN}$, $\alpha \leftarrow \alpha$, $\alpha \leftarrow 0$

1ο επίπεδο

ΓΡΑΨΕ ΟΜΙΝ, ΟΜΙΝ $\leftarrow 0$

Για το γ ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής απαντά δίνοντας τη σωστή εντολή εκχώρησης τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής εκχωρεί στη σωστή μεταβλητή λάθος αλφαριθμητική τιμή τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής εκχωρεί στη σωστή μεταβλητή αριθμητικές τιμές τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης)..
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

OMIN←O

3ο επίπεδο

OMIN←' '

2ο επίπεδο

OMIN←MIN ,OMIN←X

1ο επίπεδο

ΓΡΑΨΕ OMIN ή ΔΙΑΒΑΣΕ X ή MIN←X

Η δεύτερη δραστηριότητα (ΔΡ2) που δόθηκε στους μαθητές ήταν:

Δίνεται το παρακάτω πρόβλημα «Να γραφεί πρόγραμμα που θα διαβάζει τα ποσά που ξόδεψε κάποιος για τις εξόδους του σε ένα μήνα (θετικοί αριθμοί) και θα υπολογίζει και θα εμφανίζει το άθροισμα και το πλήθος τους». Συμπληρώστε τα κενά στο παρακάτω πρόγραμμα ώστε να αποτελεί λύση του παραπάνω προβλήματος.

ΠΡΟΓΡΑΜΜΑ Α3

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ : Χ, Σ

ΑΚΕΡΑΙΕΣ : ΠΛ

ΑΡΧΗ

ΠΛ ← 0

(α) -----

ΔΙΑΒΑΣΕ Χ

(β) -----

Σ ← Σ + Χ

(γ) -----

ΔΙΑΒΑΣΕ Χ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ Σ, ΠΛ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Για το α ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής απαντά δίνοντας τη σωστή εντολή εκχώρησης τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής εκχωρεί στη σωστή μεταβλητή μια λανθασμένη αρχική τιμή τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής εκχωρεί στη σωστή μεταβλητή αλφαριθμητικές τιμές τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης). Επίσης όταν ο μαθητής γράφει τη σωστή απάντηση του α ερωτήματος στη θέση του β ερωτήματος και το αντίστροφο τότε η απάντηση τοποθετείται στο 2ο επίπεδο.
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

$\Sigma \leftarrow 0$

3ο επίπεδο

$\Sigma \leftarrow 1$

2ο επίπεδο

$\Sigma \leftarrow ' '$

ΟΣΟ $X \geq 0$ ΕΠΑΝΕΛΑΒΕ

1ο επίπεδο

ΓΙΑ $X > 0$ ΕΠΑΝΑΛΑΒΕ ή ΓΙΑ X ΑΠΟ 1 ΜΕΧΡΙ 30 ή ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

Για το β ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν στην απάντηση του ο μαθητής υλοποιεί σωστά την επανάληψη τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν στην απάντηση του ο μαθητής υλοποιεί την επανάληψη με λάθος οριακές τιμές ή με λάθος μεταβλητή, τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν στην απάντηση του ο μαθητής δεν υλοποιεί σωστά την επανάληψη με λάθος μεταβλητές και τις αντίστοιχες οριακές τιμές η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης). Επίσης όταν ο μαθητής γράφει τη σωστή απάντηση του β ερωτήματος στη θέση του α ερωτήματος και το αντίστροφο τότε η απάντηση τοποθετείται στο 2ο επίπεδο. Καθώς και όταν αντί να χρησιμοποιήσει επανάληψη αυτός χρησιμοποιεί δομή επιλογής (Αν).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4^ο επίπεδο

ΟΣΟ $X > 0$ ΕΠΑΝΕΛΑΒΕ

3ο επίπεδο

ΟΣΟ $X \geq 0$ ΕΠΑΝΕΛΑΒΕ

ΟΣΟ $\Sigma > 0$ ΕΠΑΝΕΛΑΒΕ

2ο επίπεδο

ΑΝ $X > 0$ ΤΟΤΕ

$\Sigma \leftarrow 0$

1ο επίπεδο

ΓΙΑ $X > 0$ ΕΠΑΝΕΛΑΒΕ , ΓΙΑ $X \leq 0$ ΕΠΑΝΕΛΑΒΕ, $ΠΛ \leftarrow ΠΛ + 1$

Για το γ ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής απαντά δίνοντας τη σωστή εντολή εκχώρησης τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής απαντά γράφοντας την εντολή εκχώρησης για τη σωστή μεταβλητή που περιέχει στο δεξιό μέλος κάποια λογικά λάθη τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής απαντά γράφοντας για τη σωστή μεταβλητή την εντολή εκχώρησης με μια έκφραση εντελώς λανθασμένα τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

$ΠΛ \leftarrow ΠΛ + 1$

3ο επίπεδο

$ΠΛ \leftarrow \Sigma * ΠΛ$

$ΠΛ \leftarrow \Sigma / ΠΛ$

2ο επίπεδο

ΠΛ←Χ/Σ

1ο επίπεδο

ΓΡΑΨΕ Σ ή ΤΕΛΟΣ_ΑΝ ή ΜΕΧΡΙΣ_ΟΤΟΥ

Ανάλυση δεδομένων στο επίπεδο Αναλύω

Η πρώτη δραστηριότητα (ΔΡ1) που δόθηκε στους μαθητές ήταν:

Να γραφεί πρόγραμμα που θα διαβάζει θετικούς ακέραιους αριθμούς και θα εμφανίζει το τελευταίο ψηφίο τους, το πλήθος των άρτιων και το ποσοστό των περιττών.

Τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Όταν στην απάντηση του ο μαθητής υλοποιεί σωστά την επανάληψη, υπολογίζει σωστά το τελευταίο ψηφίο, το πλήθος των άρτιων και το ποσοστό των περιττών κάνοντας έλεγχο αν το πλήθος των επαναλήψεων είναι διάφορο ή ίσο με το μηδέν τότε η απάντηση αυτή είναι η βέλτιστη λύση οπότε η απάντηση τοποθετείται στο 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν στην απάντηση του ο μαθητής υλοποιεί σωστά την επανάληψη, υπολογίζει σωστά το τελευταίο ψηφίο, το πλήθος των άρτιων και το ποσοστό των περιττών τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό)
- iii. Όταν στην απάντηση του ο μαθητής υλοποιεί σωστά την επανάληψη, υπολογίζει σωστά το τελευταίο ψηφίο, υπάρχουν λάθη κατά τον υπολογισμό των άρτιων ή των περιττών ή το ποσοστό των περιττών υπολογίζεται λάθος τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης). Επίσης και όταν δεν υλοποιεί σωστά την επανάληψη αλλά υπολογίζει σωστά τα πλήθη και το ποσοστό.
- iv. Όταν στην απάντηση του ο μαθητής δεν υλοποιεί σωστά την επανάληψη, υπολογίζει σωστά το τελευταίο ψηφίο, υπάρχουν λάθη κατά τον υπολογισμό των άρτιων ή των περιττών ή το ποσοστό των περιττών δεν υπολογίζεται η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

5ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Π1, Π2, Π, ΑΡ, Τ

ΠΡΑΓΜΑΤΙΚΕΣ : ΠΟΣ

ΑΡΧΗ

$\Pi_1 \leftarrow 0$

$\Pi_2 \leftarrow 0$

$\Pi \leftarrow 0$

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ ΑΡ

ΜΕΧΡΙΣ_ΟΤΟΥ $ΑΡ > 0$ ΚΑΙ $ΑΡ = A_M(ΑΡ)$

ΟΣΟ $ΑΡ > 0$ ΕΠΑΝΑΛΑΒΕ

$T \leftarrow A \bmod 10$

ΓΡΑΨΕ Τ

ΑΝ $ΑΡ \bmod 2 = 0$ ΤΟΤΕ

$\Pi_1 \leftarrow \Pi_1 + 1$

ΑΛΛΙΩΣ

$\Pi_2 \leftarrow \Pi_2 + 1$

ΤΕΛΟΣ_ΑΝ

$\Pi \leftarrow \Pi + 1$

ΔΙΑΒΑΣΕ ΑΡ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ Π1

$\text{ΠΟΣ} \leftarrow 100 * \Pi_2 / \Pi$

ΓΡΑΨΕ ΠΟΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

4ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Α3

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ΠΛ_Π, ΠΛ_Α, Α, Τ

ΠΡΑΓΜΑΤΙΚΕΣ : ΠΟΣ

ΑΡΧΗ

ΠΛ_Π ← 0

ΠΛ_Α ← 0

ΓΡΑΨΕ ' ΔΩΣΕ ΘΕΤΙΚΟ ΑΚΕΡΑΙΟ ΑΡΙΘΜΟ '

ΔΙΑΒΑΣΕ Α

ΟΣΟ Α>0 ΕΠΑΝΑΛΑΒΕ

Τ ← A mod 10

ΓΡΑΨΕ Τ

ΑΝ A mod 2=0 ΤΟΤΕ

 ΠΛ_Α ← ΠΛ_Α +1

ΑΛΛΙΩΣ

 ΠΛ_Π ← ΠΛ_Π +1

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ ' ΔΩΣΕ ΘΕΤΙΚΟ ΑΚΕΡΑΙΟ ΑΡΙΘΜΟ '

ΔΙΑΒΑΣΕ Α

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ΠΛ_Α

ΠΟΣ ← ΠΛ_Π / (ΠΛ_Π + ΠΛ_Α) * 100

ΓΡΑΨΕ ' ΠΟΣΟΣΤΟ ΠΕΡΙΤΤΩΝ: ', ΠΟΣ, ' % '

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

3ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ3_1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ΠΛ, Χ, Ψ1

ΠΡΑΓΜΑΤΙΚΕΣ : Π

ΑΡΧΗ

ΠΛ ← 0

ΔΙΑΒΑΣΕ Χ

ΟΣΟ Χ > 0 ΕΠΑΝΑΛΑΒΕ

Ψ1 ← Χ mod 10

ΓΡΑΨΕ Ψ1

ΑΝ Χ mod 2 = 0 ΤΟΤΕ

 ΠΛ ← ΠΛ + 1

ΤΕΛΟΣ_ΑΝ

ΔΙΑΒΑΣΕ Χ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ΠΛ

ΠΟΣ ← 100 * Χ / ΠΛ

ΓΡΑΨΕ Π

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ Γ3_2

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ΠΠ, ΠΑ, Χ, ΤΨ, Ν,

ΠΡΑΓΜΑΤΙΚΕΣ : ΠΟΣ

ΑΡΧΗ

ΠΠ ← 0

ΠΑ ← 0

ΠΛ ← 0

ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ Ν

ΔΙΑΒΑΣΕ Χ

ΤΨ ← Χ mod 10

ΠΛ ← ΠΛ + 1

ΓΡΑΨΕ ΤΨ

ΑΝ Χ mod 2 = 0 ΤΟΤΕ

 ΠΑ ← ΠΑ + 1

ΑΛΛΙΩΣ

ΠΠ ← ΠΠ + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ΠΑ

ΠΟΣ ← 100 * ΠΠ / ΠΛ

ΓΡΑΨΕ ΠΟΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ Γ3_3

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ΠΠ, ΠΑ, Χ, ΤΨ,

Σ1, Σ2

ΠΡΑΓΜΑΤΙΚΕΣ : ΠΕΡ, ΑΡ

ΑΡΧΗ

ΠΠ ← 0

ΠΑ ← 0

ΔΙΑΒΑΣΕ Χ

ΟΣΟ Χ > 0 ΕΠΑΝΑΛΑΒΕ

ΤΨ ← Χ mod 10

ΓΡΑΨΕ ΤΨ

ΑΝ ΤΨ mod 2 = 0 ΤΟΤΕ

 ΠΑ ← ΠΑ + 1

Σ1 ← Σ1 + ΤΨ

ΑΛΛΙΩΣ

ΠΠ ← ΠΠ + 1

Σ2 ← Σ2 + ΤΨ

ΤΕΛΟΣ_ΑΝ

ΔΙΑΒΑΣΕ Χ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ΠΑ, ΠΠ, Σ1, Σ2

ΑΡ ← ΠΑ / Σ1

ΠΕΡ ← ΠΠ * Σ2 / 100

ΓΡΑΨΕ ΑΡ, ΠΕΡ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

2ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ2_1

ΜΕΤΑΒΛΗΤΕΣ

ΑΡΧΗ

$\Pi \leftarrow 0$

ΔΙΑΒΑΣΕ Χ

ΟΣΟ Χ>0 ΕΠΑΝΑΛΑΒΕ

$\Psi \leftarrow X \bmod 10$

ΓΡΑΨΕ Ψ1

ΑΝ $X \bmod 2=0$ ΤΟΤΕ

$\Pi \leftarrow \Pi + 1$

ΤΕΛΟΣ_ΑΝ

ΠΡΟΓΡΑΜΜΑ Γ2_2

ΜΕΤΑΒΛΗΤΕΣ

ΑΡΧΗ

$\Pi \leftarrow 0$

ΔΙΑΒΑΣΕ Χ

ΜΕΧΡΙΣ_ΟΤΟΥ Χ>0

ΑΝ $X \bmod 2=0$ ΤΟΤΕ

$\Pi \leftarrow \Pi + 1$

ΤΕΛΟΣ_ΑΝ

ΑΝ $X \bmod 2 <> 0$ ΤΟΤΕ

$\Pi \leftarrow \Pi + 1$

ΤΕΛΟΣ_ΑΝ

ΠΡΟΓΡΑΜΜΑ Γ2_3

ΜΕΤΑΒΛΗΤΕΣ

ΑΡΧΗ

$\Pi \leftarrow 0$

ΔΙΑΒΑΣΕ Χ

ΟΣΟ Χ>0 ΕΠΑΝΑΛΑΒΕ

$\Psi \leftarrow X \bmod 10$

$\Pi \leftarrow \Pi + 1$

ΑΝ $X \bmod 2=0$ ΤΟΤΕ

ΔΙΑΒΑΣΕ Χ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ Ψ, Π

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

1ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ1_1

ΜΕΤΑΒΛΗΤΕΣ

ΑΡΧΗ

ΑΝ $X \bmod 2=0$ ΕΠΑΝΕΛΑΒΕ

ΔΙΑΒΑΣΕ Χ

ΑΝ $X \text{ div } 2=0$ ΤΟΤΕ

$\chi \leftarrow \alpha\rho$

ΑΛΛΙΩΣ

$\chi \leftarrow \pi\rho\rho$

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

$\Pi \leftarrow \alpha\rho * \chi$

ΓΡΑΨΕ Π

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ Γ1_2

ΜΕΤΑΒΛΗΤΕΣ

ΑΡΧΗ

$\Pi \leftarrow 0$

$\Pi\alpha\rho \leftarrow 0$

$\Pi\pi \leftarrow 0$

ΔΙΑΒΑΣΕ Χ

ΑΝ Χ>0 ΤΟΤΕ

$\Psi \leftarrow X \bmod 10$

ΓΡΑΨΕ Ψ

$\Pi\omicron\varsigma \leftarrow \Pi\alpha\rho + 1(\Pi\pi/\Pi)*100$

ΓΡΑΨΕ ΠΟΨ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ Γ1_3

ΜΕΤΑΒΛΗΤΕΣ

ΑΡΧΗ

$\Pi \leftarrow 0$

ΑΝ $X \bmod 2=0$ ΤΟΤΕ

$\Pi \leftarrow \Pi + 1$

ΓΡΑΨΕ Ψ

ΑΛΛΙΩΣ

$\Pi \leftarrow \Pi + 1$

ΓΡΑΨΕ Π

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Η δεύτερη δραστηριότητα (ΔΡ2) που δόθηκε στους μαθητές ήταν:

Μια εταιρεία δημοσκοπήσεων θέτει κάποιο ερώτημα σε ένα δείγμα 2000 πολιτών. Για την επεξεργασία των δεδομένων να δημιουργήσετε πρόγραμμα που θα κάνει τα εξής: Διαβάζει το φύλο του πολίτη ελέγχοντας ότι είναι Άνδρας ή Γυναίκα (Α=Άνδρας, Γ=Γυναίκα) να διαβάζει την απάντηση που έδωσε κάθε πολίτης, η οποία μπορεί να είναι «ΝΑΙ», «ΟΧΙ», «ΔΕΝ ΞΕΡΩ» ελέγχοντας ώστε η απάντηση να παίρνει αποκλειστικά μόνο αυτές τις τιμές. Κατόπιν να υπολογίζει και να εμφανίζει το πλήθος των ατόμων που απάντησαν «ΝΑΙ» καθώς και το ποσοστό των γυναικών και των ανδρών που απάντησαν «ΟΧΙ».

Τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Όταν στην απάντηση του ο μαθητής χρησιμοποιεί τη δομή επανάληψης Για με όριο το 2000, γίνεται έλεγχος εγκυρότητας για το φύλο και την απάντηση, και σωστός υπολογισμός του πλήθους των ΝΑΙ και του ζητούμενου ποσοστού τότε η απάντηση αυτή είναι η βέλτιστη λύση οπότε η απάντηση τοποθετείται στο 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν στην απάντηση του ο μαθητής χρησιμοποιεί τη δομή επανάληψης Για με όριο το 2000, δεν γίνεται έλεγχος εγκυρότητας για το φύλο και την απάντηση, γίνεται σωστός υπολογισμός του πλήθους των ΝΑΙ και του ζητούμενου ποσοστού τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν στην απάντηση του ο μαθητής χρησιμοποιεί δομή επανάληψης με όριο το 2000, γίνεται έλεγχος εγκυρότητας για το φύλο και την απάντηση, γίνεται σωστός υπολογισμός του πλήθους των ΝΑΙ και υπολογίζει το ποσοστό στο σύνολο των γυναικών ή υπολογίζει τα ποσοστά στην απάντηση ΝΑΙ τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης). Επίσης όταν τα ποσοστά υπολογίζονται με λανθασμένο γενικά τρόπο τότε η απάντηση τοποθετείται στο 3ο επίπεδο.
- iv. Όταν στην απάντηση του ο μαθητής χρησιμοποιεί δομή επανάληψης με όριο το 2000, γίνεται έλεγχος εγκυρότητας για το φύλο και την απάντηση, γίνεται σωστός ή εν μέρει υπολογισμός του πλήθους των απαντήσεων η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

5ο επίπεδο

Οι μαθητές δεν απάντησαν

4ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ4

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Ι, Π, Π1, Π2, Π3

ΠΡΑΓΜΑΤΙΚΕΣ : ΠΑ, ΠΓ

ΧΑΡΑΚΤΗΡΕΣ: Α, Φ

ΑΡΧΗ

$\Pi \leftarrow 0$

$\Pi_2 \leftarrow 0$

$\Pi_3 \leftarrow 0$

$\Pi \leftarrow 0$

ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ 2000

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Φ, Α

ΜΕΧΡΙΣ_ΟΤΟΥ Φ=Α Ή Φ=Γ ΚΑΙ Α=ΝΑΙ Ή Α=ΟΧΙ Ή Α=ΔΕΝ ΞΕΡΩ

ΑΝ Α=ΝΑΙ ΤΟΤΕ

$\Pi \leftarrow \Pi + 1$

ΤΕΛΟΣ_ΑΝ

ΑΝ Α=ΟΧΙ ΤΟΤΕ

$\Pi_3 \leftarrow \Pi_3 + 1$

ΑΝ Φ=Α ΤΟΤΕ

$\Pi_1 \leftarrow \Pi_1 + 1$

ΑΛΛΙΩΣ

$\Pi_2 \leftarrow \Pi_2 + 1$

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ Π

ΑΝ $\Pi_3 < 0$ ΤΟΤΕ

$\Pi_A \leftarrow 100 * \Pi_1 / \Pi_3$

$\Pi_G \leftarrow 100 * \Pi_2 / \Pi_3$

ΓΡΑΨΕ ΠΑ, ΠΓ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

3ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ3_1

ΜΕΤΑΒΛΗΤΕΣ

ΝΑΙ ← ΓΝ + ΑΝ

ΑΚΕΡΑΙΕΣ: Ι, ΓΝ, ΓΟ, ΑΝ, ΑΟ, ΝΑΙ

ΓΡΑΨΕ ΝΑΙ

ΠΡΑΓΜΑΤΙΚΕΣ : ΠΟΣΑ, ΠΟΣΓ

ΠΟΣΓ ← ΓΟ / (ΓΝ + ΓΟ)

ΧΑΡΑΚΤΗΡΕΣ: Α, Φ

ΠΟΣΑ ← ΑΟ / (ΑΝ + ΑΟ)

ΑΡΧΗ

ΓΡΑΨΕ ΠΟΣΓ, ΠΟΣΑ

ΓΝ ← 0

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΓΟ ← 0

ΑΝ ← 0

ΑΟ ← 0

ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ 2000

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Φ

ΜΕΧΡΙΣ_ΟΤΟΥ Φ = Α Ή Φ = Γ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Α

ΜΕΧΡΙΣ_ΟΤΟΥ Α = ΝΑΙ Ή Α = ΟΧΙ Ή Α = ΔΕΝ ΞΕΡΩ

ΑΝ Φ = Γ ΤΟΤΕ

ΑΝ Α = ΝΑΙ ΤΟΤΕ

ΓΝ ← ΓΝ + 1

ΑΛΛΙΩΣ

ΓΟ ← ΓΟ + 1

ΤΕΛΟΣ_ΑΝ

ΑΛΛΙΩΣ

ΑΝ Α = ΝΑΙ ΤΟΤΕ

ΑΝ ← ΑΝ + 1

ΑΛΛΙΩΣ

ΑΟ ← ΑΟ + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΠΡΟΓΡΑΜΜΑ Γ3_2
 ΜΕΤΑΒΛΗΤΕΣ
 ΑΚΕΡΑΙΕΣ: Ι, Π1, Π2, Π3
 ΠΡΑΓΜΑΤΙΚΕΣ : ΠΟΣ1, ΠΟΣ2
 ΧΑΡΑΚΤΗΡΕΣ: Α, Φ
 ΑΡΧΗ
 Π1 ← 0
 Π2 ← 0
 Π3 ← 0
 ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ 2000
 ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
 ΔΙΑΒΑΣΕ Φ
 ΜΕΧΡΙΣ_ΟΤΟΥ Φ=Α Ή Φ=Γ
 ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
 ΔΙΑΒΑΣΕ Α
 ΜΕΧΡΙΣ_ΟΤΟΥ Α=ΝΑΙ Ή Α=ΟΧΙ Ή Α=ΔΕΝ ΞΕΡΩ
 ΑΝ Α=ΝΑΙ ΤΟΤΕ
 Π1 ← Π1 +1
 ΤΕΛΟΣ_ΑΝ
 ΑΝ Α=ΟΧΙ ΚΑΙ Φ=Γ ΤΟΤΕ
 Π2 ← Π2 +1
 ΤΕΛΟΣ_ΑΝ
 ΑΝ Α=ΟΧΙ ΚΑΙ Φ=Α ΤΟΤΕ
 Π3 ← Π3 +1
 ΤΕΛΟΣ_ΑΝ
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 ΠΟΣ1 ← 100*Π2/2000
 ΠΟΣ2 ← 100*Π3/2000
 ΓΡΑΨΕ Π1, Π2, Π3, ΠΟΣ1, ΠΟΣ2
 ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ Γ3_3
 ΜΕΤΑΒΛΗΤΕΣ
 ΑΚΕΡΑΙΕΣ: Ι, Π, Π1, Π2
 ΠΡΑΓΜΑΤΙΚΕΣ : ΠΑ, ΠΓ
 ΧΑΡΑΚΤΗΡΕΣ: Α, Φ
 ΑΡΧΗ
 Π1 ← 0
 Π2 ← 0
 Π3 ← 0
 Π ← 0
 ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ 2000
 ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
 ΔΙΑΒΑΣΕ Φ
 ΜΕΧΡΙΣ_ΟΤΟΥ Φ=Α Ή Φ=Γ
 ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
 ΔΙΑΒΑΣΕ Α
 ΜΕΧΡΙΣ_ΟΤΟΥ Α=ΝΑΙ Ή Α=ΟΧΙ Ή Α=ΔΕΝ ΞΕΡΩ
 ΑΝ Α=ΝΑΙ ΤΟΤΕ
 Π ← Π +1
 ΤΕΛΟΣ_ΑΝ
 ΑΝ Α=ΟΧΙ ΤΟΤΕ
 ΑΝ Φ=Α ΤΟΤΕ
 Π1 ← Π1 +1
 ΑΛΛΙΩΣ
 Π2 ← Π2 +1
 ΤΕΛΟΣ_ΑΝ
 ΤΕΛΟΣ_ΑΝ
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 ΓΡΑΨΕ Π
 ΠΑ ← 100*Π1/Π
 ΠΓ ← 100*Π2/Π
 ΓΡΑΨΕ ΠΑ, ΠΓ
 ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

2ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ2_1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Ι, ΠΛ, ΠΛ1, ΠΛ2

ΠΡΑΓΜΑΤΙΚΕΣ : ΠΑ, ΠΓ

ΧΑΡΑΚΤΗΡΕΣ: Α, Φ

ΑΡΧΗ

ΠΛ1 ← 0

ΠΛ2 ← 0

ΠΛ3 ← 0

ΠΛ ← 0

ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ 2000

ΔΙΑΒΑΣΕ Φ, Α

ΑΝ Φ = Α ΤΟΤΕ

 ΓΡΑΨΕ 'ΑΝΔΡΑΣ'

ΑΛΛΙΩΣ

 ΓΡΑΨΕ 'ΓΥΝΑΙΚΑ'

ΤΕΛΟΣ_ΑΝ

ΑΝ Α = 'ΝΑΙ' ΤΟΤΕ

 ΠΛ1 ← ΠΛ1 + 1

ΑΛΛΙΩΣ_ΑΝ Α = 'ΟΧΙ' ΤΟΤΕ

 ΠΛ2 ← ΠΛ2 + 1

ΑΛΛΙΩΣ_ΑΝ Α = 'ΔΕΝ ΞΕΡΩ' ΤΟΤΕ

ΠΛ3 ← ΠΛ3 + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΠΑ ← 100 * ΠΛ1 / ΠΛ

ΠΓ ← 100 * ΠΛ2 / ΠΛ

ΓΡΑΨΕ ΠΑ, ΠΓ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ Γ2_2

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ : ΠΟΣΑ, ΠΟΣΓ

ΧΑΡΑΚΤΗΡΕΣ: Α, Φ

ΑΡΧΗ

ΠΝ ← 0

ΑΝ ← 0

ΓΝ ← 0

ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ 2000

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

 ΔΙΑΒΑΣΕ Φ

ΜΕΧΡΙΣ_ΟΤΟΥ Φ = Α Ή Φ = Γ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

 ΔΙΑΒΑΣΕ Α

ΜΕΧΡΙΣ_ΟΤΟΥ Α = 'ΝΑΙ' Ή Α = 'ΟΧΙ' Ή Α = 'ΔΕΝ ΞΕΡΩ'

ΑΝ Α = 'ΝΑΙ' ΤΟΤΕ

 ΠΝ ← ΠΝ + 1

 ΑΝ ← ΑΝ + 1

ΑΛΛΙΩΣ

 ΓΝ ← ΓΝ + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΠΟΣΓ ← ΓΝ / ΠΝ * 100

ΠΟΣΑ ← ΑΝ / ΠΝ * 100

ΓΡΑΨΕ ΠΟΣΓ, ΠΟΣΑ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

1ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Α, Γ

ΑΡΧΗ

ΠΛ←0

ΔΙΑΒΑΣΕ Α, Γ

ΓΡΑΨΕ 'ΝΑΙ', 'ΟΧΙ', 'ΔΕΝ

ΞΕΡΩ'

ΠΡΟΓΡΑΜΜΑ Γ1

ΜΕΤΑΒΛΗΤΕΣ

ΧΑΡΑΚΤΗΡΕΣ: ΑΠ, Φ

ΑΡΧΗ

ΠΛ←0

ΑΝ Φ= 'Α' ΤΟΤΕ

ΓΡΑΨΕ 'ΑΝΔΡΑΣ'

ΑΝ Φ= 'Γ' ΤΟΤΕ

ΓΡΑΨΕ 'ΓΥΝΑΙΚΑ'

ΠΡΟΓΡΑΜΜΑ Γ1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ:

ΠΡΑΓΜΑΤΙΚΕΣ

ΑΡΧΗ

ΠΛ←0

ΔΙΑΒΑΣΕ Φ

Ανάλυση δεδομένων στο επίπεδο Αξιολογώ

Η πρώτη δραστηριότητα (ΔΡ1) που δόθηκε στους μαθητές ήταν:

Δίνεται το πρόγραμμα:

ΠΡΟΓΡΑΜΜΑ Α1

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΑΤ

ΑΡΧΗ

ΔΙΑΒΑΣΕ Χ

ΟΣΟ Χ<>0 ΕΠΑΝΑΛΑΒΕ

ΑΝ Χ>0 ΤΟΤΕ

ΑΤ ← Χ

ΑΛΛΙΩΣ

ΑΤ ← Χ*(-1)

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ΑΤ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

α)Τι υπολογίζει η μεταβλητή ΑΤ;

β)Εντοπίστε και διορθώστε το λάθος στο παραπάνω πρόγραμμα

γ)πως μπορεί να τροποποιηθεί το παραπάνω πρόγραμμα ώστε με λιγότερες εντολές να δίνει το ίδιο αποτέλεσμα;

Για το α ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5^ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής αντιλαμβάνεται σωστά το πρόβλημα και δίνει ως απάντηση την απόλυτη τιμή τότε η απάντηση τοποθετείται στο 4^ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής δίνει ως απάντηση ότι το πρόγραμμα υπολογίζει τις θετικές τιμές του X τότε η απάντηση τοποθετείται στο 3^ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής δίνει ως απάντηση ότι το πρόγραμμα υπολογίζει τις τιμές του X τότε η απάντηση τοποθετείται στο 2^ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1^ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

Οι μαθητές απάντησαν την απόλυτη τιμή του X

3ο επίπεδο

Τις θετικές τιμές του X

2ο επίπεδο

Υπολογίζει τις τιμές του X ή τις τιμές του X αν είναι θετικό ή αν είναι αρνητικό

1ο επίπεδο

Αρχική τιμή ή ακέραιες τιμές ή αλφαριθμητικές τιμές

Για το β ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Όταν ο μαθητής αντιλαμβάνεται σωστά το πρόβλημα και δίνει ως απάντηση τη σωστή εντολή στη σωστή θέση και εμφανίζει την απόλυτη τιμή μέσα στην επανάληψη στη σωστή θέση τότε η απάντηση τοποθετείται στο 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής αντιλαμβάνεται σωστά το πρόβλημα και δίνει ως απάντηση τη σωστή εντολή στη σωστή θέση τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής δίνει ως απάντηση τη σωστή εντολή σε λάθος θέση μέσα στο πρόγραμμα τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής δίνει ως απάντηση μια λανθασμένη εντολή που αφορά τη σωστή μεταβλητή τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

5ο επίπεδο	4ο επίπεδο	3ο επίπεδο
ΠΡΟΓΡΑΜΜΑ Α1	ΠΡΟΓΡΑΜΜΑ Α1	ΠΡΟΓΡΑΜΜΑ Α1
ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΑΤ	ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΑΤ	ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΑΤ
ΑΡΧΗ	ΑΡΧΗ	ΑΡΧΗ
ΔΙΑΒΑΣΕ Χ	ΔΙΑΒΑΣΕ Χ	ΔΙΑΒΑΣΕ Χ
ΟΣΟ Χ<>0 ΕΠΑΝΑΛΑΒΕ	ΟΣΟ Χ<>0 ΕΠΑΝΑΛΑΒΕ	ΟΣΟ Χ<>0 ΕΠΑΝΑΛΑΒΕ
ΑΝ Χ>0 ΤΟΤΕ	ΑΝ Χ>0 ΤΟΤΕ	ΔΙΑΒΑΣΕ Χ
ΑΤ ← Χ	ΑΤ ← Χ	ΑΝ Χ>0 ΤΟΤΕ
ΑΛΛΙΩΣ	ΑΛΛΙΩΣ	ΑΤ ← Χ
ΑΤ ← Χ*(-1)	ΑΤ ← Χ*(-1)	ΑΛΛΙΩΣ
ΤΕΛΟΣ_ΑΝ	ΤΕΛΟΣ_ΑΝ	ΑΤ ← Χ*(-1)
ΓΡΑΨΕ ΑΤ	ΔΙΑΒΑΣΕ Χ	ΤΕΛΟΣ_ΑΝ
ΔΙΑΒΑΣΕ Χ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΓΡΑΨΕ ΑΤ	ΓΡΑΨΕ ΑΤ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

2ο επίπεδο

Δεν εμφανίζεται το Χ

Λείπει η αρχικοποίηση του Χ

1ο επίπεδο

Η εντολή ΔΙΑΒΑΣΕ ΑΤ σε διάφορες θέσεις μέσα στο πρόγραμμα(μέσα στο Αν ή πριν το ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ)

Αρχικοποίηση της μεταβλητής ΑΤ

Για το γ ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Όταν ο μαθητής αντιλαμβάνεται σωστά το πρόβλημα και δίνει ως απάντηση τη βέλτιστη λύση με χρήση της συνάρτησης της απόλυτης τιμής τότε η απάντηση τοποθετείται στο 5ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής αντιλαμβάνεται σωστά το πρόβλημα και δίνει ως απάντηση σωστή λύση με απλή επιλογή (Αν) τότε η απάντηση τοποθετείται στο 4ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής υπολογίζει μόνο την περίπτωση που δίνεται αρνητικός αριθμός ως τιμή μέσα στο πρόγραμμα τότε η απάντηση τοποθετείται στο 3ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής αντικαθιστά τη Όσο με Μέχρις_ότου ή με Αν τότε η απάντηση τοποθετείται στο 2ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

5ο επίπεδο	4ο επίπεδο	3ο επίπεδο
ΠΡΟΓΡΑΜΜΑ Α1	ΠΡΟΓΡΑΜΜΑ Α1	ΠΡΟΓΡΑΜΜΑ Α1
ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΑΤ	ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΑΤ	ΠΡΑΓΜΑΤΙΚΕΣ : Χ, ΑΤ
ΑΡΧΗ	ΑΡΧΗ	ΑΡΧΗ
ΔΙΑΒΑΣΕ Χ	ΔΙΑΒΑΣΕ Χ	ΔΙΑΒΑΣΕ Χ
ΟΣΟ Χ<>0 ΕΠΑΝΑΛΑΒΕ	ΟΣΟ Χ<>0 ΕΠΑΝΑΛΑΒΕ	ΟΣΟ Χ<>0 ΕΠΑΝΑΛΑΒΕ
ΑΤ←Α_Τ(Χ)	ΑΤ ← Χ	ΑΝ Χ<0 ΤΟΤΕ
ΔΙΑΒΑΣΕ Χ	ΑΝ Χ<0 ΤΟΤΕ	ΑΤ ← Χ*(-1)
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΑΤ ← Χ*(-1)	ΤΕΛΟΣ_ΑΝ
ΓΡΑΨΕ ΑΤ	ΤΕΛΟΣ_ΑΝ	ΔΙΑΒΑΣΕ Χ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	ΔΙΑΒΑΣΕ Χ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΓΡΑΨΕ ΑΤ
	ΓΡΑΨΕ ΑΤ	ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
	ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	

2ο επίπεδο

ΠΡΟΓΡΑΜΜΑ A2_1
ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ : X, AT
ΑΡΧΗ
ΔΙΑΒΑΣΕ X
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
 AN X>0 ΤΟΤΕ
 AT ← X
 ΑΛΛΙΩΣ
 AT ← X*(-1)
ΤΕΛΟΣ_ΑΝ
ΔΙΑΒΑΣΕ X
ΜΕΧΡΙΣ_ΟΤΟΥ X=0
ΓΡΑΨΕ AT
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ A2_2
ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ : X, AT
ΑΡΧΗ
ΔΙΑΒΑΣΕ X
AN X<>0 ΤΟΤΕ
 AN X>0 ΤΟΤΕ
 AT ← X
 ΑΛΛΙΩΣ
 AT ← X*(-1)
ΤΕΛΟΣ_ΑΝ
ΔΙΑΒΑΣΕ X
ΤΕΛΟΣ_ΑΝ
ΓΡΑΨΕ AT
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ A2_3
ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ : X, AT
ΑΡΧΗ
ΔΙΑΒΑΣΕ X
 AN X>0 ΤΟΤΕ
 AT ← X
 ΑΛΛΙΩΣ
 AT ← X*(-1)
ΤΕΛΟΣ_ΑΝ
ΓΡΑΨΕ AT
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

1ο επίπεδο

ΠΡΟΓΡΑΜΜΑ A1_1
ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ : X, AT
ΑΡΧΗ
ΔΙΑΒΑΣΕ X
ΓΙΑ X<>0 ΕΠΑΝΑΛΑΒΕ
 AN X>0 ΤΟΤΕ
 AT ← X
 ΑΛΛΙΩΣ
 AT ← X*(-1)
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ AT
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ A1_2
ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ : X, AT
ΑΡΧΗ
ΔΙΑΒΑΣΕ X
ΓΙΑ X<>0 ΕΠΑΝΑΛΑΒΕ
 ΓΙΑ X>0 ΤΟΤΕ
 AT ← X
 ΑΛΛΙΩΣ
 AT ← X*(-1)
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ AT
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Η δεύτερη δραστηριότητα (ΔΡ2) που δόθηκε στους μαθητές ήταν:

Δίνεται το πρόγραμμα:

ΠΡΟΓΡΑΜΜΑ Α2

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ :B, M

ΑΚΕΡΑΙΕΣ : K

ΑΡΧΗ

$K \leftarrow 1$

ΟΣΟ $K \leq 10$ ΕΠΑΝΑΛΑΒΕ

ΔΙΑΒΑΣΕ B

ΑΝ $B > M$ ΤΟΤΕ

$M \leftarrow B$

ΤΕΛΟΣ_ΑΝ

$K \leftarrow K + 1$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ M

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

α) Τι υπολογίζει η μεταβλητή M;

β) Εντοπίστε και διορθώστε το λάθος στο παραπάνω πρόγραμμα

γ) Πως μπορεί να τροποποιηθεί το παραπάνω πρόγραμμα ώστε με λιγότερες εντολές να δίνει το ίδιο αποτέλεσμα;

Για το α ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5^ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής αντιλαμβάνεται σωστά το πρόβλημα και δίνει ως απάντηση το μέγιστο τότε η απάντηση τοποθετείται στο 4^ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής δίνει ως απάντηση τη μεταβλητή max τότε η απάντηση τοποθετείται στο 3^ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής δίνει ως απάντηση τιμές που σχετίζονται με τη μεταβλητή B τότε η απάντηση τοποθετείται στο 2^ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1^ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο

Οι μαθητές απάντησαν τη μέγιστη τιμή

3ο επίπεδο

Την μεταβλητή \max

2ο επίπεδο

Μια πραγματική τιμή ή την μεταβλητή B

1ο επίπεδο

Δεν απαντούν

Για το β ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5^ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής αντιλαμβάνεται σωστά το πρόβλημα και δίνει ως απάντηση τη σωστή εντολή στη σωστή θέση τότε η απάντηση τοποθετείται στο 4^ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής δίνει ως απάντηση τη σωστή εντολή σε λάθος θέση ή στη σωστή θέση με λάθος τιμή μέσα στο πρόγραμμα τότε η απάντηση τοποθετείται στο 3^ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής δίνει ως απάντηση μια λανθασμένη εντολή που αφορά τη σωστή μεταβλητή τότε η απάντηση τοποθετείται στο 2^ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1^ο επίπεδο (προδρομικό).

4ο επίπεδο

Λείπει η αρχικοποίηση του Μ πριν την επανάληψη

ΠΡΟΓΡΑΜΜΑ Α2

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ :Β, Μ

ΑΚΕΡΑΙΕΣ : Κ

ΑΡΧΗ

Κ ← 1

Μ ← -9999999999999999

ΟΣΟ Κ <= 10 ΕΠΑΝΑΛΑΒΕ

ΔΙΑΒΑΣΕ Β

ΑΝ Β > Μ ΤΟΤΕ

 Β ← Μ

ΤΕΛΟΣ_ΑΝ

Κ ← Κ + 1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ Μ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

3ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Α2

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ :B, M

ΑΚΕΡΑΙΕΣ : K

ΑΡΧΗ

$K \leftarrow 1$

$M \leftarrow 0$

ΟΣΟ $K \leq 10$ ΕΠΑΝΑΛΑΒΕ

ΔΙΑΒΑΣΕ B

ΑΝ $B > M$ ΤΟΤΕ

$B \leftarrow M$

ΤΕΛΟΣ_ΑΝ

$K \leftarrow K + 1$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ M

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ή

Λείπει η αρχικοποίηση του M

$M \leftarrow -9999999999999999$

2ο επίπεδο

Λείπει η αρχικοποίηση του M

1ο επίπεδο

ΔΙΑΒΑΣΕ B ή ΔΙΑΒΑΣΕ M ή ΔΙΑΒΑΣΕ K

Για το γ ερώτημα τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Λόγω της φύσης της δραστηριότητας δεν ορίζεται το 5^ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν ο μαθητής αντιλαμβάνεται σωστά το πρόβλημα και μετατρέπει το Όσο σε Για σωστά συντακτικά τότε η απάντηση τοποθετείται στο 4^ο επίπεδο (συνδυαστικό).
- iii. Όταν ο μαθητής μετατρέπει το Όσο σε Για με ελάχιστα συντακτικά λάθη τότε η απάντηση τοποθετείται στο 3^ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν ο μαθητής μετατρέπει το Όσο σε Για με πολλά συντακτικά λάθη τότε η απάντηση τότε η απάντηση τοποθετείται στο 2^ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1^ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

4ο επίπεδο	3ο επίπεδο
ΠΡΟΓΡΑΜΜΑ Α2	ΠΡΟΓΡΑΜΜΑ Α2
ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ :B, M	ΠΡΑΓΜΑΤΙΚΕΣ :B, M
ΑΚΕΡΑΙΕΣ : K	ΑΚΕΡΑΙΕΣ : K
ΑΡΧΗ	ΑΡΧΗ
M←-999999999999999999	M←-999999999999999999
ΓΙΑ Κ ΑΠΟ 1 ΜΕΧΡΙ 10	ΓΙΑ Κ ΑΠΟ 1 ΜΕΧΡΙ 10
ΔΙΑΒΑΣΕ Β	ΔΙΑΒΑΣΕ Β
ΑΝ Β>Μ ΤΟΤΕ	ΑΝ Β>Μ ΤΟΤΕ
B ← M	B ← M
ΤΕΛΟΣ_ΑΝ	ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	K←K+1
ΓΡΑΨΕ Μ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	ΓΡΑΨΕ Μ
	ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

2ο επίπεδο

Οι μαθητές δεν απάντησαν

1ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Α1_1	ΠΡΟΓΡΑΜΜΑ Α1_2	ΠΡΟΓΡΑΜΜΑ Α1_3
ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ
ΠΡΑΓΜΑΤΙΚΕΣ :B, M	ΠΡΑΓΜΑΤΙΚΕΣ :B, M	ΠΡΑΓΜΑΤΙΚΕΣ :B, M
ΑΚΕΡΑΙΕΣ : K	ΑΚΕΡΑΙΕΣ : K	ΑΚΕΡΑΙΕΣ : K
ΑΡΧΗ	ΑΡΧΗ	ΑΡΧΗ
$K \leftarrow 1$	$K \leftarrow 1$	$K \leftarrow 1$
ΟΣΟ $K \leq 10$ ΕΠΑΝΑΛΑΒΕ	ΓΙΑ $K \leq 10$ ΚΑΙ	ΓΙΑ $K \leq 10$ ΕΠΑΝΑΛΑΒΕ
ΑΝ $B > M$ ΤΟΤΕ	$B > M$ ΕΠΑΝΑΛΑΒΕ	ΑΝ $B > M$ ΤΟΤΕ
$B \leftarrow M$	$K \leftarrow K + 1$	$B \leftarrow M$
ΤΕΛΟΣ_ΑΝ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΤΕΛΟΣ_ΑΝ
$K \leftarrow K + 1$	ΓΡΑΨΕ M	$K \leftarrow K + 1$
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ M		ΓΡΑΨΕ M
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ		ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Ανάλυση δεδομένων στο επίπεδο Δημιουργώ

Η δραστηριότητα που δόθηκε στους μαθητές ήταν:

Σε ένα κινητό τηλέφωνο ο χρήστης πρέπει να εισάγει το PIN στην αρχή της λειτουργίας του. Το κινητό ανοίγει όταν δοθεί το σωστό PIN. Ο χρήστης μπορεί να κάνει το πολύ 3 προσπάθειες. Μετά από τις 3 αποτυχημένες προσπάθειες το τηλέφωνο κλειδώνει. Να γραφεί πρόγραμμα που θα διαβάζει το αποθηκευμένο PIN του κινητού. Επίσης θα διαβάζει τους αριθμούς που θα εισάγει ο χρήστης. Θα εμφανίζει το μήνυμα «Καλώς ήρθατε» όταν το κινητό ανοίγει ή το μήνυμα «κλείδωσε» σε περίπτωση αποτυχίας. Επίσης θα εμφανίζει τον αριθμό προσπαθειών εισαγωγής PIN στο κινητό.

Τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ορίζονται ως εξής:

- i. Όταν στην απάντηση του ο μαθητής χρησιμοποιεί τη δομή επανάληψης Μέχρις_ότου με συνθήκη το πλήθος των επαναλήψεων να είναι 3 ή το αποθηκευμένο Pin να είναι ίσο με το τυχαίο Pin, γίνεται σωστός υπολογισμός των προσπαθειών και σωστή εμφάνιση των κατάλληλων μηνυμάτων τότε η απάντηση αυτή είναι η βέλτιστη λύση οπότε η απάντηση τοποθετείται στο 5^ο επίπεδο (εκτεταμένης θεώρησης).
- ii. Όταν στην απάντηση του ο μαθητής χρησιμοποιεί τη δομή επανάληψης Όσο .. Επανάλαβε με συνθήκη που πληρεί τις προϋποθέσεις του προβλήματος και γίνεται σωστός υπολογισμός των προσπαθειών και σωστή εμφάνιση των κατάλληλων μηνυμάτων τότε η απάντηση τοποθετείται στο 4^ο επίπεδο (συνδυαστικό).
- iii. Όταν στην απάντηση του ο μαθητής χρησιμοποιεί δομή επανάληψης με συνθήκη που δεν ελέγχει όλες τις απαιτήσεις του προβλήματος δηλαδή όλες τις οριακές τιμές ώστε να γίνεται σωστός υπολογισμός των προσπαθειών ή δεν γίνεται ο κατάλληλος έλεγχος με Αν για τη σωστή εμφάνιση μηνυμάτων τότε η απάντηση τοποθετείται στο 3^ο επίπεδο (προσεγγιστικής κατανόησης).
- iv. Όταν στην απάντηση του ο μαθητής χρησιμοποιεί δομή επανάληψης εντελώς λανθασμένα όμως γίνεται ο κατάλληλος έλεγχος για τη σωστή εμφάνιση μηνυμάτων η απάντηση τοποθετείται στο 2^ο επίπεδο (επιμέρους κατανόησης).
- v. Όταν ο μαθητής δεν απαντά ή δίνει μια άσχετη απάντηση, αυτή τοποθετείται στο 1^ο επίπεδο (προδρομικό).

Ενδεικτικές απαντήσεις μαθητών

5ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ5

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ : Π

ΧΑΡΑΚΤΗΡΕΣ: ΠΙΝ, ΠΡ

ΑΡΧΗ

$\Pi \leftarrow 0$

ΔΙΑΒΑΣΕ ΠΙΝ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ ΠΡ

$\Pi \leftarrow \Pi + 1$

ΜΕΧΡΙΣ_ΟΤΟΥ ΠΡ=ΠΙΝ Ή Π=3

ΑΝ ΠΡ=ΠΙΝ ΤΟΤΕ

ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ '

ΑΛΛΙΩΣ

ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ '

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ Π

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

4ο επίπεδο

Δεν απάντησαν οι μαθητές

3^ο επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ3_1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ : Π

ΧΑΡΑΚΤΗΡΕΣ: ΠΙΝ, ΠΡ

ΑΡΧΗ

Π ← 0

ΔΙΑΒΑΣΕ ΠΙΝ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ ΠΡ

Π ← Π + 1

ΜΕΧΡΙΣ_ΟΤΟΥ Π=3

ΑΝ ΠΡ=ΠΙΝ ΤΟΤΕ

ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ '

ΑΛΛΙΩΣ

ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ '

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ Π

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ Γ3_2

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ : Κ, ΠΙΝ, ΠΡ

ΑΡΧΗ

Κ ← 1

ΔΙΑΒΑΣΕ ΠΙΝ, ΠΡ

ΟΣΟ Κ ≤ 2 ΚΑΙ ΠΡ <> ΠΙΝ

ΕΠΑΝΑΛΑΒΕ

ΔΙΑΒΑΣΕ ΠΡ

Κ ← Κ + 1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ ΠΡ=ΠΙΝ ΤΟΤΕ

ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ '

ΑΛΛΙΩΣ

ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ '

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ Κ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΠΡΟΓΡΑΜΜΑ Γ3_3

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ : Π

ΧΑΡΑΚΤΗΡΕΣ: ΠΙΝ, ΠΡ

ΑΡΧΗ

Π ← 0

ΔΙΑΒΑΣΕ ΠΙΝ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ ΠΡ

Π ← Π + 1

ΜΕΧΡΙΣ_ΟΤΟΥ ΠΡ=ΠΙΝ

ΑΝ ΠΡ=ΠΙΝ ΤΟΤΕ

ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ '

ΑΛΛΙΩΣ

ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ '

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ Π

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

2° επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ2_1	ΠΡΟΓΡΑΜΜΑ Γ2_2	ΠΡΟΓΡΑΜΜΑ Γ2_3
ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ
ΑΡΧΗ	ΑΚΕΡΑΙΕΣ : Χ	ΑΚΕΡΑΙΕΣ : ΠΙΝ, Φ, Ψ, ΑΡΠ
Π←123	ΧΑΡΑΚΤΗΡΕΣ: ΠΙΝ, ΠΙΝ1	ΑΡΧΗ
ΔΙΑΒΑΣΕ ΠΙΝ	ΑΡΧΗ	Φ←0
ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ 3	Χ←0	ΑΡΠ←0
ΑΝ ΠΙΝ<>Π ΤΟΤΕ	ΔΙΑΒΑΣΕ ΠΙΝ, ΠΙΝ1	ΟΣΟ Φ<3 ΚΑΙ Ψ<>' '
ΓΡΑΨΕ ' ΛΑΘΟΣ ΡΙΝ'	ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ 2	&ΕΠΑΝΑΛΑΒΕ
ΔΙΑΒΑΣΕ ΠΙΝ	ΑΝ ΠΙΝ<>ΠΙΝ1 ΤΟΤΕ	ΑΝ ΠΙΝ ΣΩΣΤΟ ΤΟΤΕ
ΤΕΛΟΣ_ΑΝ	Χ←Χ+1	ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΑΛΛΙΩΣ_ΑΝ ΠΙΝ=ΠΙΝ1ΤΟΤΕ	ΑΛΛΙΩΣ
ΑΝ Π=ΠΙΝ ΤΟΤΕ	ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ '	ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ'
ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ '	ΤΕΛΟΣ_ΑΝ	ΤΕΛΟΣ_ΑΝ
ΑΛΛΙΩΣ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	Φ←Φ+1
ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ'	ΑΝ Χ=3 ΤΟΤΕ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΑΝ	ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ'	ΓΡΑΨΕ ΑΡΠ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	ΤΕΛΟΣ_ΑΝ	ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
	ΓΡΑΨΕ Χ	
	ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	

1° επίπεδο

ΠΡΟΓΡΑΜΜΑ Γ1_1	ΠΡΟΓΡΑΜΜΑ Γ1_2	ΠΡΟΓΡΑΜΜΑ Γ1_3
ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ
ΑΡΧΗ	ΑΡΧΗ	ΑΚΕΡΑΙΕΣ : Χ
Χ←3	ΔΙΑΒΑΣΕ Κ	ΧΑΡΑΚΤΗΡΕΣ: Π
ΟΣΟ Χ<>3 ΕΠΑΝΑΛΑΒΕ	ΟΣΟ Κ=ΑΛΗΘΗΣ ΕΠΑΝΑΛΑΒΕ	ΑΡΧΗ
ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ'	ΓΡΑΨΕ ' ΚΑΛΩΣΗΡΘΑΤΕ'	ΔΙΑΒΑΣΕ Π, Χ
ΑΛΛΙΩΣ	ΔΙΑΒΑΣΕ Κ	
ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ'	ΑΝ Κ<> ΑΛΗΘΗΣ ΤΟΤΕ	
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΓΡΑΨΕ ' ΚΛΕΙΔΩΣΕ'	
ΓΡΑΨΕ ΠΙΝ	ΤΕΛΟΣ_ΑΝ	
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ	ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	

5. Αποτελέσματα

5.1 Αποτελέσματα ως προς το Επίπεδο Θυμάμαι

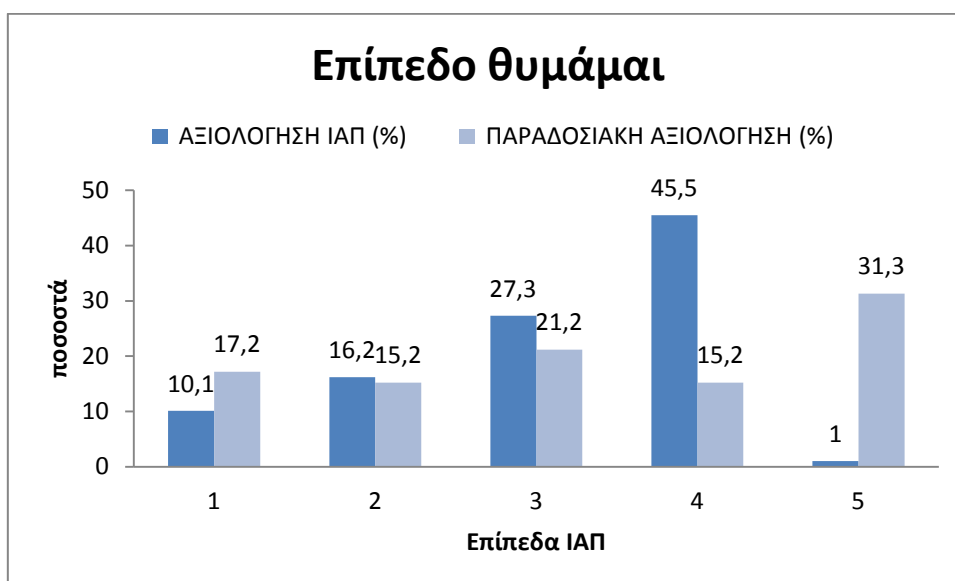
Το επίπεδο «Θυμάμαι», ορίζεται ως η ανάκτηση σχετικής γνώσης από τη μακροχρόνια μνήμη. Οι δραστηριότητες που δόθηκαν απαιτούσαν οι μαθητές να μπορούν να εκτελούν εργασίες με γνώσεις που είχαν ήδη μάθει κατά τη διδασκαλία του προγραμματισμού.

Πίνακας 5.1:Αξιολόγηση ΙΑΠ επίπεδο Θυμάμαι

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΘΥΜΑΜΑΙ			ΑΞΙΟΛΟΓΗΣΗ ΙΑΠ (%)
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	
1	5	5	0	10,1
2	7	6	3	16,2
3	13	6	8	27,3
4	8	15	22	45,5
5	0	1	0	1,0

Πίνακας 5.2:Αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Θυμάμαι

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΘΥΜΑΜΑΙ			ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ (%)
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	
1	7	9	1	17,2
2	9	4	2	15,2
3	7	4	10	21,2
4	2	4	9	15,2
5	8	12	11	31,3



Σχήμα 5.1: Σύγκριση αποτελεσμάτων επίπεδο θυμάμαι

Στον πίνακα 5.1 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.2 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.1 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο θυμάμαι της αναθεωρημένης ταξινόμιας Bloom και με τους δύο τρόπους αξιολόγησης.

Αξιολόγηση γνώσεων σύμφωνα με ΙΑΠ

Η δραστηριότητα που δόθηκε στους μαθητές για το επίπεδο θυμάμαι είχε τρία ερωτήματα. Το πρώτο ερώτημα σχετιζόταν με τους τύπους των δεδομένων που χρησιμοποιούνται στο πρόγραμμα και παραδείγματα με εντολή εκχώρησης. Από τον πίνακα 5.1 διαπιστώνεται ότι οι απαντήσεις των περισσότερων μαθητών βρίσκονται στο 3^ο (13 απαντήσεις) και 4^ο επίπεδο (8 απαντήσεις) ΙΑΠ. Οπότε συμπεραίνουμε ότι οι περισσότεροι μαθητές γνωρίζουν καλά τους τύπους δεδομένων και μπορούν να δώσουν παράδειγμα για κάθε τύπο αλλά αυτό γίνεται συνήθως χωρίς εντολή εκχώρησης. Πιθανά οι μαθητές να εστίασαν στη λέξη παραδείγματα που αναγράφονταν στην εκφώνηση της δραστηριότητας και όχι στον τρόπο που έπρεπε να δώσουν τα παραδείγματα τους. Ενδέχεται όμως οι μαθητές να μη κατανοούν την έννοια της εντολής εκχώρησης δηλαδή την ορολογία αυτή επειδή από ότι διαφαίνεται στην πράξη την χρησιμοποιούν σωστά.

Επίσης μόνο 5 από τις 33 απαντήσεις ανήκουν στο 1^ο επίπεδο (προδρομικό). Λίγοι μαθητές δεν γνωρίζουν τους τύπους δεδομένων που χρησιμοποιούνται σε ένα πρόγραμμα. Κανένας μαθητής δεν έδωσε απάντηση που να ανήκει στο επίπεδο της εκτεταμένης θεώρησης.

Στο δεύτερο ερώτημα της δραστηριότητας ζητήθηκε από τους μαθητές να γράψουν τη σύνταξη των τριών δομών επανάληψης. Από τον πίνακα 5.1 παρατηρούμε ότι οι μισοί περίπου μαθητές έχουν τη δυνατότητα να γράφουν σωστά τη σύνταξη σε κάθε δομή επανάληψης. Οι απαντήσεις αυτών των μαθητών τοποθετούνται στο 4^ο επίπεδο ΙΑΠ. Ενώ ισάριθμες είναι οι απαντήσεις των μαθητών στο 2^ο και 3^ο επίπεδο ΙΑΠ. Μόνο 5 από τους 33 μαθητές ανήκουν στο 1^ο επίπεδο ΙΑΠ, δεν γνωρίζουν τις δομές επανάληψης καθόλου. Επίσης ένας μαθητής απαντά στο 5^ο επίπεδο. Επομένως οι μαθητές έχουν ένα καλό επίπεδο γνώσεων για το πώς συντάσσονται οι τρεις δομές επανάληψης.

Στο τρίτο ερώτημα της δραστηριότητας ζητήθηκε από τους μαθητές να γράψουν όλα τα σχήματα που χρησιμοποιούνται σε ένα διάγραμμα ροής και να εξηγήσουν την λειτουργία τους. Από τον πίνακα 5.1 παρατηρούμε ότι 22 απαντήσεις ανήκουν στο 4^ο επίπεδο ΙΑΠ και 8 απαντήσεις ανήκουν στο 3^ο επίπεδο ΙΑΠ. Επομένως σχεδόν όλοι οι μαθητές γνωρίζουν τα σχήματα (είτε όλα είτε τα περισσότερα από αυτά) στο διάγραμμα ροής και ποια είναι η χρήση τους. Δεν υπάρχει μαθητής που να μη γνωρίζει κανένα σχήμα εφόσον καμιά απάντηση δεν υπάρχει στο 1^ο επίπεδο ΙΑΠ.

Συνολικά για το επίπεδο Θυμάμαι έχουμε τα εξής αποτελέσματα:

- το 10,1% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να ανακαλέσουν καμιά γνώση προγραμματισμού από αυτές που τους ζητήθηκε να απαντήσουν στη συγκεκριμένη έρευνα. Από αυτούς το 5% δεν θυμάται τους τύπους δεδομένων που χρησιμοποιούνται σε ένα πρόγραμμα και το άλλο 5% δεν θυμούνται τις δομές επανάληψης, τη δομή ΟΣΟ...ΕΠΑΝΑΛΑΒΕ, τη δομή ΓΙΑ ...ΑΠΟ ΜΕΧΡΙΜΕ_ΒΗΜΑ και τη δομή ΜΕΧΡΙΣ_ΟΤΟΥ.

- Το 16,2% των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Οι μαθητές αυτοί έχουν κάποιες απλές γνώσεις προγραμματισμού. Από αυτούς το 7.1% των μαθητών αναφέρει όλους τους τύπους μεταβλητών και για κάθε τύπο δίνουν ένα παράδειγμα τιμής της κάθε μεταβλητής κάνοντας κάποιο λάθος είτε στον τύπο της μεταβλητής είτε στο παράδειγμα ή εναλλακτικά αναφέρουν μόνο τους τύπους δεδομένων. Επίσης το 6,1% των μαθητών γράφουν τη σύνταξη μιας μόνο δομής σωστά ή λείπει εντελώς η συνθήκη. Τέλος το 3% των μαθητών έχουν γράψει κάποιο σχήμα λάθος ή δίνουν μια λανθασμένη εξήγηση για κάποιο σχήμα ή γράφουν σωστά μόνο τα σχήματα που χρησιμοποιούνται στο διάγραμμα ροής.

- Το 27,3% των μαθητών προσεγγίζουν σωστά τις γνώσεις που χρειάστηκε να ανακαλέσουν από τη μνήμη τους απαντώντας στο 3^ο επίπεδο ΙΑΠ. Συγκεκριμένα το 13,1% των μαθητών αναφέρουν όλους τους τύπους δεδομένων και για κάθε τύπο δίνουν ένα παράδειγμα τιμής της κάθε μεταβλητής ή λείπει κάποιος τύπος δεδομένων. Το 6,1% των μαθητών έχουν γράψει τη σύνταξη των τριών δομών επανάληψης αλλά υπάρχουν λάθη στη σύνταξη ή παραλείπεται η σύνταξη από μία δομή επανάληψης. Ενώ το 8,1% των μαθητών παραλείπει κάποια σχήματα ή δεν εξηγεί τη λειτουργία όλων των σχημάτων.

- Το 45,5% των μαθητών αποδίδει με πληρότητα τα επιμέρους ερωτήματα και οι απαντήσεις τους τοποθετούνται στο 4^ο επίπεδο ΙΑΠ. Από αυτούς το 8,1% των μαθητών αναφέρει τους τύπους μεταβλητών και για κάθε τύπο δίνει την κατάλληλη εντολή εκχώρησης. Ενώ το 15,2% των μαθητών έχουν γράψει σωστά τη σύνταξη των δομών επανάληψης στη γενική μορφή ή με παράδειγμα. Τέλος το 22,2% των μαθητών αυτών έχει γράψει όλα τα σχήματα που χρησιμοποιούνται στο διάγραμμα ροής καθώς επίσης εξηγεί σωστά τη λειτουργία του κάθε σχήματος.

- Το 1% των μαθητών δίνει απαντήσεις που τοποθετούνται στο 5^ο επίπεδο ΙΑΠ. Για το πρώτο ερώτημα δεν υπήρχε κανένας μαθητής που να αναφέρει όλους τους τύπους μεταβλητών γράφοντας και τους αντίστοιχους ορισμούς και για κάθε τύπο να γράφει μια εντολή εκχώρησης. Ενώ για το δεύτερο ερώτημα μόνο ένας μαθητής έγραψε τη σύνταξη των δομών επανάληψης στη γενική μορφή και τη λειτουργία της κάθε δομής επανάληψης. Επίσης να αναφερθεί ότι στο τρίτο ερώτημα που αφορούσε τα σχήματα στο διάγραμμα ροής και τη χρήση τους δεν ορίζεται το 5^ο επίπεδο επειδή η δραστηριότητα ήταν πάρα πολύ απλή.

Σύγκριση τρόπων αξιολόγησης για το επίπεδο Θυμάμαι

Στο επίπεδο Θυμάμαι της αναθεωρημένης ταξινομίας Bloom ελέγχεται αν ο μαθητής είναι σε θέση να αναγνωρίσει ή να ανακαλέσει γνώσεις που έχει διδαχθεί στην τάξη χωρίς να είναι απαραίτητη οποιαδήποτε κατανόηση γι 'αυτό. Η αξιολόγηση της δραστηριότητας, η οποία είχε τρία επιμέρους ερωτήματα, κατηγοριοποιήθηκε στα αντίστοιχα 5 επίπεδα ΙΑΠ όπως φαίνεται στον πίνακα 5.1. Στη συνέχεια οι αντίστοιχες βαθμολογίες των ανεξάρτητων βαθμολογητών ομαδοποιήθηκαν στα 5 επίπεδα της ΙΑΠ σύμφωνα με τον πίνακα 5.2, ώστε να μπορούν να συγκριθούν οι δύο τρόποι αξιολόγησης ως προς τα αποτελέσματα τους.

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 10,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 17,2% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν να ανακαλέσουν γνώσεις προγραμματισμού που αφορούν τους τύπους δεδομένων, τις δομές επανάληψης, και τα σχήματα στο διάγραμμα ροής δεν διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης. Στο προδρομικό επίπεδο η ανάκληση γνώσεων τείνει να προσεγγίζει την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 16,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 15,2% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις γνώσεις για τους τύπους δεδομένων, τις δομές επανάληψης και τα σχήματα στο διάγραμμα ροής σχεδόν ταυτίζονται και με τους δύο τρόπους αξιολόγησης.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 27,3% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 21,2% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που μπορούν να ανακαλέσουν γνώσεις προγραμματισμού σε ικανοποιητικό βαθμό που αφορούν τους τύπους

δεδομένων, τις δομές επανάληψης, και τα σχήματα στο διάγραμμα ροής δεν διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης.

- Στο 4^ο και 5^ο επίπεδο αθροιστικά οι μαθητές απαντούν με το ίδιο ποσοστό (46,5%) και με τους δύο τρόπους αξιολόγησης. Στην συγκεκριμένη δραστηριότητα που αφορά την απομνημόνευση και την ανάκληση γνώσεων για τους τύπους δεδομένων, τις δομές επανάληψης και τα διαγράμματα ροής διαπιστώνεται ότι οι τρόποι αξιολόγησης δίνουν ακριβώς το ίδιο αποτέλεσμα συνολικά και στα δύο αυτά επίπεδα ΙΑΠ. Επειδή δεν ορίζεται σε όλα τα ερωτήματα το 5^ο επίπεδο για αυτή τη δραστηριότητα θα θεωρήσουμε ότι αποτελεί μαζί με το 4^ο ένα επίπεδο.

5.2 Αποτελέσματα ως προς το Επίπεδο Κατανόω

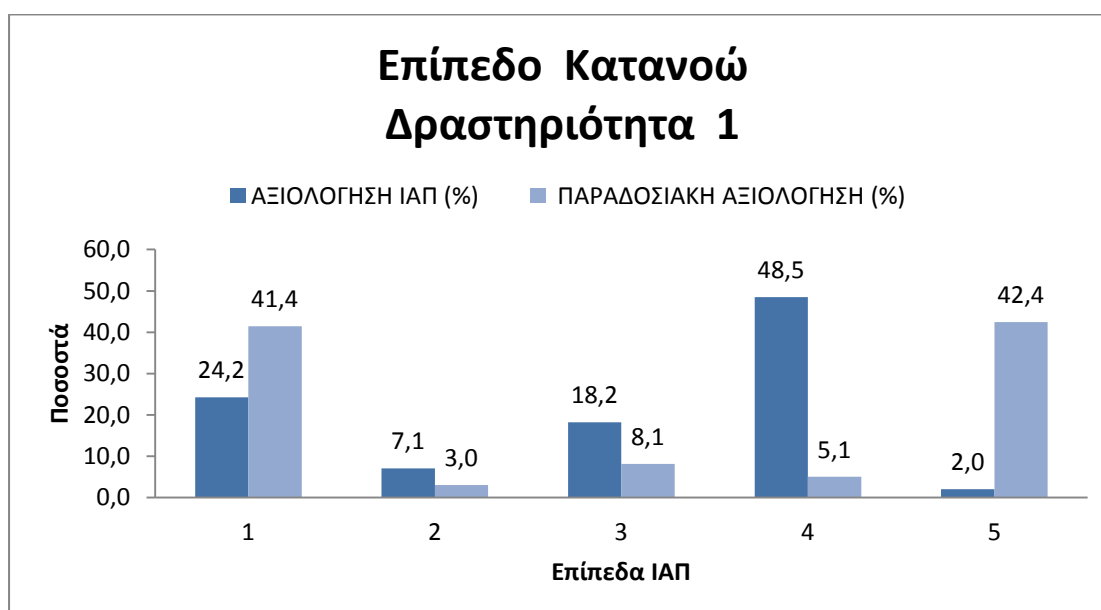
Το επίπεδο «Κατανόω» έχει ορισθεί ως η δημιουργία νοήματος από προφορικά, γραπτά και γραφικά μηνύματα. Περιλαμβάνει ερμηνεία, διευκρίνιση, ταξινόμηση, ανακεφαλαίωση, συμπέρασμα. Οι δραστηριότητες σε αυτό το επίπεδο έχουν ως πιθανά ζητούμενα τη μετατροπή ενός αλγόριθμου από μια μορφή αναπαράστασης σε μια άλλη μορφή ή μιας δομής σε άλλη ισοδύναμη, την εξήγηση μιας έννοιας ή ενός αλγορίθμου ή ενός προγράμματος.

Πίνακας 5.1:ΔΡ1 αξιολόγηση ΙΑΠ επίπεδο Κατανόω

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΚΑΤΑΝΟΩ			ΑΞΙΟΛΟΓΗΣΗ ΙΑΠ (%)
	ΔΡΑΣΤΗΡΙΟΤΗΤΑ 1			
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	
1	12	4	8	24,2
2	1	4	2	7,1
3	6	8	4	18,2
4	14	17	17	48,5
5	0	0	2	2,0

Πίνακας 5.2: ΔΡ1 αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Κατανόω

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΚΑΤΑΝΟΩ			ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ (%)
	ΔΡΣΤΗΡΙΟΤΗΤΑ 1			
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	
1	12	15	14	41,4
2	1	0	2	3,0
3	5	0	3	8,1
4	1	0	4	5,1
5	14	18	10	42,4



Σχήμα 5.2: ΔΡ1 σύγκριση αποτελεσμάτων επίπεδο Κατανόω

Αξιολόγηση γνώσεων μαθητών σύμφωνα με την ΙΑΠ

Στον πίνακα 5.3 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.4 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.2 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Κατανόω της αναθεωρημένης ταξινόμιας Bloom και με τους δύο τρόπους αξιολόγησης.

Η πρώτη δραστηριότητα που δόθηκε στους μαθητές για το επίπεδο Κατανοώ είχε τρία ερωτήματα. Στο πρώτο ερώτημα θα έπρεπε οι μαθητές να εκτελέσουν μια δομή επανάληψης ΟΣΟ και να περιγράψουν τι εμφανίζεται κατά την εκτέλεση. Από τον πίνακα 5.3 διαφαίνεται ότι οι απαντήσεις των μαθητών είναι σχεδόν ισάριθμες στο 4^ο επίπεδο (14 απαντήσεις) και στο 1^ο επίπεδο (12 απαντήσεις). Άρα οι μαθητές είτε μπορούν και εκτελούν σωστά το πρόγραμμα είτε όχι. Αξίζει να επισημανθεί ότι 6 μαθητές εκτελούν λιγότερες ή περισσότερες επαναλήψεις (3^ο επίπεδο). Αυτό οφείλεται στις τιμές εισόδου και τερματισμού της επανάληψης.

Στο δεύτερο ερώτημα οι μαθητές έπρεπε να βρουν την βέλτιστη οριακή τιμή ώστε το τμήμα προγράμματος να δίνει ίδια αποτελέσματα με αυτό που τους δόθηκε. Διαπιστώνεται ότι οι μισοί περίπου μαθητές (17 απαντήσεις) ανήκουν στο 4^ο επίπεδο, επομένως έχουν τη δυνατότητα να βρίσκουν τη σωστή βέλτιστη οριακή τιμή, ενώ 4 απαντήσεις ανήκουν στο 1^ο επίπεδο. Στο τρίτο ερώτημα οι μαθητές έπρεπε να μετατρέψουν την δομή επανάληψης ΟΣΟ σε ΓΙΑ ή εναλλακτικά σε ΜΕΧΡΙΣ_ΟΤΟΥ. Και σε αυτό το ερώτημα επίσης οι μαθητές μπορούν να κάνουν σωστά αυτή τη μετατροπή καθώς στο 4^ο επίπεδο ανήκουν 17 απαντήσεις μαθητών. Επίσης αξιοσημείωτο είναι το γεγονός ότι 8 μαθητές (περίπου το 25%) απαντούν στο 1^ο επίπεδο, επομένως δεν γνωρίζουν πως μετατρέπεται η δομή ΟΣΟ σε κάποια άλλη δομή επανάληψης.

Συνολικά ως προς το επίπεδο Κατανοώ για την 1η δραστηριότητα έχουμε τα εξής αποτελέσματα:

- Το 24,2% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να ερμηνεύσουν τη λειτουργία ενός προγράμματος, να το εκτελέσουν σωστά και να μετατρέψουν σωστά μια δομή επανάληψης σε άλλη. Συγκεκριμένα, από αυτούς το 12,1% δεν έχει τη δυνατότητα να εκτελέσει το πρόγραμμα, το 4% δεν μπορεί να το ερμηνεύσει και 8,1% δε μπορεί να μετατρέψει την ΟΣΟ σε ΓΙΑ ή ΜΕΧΡΙΣ_ΟΤΟΥ

- Το 7,1% των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να εκτελέσουν, να ερμηνεύσουν και να μετατρέψουν στοιχειώδεις και πολύ απλές λειτουργίες ενός προγράμματος. Το 1% εκτελεί την επανάληψη θεωρώντας ότι το βήμα είναι το 1, το 4 % εκτελεί λιγότερες ή περισσότερες επαναλήψεις, ενώ το 2,1 % μετατρέπει την ΟΣΟ όχι σε μια άλλη δομή επανάληψης αλλά σε δομή επιλογής.

- Το 18,2% απαντά στο 3^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές έχουν τη δυνατότητα να προσεγγίσουν σε ικανοποιητικό βαθμό, την εκτέλεση, ερμηνεία ενός προγράμματος καθώς και τη μετατροπή μιας δομής επανάληψης σε άλλη. Το 6,1% δυσκολεύεται να κατανοήσει με ποιες αρχικές ή οριακές τιμές εκτελείται η επανάληψη, επίσης το 8,1%, στη βέλτιστη τελευταία τιμή εκτέλεσης και το 4 % στην συνθήκη τερματισμού της επανάληψης.

- Το 48,5% των μαθητών αποδίδουν με πληρότητα τα επιμέρους ερωτήματα και οι απαντήσεις τους τοποθετούνται στο 4^ο επίπεδο ΙΑΠ. Από αυτούς το 14,1% εκτελούν σωστά το

πρόγραμμα, το 17,2% μπορούν να ερμηνεύσουν με πληρότητα το πρόγραμμα και επίσης το 17,2% κάνουν σωστή μετατροπή της ΟΣΟ σε άλλη δομή επανάληψης.

- Το 5^ο επίπεδο δεν ορίζεται για το πρώτο και δεύτερο ερώτημα. Ενώ στο τρίτο ερώτημα που αφορούσε την μετατροπή της ΟΣΟ σε άλλη δομή επανάληψης μόνο το 2% των μαθητών απαντά στο επίπεδο αυτό.

Σύγκριση τρόπων αξιολόγησης για το επίπεδο Κατανού

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 24,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 41,4% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν να εκτελέσουν μια δομή επανάληψης ΟΣΟ και να περιγράψουν τι εμφανίζεται κατά την εκτέλεση, να βρουν την βέλτιστη οριακή τιμή ώστε το τμήμα προγράμματος να δίνει ίδια αποτελέσματα με αυτό που τους δόθηκε, καθώς και να μετατρέψουν μια δομή επανάληψης σε κάποια άλλη διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης. Για το προδρομικό επίπεδο η κατανόηση γνώσεων, υποδιπλασιάζεται με την αξιολόγηση ΙΑΠ συγκριτικά με την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 7,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 3% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις γνώσεις για την εκτέλεση ενός προγράμματος, την ερμηνεία του προγράμματος καθώς και τη μετατροπή μιας δομής σε άλλη οι διπλάσιοι σε αριθμό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση. Στο επίπεδο επιμέρους κατανόησης η κατανόηση γνώσεων είναι πολλαπλάσια με την αξιολόγηση ΙΑΠ από ότι με την παραδοσιακή αξιολόγηση.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 18,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 8,1% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που μπορούν να κατανοήσουν γνώσεις προγραμματισμού σε ικανοποιητικό βαθμό που αφορούν την εκτέλεση επαναλήψεων, τις οριακές τιμές ελέγχου μιας συνθήκης καθώς και τις μετατροπές από μια δομή σε άλλη υπερδιπλασιάζονται με την αξιολόγηση ΙΑΠ. Στο επίπεδο προσεγγιστικής κατανόησης η κατανόηση γνώσεων πολλαπλασιάζεται με την ΙΑΠ.

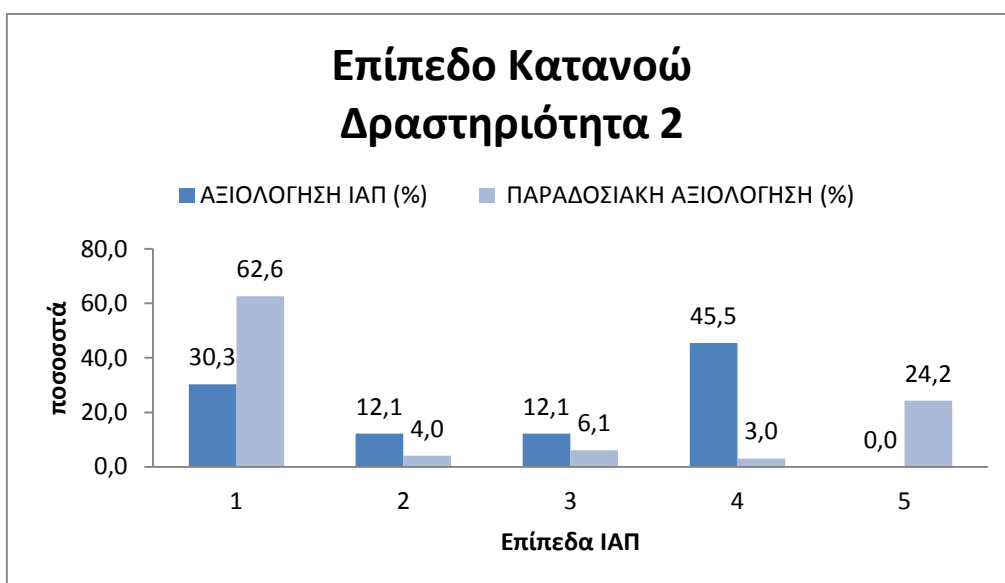
- Στο 4^ο και 5^ο επίπεδο αθροιστικά οι μαθητές απαντούν σε ποσοστό 50,5% με την ΙΑΠ και 47,5% με την παραδοσιακή αξιολόγηση. Στην συγκεκριμένη δραστηριότητα που αφορά την ερμηνεία και την κατανόηση γνώσεων στις δομές επανάληψης διαπιστώνεται ότι οι δύο τρόποι αξιολόγησης συγκλίνουν ως προς το αποτέλεσμα συνολικά και στα 2 επίπεδα ΙΑΠ. Επειδή δεν ορίζεται σε όλα τα ερωτήματα το 5^ο επίπεδο της ΙΑΠ για αυτή τη δραστηριότητα θα θεωρήσουμε ότι αποτελεί ένα επίπεδο μαζί με το 4^ο επίπεδο.

Πίνακας 5.3: ΔΡ2 αξιολόγηση ΙΑΠ επίπεδο Κατανοώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΚΑΤΑΝΟΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2			ΑΞΙΟΛΟΓΗΣΗ ΙΑΠ (%)
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	
	1	11	10	
2	4	6	2	12,1
3	10	1	1	12,1
4	8	16	21	45,5
5	0	0	0	0,0

Πίνακας 5.4: ΔΡ2 αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο κατανοώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΚΑΤΑΝΟΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2			ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ (%)
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	
	1	17	19	
2	2	0	2	4,0
3	4	0	2	6,1
4	2	0	1	3,0
5	8	14	2	24,2



Σχήμα 5.3: ΔΡ2 σύγκριση αποτελεσμάτων επίπεδο Καταννώ

Αξιολόγηση γνώσεων μαθητών σύμφωνα με την ΙΑΠ

Στον πίνακα 5.5 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.6 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.3 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Καταννώ της αναθεωρημένης ταξινόμιας Bloom και με τους δύο τρόπους αξιολόγησης.

Η δεύτερη δραστηριότητα που δόθηκε στους μαθητές για το επίπεδο Καταννώ είχε τρία ερωτήματα. Στο πρώτο, θα έπρεπε οι μαθητές να υπολογίσουν το άθροισμα τιμών που εισάγονται από το πληκτρολόγιο σε μια δομή επανάληψης ΟΣΟ καθώς και με ποια συνθήκη υπολογίζεται ένα προκαθορισμένο άθροισμα. Από τον πίνακα 5.5 διαφαίνεται ότι οι απαντήσεις των μαθητών για το ερώτημα αυτό είναι σχεδόν ισάριθμες στο 1^ο (11 απαντήσεις) και 3^ο επίπεδο (10 απαντήσεις). Οπότε συμπεραίνουμε ότι όσοι μαθητές μπορούν να αθροίζουν σε μια επανάληψη χωρίς να μπορούν να τροποποιούν μια συνθήκη με βάση το άθροισμα, άλλοι τόσοι αδυνατούν να εκτελέσουν μια τέτοια διαδικασία. Επίσης να αναφερθεί ότι μόνο 4 μαθητές εκτελούν την επανάληψη 1 φορά (2^ο επίπεδο).

Στο δεύτερο ερώτημα χρειάζεται να εκτελεστεί η επανάληψη για την οριακή τιμή. Διαπιστώνεται ότι ο αριθμός των μαθητών, αθροιστικά, που ανήκουν στο 1^ο (10 απαντήσεις) και 2^ο επίπεδο (6 απαντήσεις) είναι ισάριθμος με τον αριθμό των μαθητών που είναι στο 4^ο επίπεδο (16 απαντήσεις). Οπότε συμπεραίνουμε ότι όσοι μαθητές δεν ξέρουν να εκτελούν την επανάληψη μαζί με αυτούς

που δεν ξέρουν τι εμφανίζει είναι ισάριθμοι με τους μαθητές που γνωρίζουν τι εμφανίζει η επανάληψη σε οριακές τιμές.

Στο τρίτο ερώτημα που απαιτεί ερμηνεία του προγράμματος από τους μαθητές είναι αξιοσημείωτο ότι οι μαθητές κατά πλειοψηφία (21 απαντήσεις) έχουν κατανοήσει ότι το συγκεκριμένο πρόγραμμα υπολογίζει ένα άθροισμα μέσα σε μια επανάληψη (4^ο επίπεδο). Επίσης το ένα τρίτο των μαθητών (1^ο και 2^ο επίπεδο), αθροιστικά, δεν γνωρίζει τι υπολογίζει το πρόγραμμα ή θεωρεί ότι υπολογίζει τον αριθμό των επαναλήψεων.

Συνολικά ως προς το επίπεδο Κατανοώ για την 2η δραστηριότητα έχουμε τα εξής αποτελέσματα:

- Το 30,3% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να ερμηνεύσουν τη λειτουργία ενός προγράμματος, να το εκτελέσουν σωστά και να ελέγξουν τις οριακές τιμές σε μια επανάληψη. Από αυτούς το 11,1% δεν έχει τη δυνατότητα να εκτελέσει το πρόγραμμα, το 10,1% δεν μπορεί να ελέγξει τις οριακές τιμές στην επανάληψη και 9,1% δε μπορεί να ερμηνεύσει το πρόγραμμα.

- Το 12,1% των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να εκτελέσουν, να ερμηνεύσουν και να ελέγξουν στοιχειώδεις και πολύ απλές λειτουργίες ενός προγράμματος. Το 4% εκτελεί την επανάληψη μόνο μία φορά, το 6,1% δεν κατανοεί τη λειτουργία της εμφάνισης τιμών, ενώ το 2% δεν είναι σε θέση να γνωρίζει τι ακριβώς υπολογίζει το πρόγραμμα.

- Το 12,1% απαντά στο 3^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές έχουν τη δυνατότητα να προσεγγίσουν σε ικανοποιητικό βαθμό, την εκτέλεση, την ερμηνεία ενός προγράμματος καθώς και των έλεγχο των οριακών τιμών μιας επανάληψης. Το 10,1% εκτελεί και υπολογίζει σωστά το άθροισμα τιμών σε μια επανάληψη χωρίς να μπορεί να βρίσκει τη συνθήκη όταν γνωρίζει το άθροισμα, επίσης το 1%, γνωρίζει τι υπολογίζει το πρόγραμμα αλλά δεν μπορεί να αντιστοιχίσει τις μεταβλητές με τις αντίστοιχες έννοιες και το 1% δυσκολεύεται να ερμηνεύσει σωστά το πρόγραμμα.

- Το 45,5% των μαθητών αποδίδει με πληρότητα τα επιμέρους ερωτήματα και οι απαντήσεις τους τοποθετούνται στο 4^ο επίπεδο ΙΑΠ. Από αυτούς το 8,1 % εκτελούν σωστά το πρόγραμμα, το 16,2 % μπορούν να ελέγξουν με πληρότητα τις οριακές τιμές με τις οποίες εκτελείται το πρόγραμμα καθώς επίσης το 21,2% κάνει σωστή ερμηνεία του προγράμματος.

- Το 5^ο επίπεδο δεν ορίζεται για τη συγκεκριμένη δραστηριότητα.

Σύγκριση τρόπων αξιολόγησης στο επίπεδο Κατανοώ

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 30,3% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 62,6% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν να εκτελέσουν μια δομή επανάληψης ΟΣΟ ώστε να υπολογίσουν το άθροισμα

προκαθορισμένων τιμών, επίσης να γνωρίζουν το άθροισμα και να υπολογίζουν τη συνθήκη ελέγχου, να ελέγξουν οριακές τιμές εκτέλεσης μιας επανάληψης, καθώς και να ερμηνεύουν το πρόγραμμα, διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης. Για το προδρομικό επίπεδο η κατανόηση γνώσεων, υποδιπλασιάζεται με την αξιολόγηση ΙΑΠ συγκριτικά με την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 12,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 4% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις γνώσεις για την εκτέλεση ενός προγράμματος και τις οριακές τιμές και την ερμηνεία του προγράμματος οι τριπλάσιοι σε ποσοστό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση. Στο επίπεδο επιμέρους κατανόησης η κατανόηση γνώσεων είναι πολλαπλάσια με την αξιολόγηση ΙΑΠ από ότι με την παραδοσιακή αξιολόγηση.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 12,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 6,1% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που μπορούν να κατανοήσουν γνώσεις προγραμματισμού σε ικανοποιητικό βαθμό που αφορούν την εκτέλεση επαναλήψεων, τις οριακές τιμές ελέγχου μιας συνθήκης καθώς και την ερμηνεία του προγράμματος υπερδιπλασιάζονται με την αξιολόγηση ΙΑΠ.

- Στο 4^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 45,5% με την ΙΑΠ και 3% με την παραδοσιακή αξιολόγηση. Ενώ το 5^ο επίπεδο δεν ορίζεται με την αξιολόγηση ΙΑΠ, το ποσοστό για το επίπεδο αυτό με την παραδοσιακή αξιολόγηση είναι 24,2%. Στην συγκεκριμένη δραστηριότητα που αφορά την ερμηνεία και την κατανόηση γνώσεων στις δομές επανάληψης διαπιστώνεται ότι οι μισοί μαθητές έχουν ένα πάρα πολύ καλό επίπεδο γνώσεων ώστε να απαντούν σωστά στα ερωτήματα της δραστηριότητας. Ενώ το ένα τέταρτο των μαθητών σύμφωνα με την παραδοσιακή αξιολόγηση έχει κατανοήσει απόλυτα την εκτέλεση, τους περιορισμούς και την ερμηνεία μιας επαναληπτικής δομής.

5.3 Αποτελέσματα ως προς το Επίπεδο Εφαρμόζω

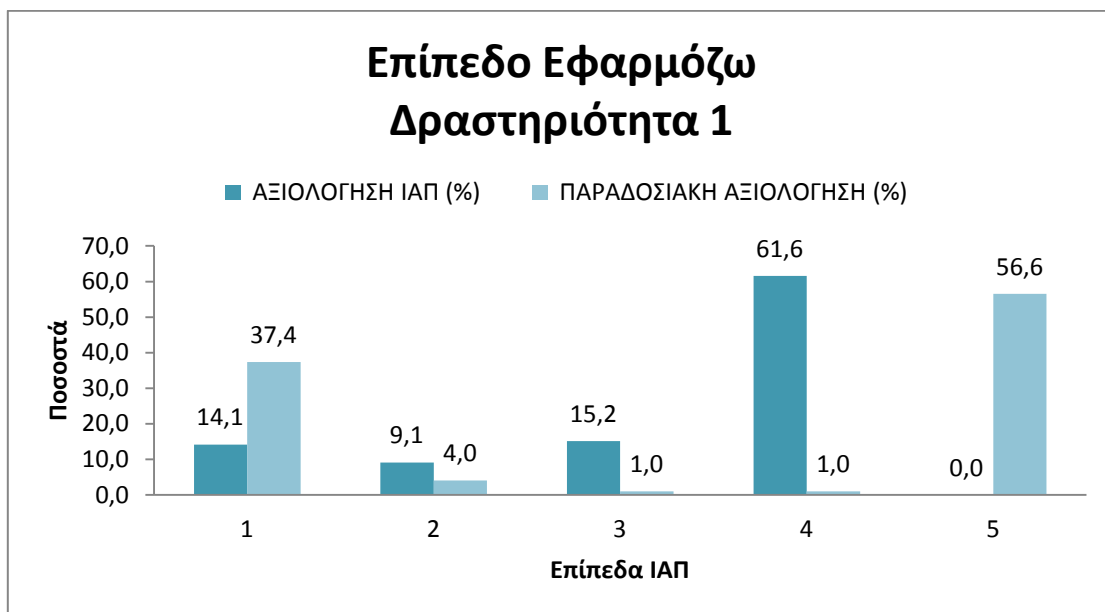
Στο επίπεδο εφαρμόζω ο μαθητής είναι σε θέση να επιλέξει και να χρησιμοποιήσει δεδομένα και μεθόδους για την επίλυση ενός συγκεκριμένου προβλήματος. Το επίπεδο «Εφαρμόζω», ορίζεται ως εκτέλεση ή χρήση μιας διαδικασίας σε μια δεδομένη κατάσταση. Η κατηγορία αυτή περιλαμβάνει Εκτέλεση και Εφαρμογή. Η διαδικασία και ο αλγόριθμος ή το πρόγραμμα είναι γνωστά στον εκπαιδευόμενο, και τα δύο εφαρμόζονται σε ένα πρόβλημα που είναι γνωστό, αλλά που δεν έχει λυθεί προηγουμένως στο ίδιο πλαίσιο ή με τα ίδια δεδομένα ή εφαρμόζονται σε άγνωστο πρόγραμμα.

Πίνακας 5.5: ΔΡ1 αξιολόγηση ΙΑΠ επίπεδο Εφαρμοζώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΕΦΑΡΜΟΖΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ1			ΑΞΙΟΛΟΓΗΣΗ
	Ερώτημα	Ερώτημα	Ερώτημα	ΙΑΠ (%)
	A	B	Γ	
	(αριθμός)	(αριθμός)	(αριθμός)	
1	2	4	8	14,1
2	2	4	3	9,1
3	8	6	1	15,2
4	21	19	21	61,6
5	0	0	0	0,0

Πίνακας 5.6: ΔΡ1 αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Εφαρμοζώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΕΦΑΡΜΟΖΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 1			ΠΑΡΑΔΟΣΙΑΚΗ
	Ερώτημα	Ερώτημα	Ερώτημα	ΑΞΙΟΛΟΓΗΣΗ (%)
	A	B	Γ	
	(αριθμός)	(αριθμός)	(αριθμός)	
1	12	11	14	37,4
2	0	2	2	4,0
3	1	0	0	1,0
4	0	0	1	1,0
5	20	20	16	56,6



Σχήμα 5.4: ΔΡ1 σύγκριση αποτελεσμάτων επίπεδο Εφαρμόζω

Αξιολόγηση γνώσεων μαθητών σύμφωνα με την ΙΑΠ

Για την πρώτη δραστηριότητα στον πίνακα 5.7 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.8 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.4 απεικονίζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Εφαρμόζω της αναθεωρημένης ταξινομίας Bloom και με τους δύο τρόπους αξιολόγησης.

Η πρώτη δραστηριότητα που δόθηκε στους μαθητές για το επίπεδο Εφαρμόζω υπολόγιζε τον ελάχιστο χρόνο και ποιος αθλητής τον είχε από 20 τιμές (χρόνους) που διαβάζονται από το πληκτρολόγιο και είχε τρία ερωτήματα. Στο πρώτο ερώτημα θα έπρεπε οι μαθητές να γράψουν την κατάλληλη δομή ώστε να γίνονται 20 επαναλήψεις. Στο δεύτερο ερώτημα οι μαθητές έπρεπε να γράψουν την εντολή εκχώρησης στη μεταβλητή min, ενώ στο τρίτο ερώτημα έπρεπε να γράψουν την εντολή εκχώρησης στο όνομα του αθλητή που είχε τον ελάχιστο χρόνο.

Από τον πίνακα 5.7 διαφαίνεται ότι η πλειοψηφία των μαθητών αναγνωρίζει ότι έπρεπε να γράψει δομή επανάληψης για με 10 επαναλήψεις. Παρατηρούμε ότι 21 απαντήσεις είναι στο 4^ο επίπεδο ΙΑΠ καθώς και 8 απαντήσεις ανήκουν στο 3^ο επίπεδο ΙΑΠ. Οι μαθητές κατά πλειοψηφία αντιλαμβάνονται ότι το πρόγραμμα απαιτεί επανάληψη και είναι σε θέση να εφαρμόσουν τη σωστή δομή επανάληψης για γνωστό πλήθος επαναλήψεων

Στο δεύτερο ερώτημα όπως αναφέρθηκε σε προηγούμενη παράγραφο οι μαθητές έπρεπε να εκχωρήσουν στην μεταβλητή min τον χρόνο του κάθε αθλητή. Από τον πίνακα 5.7 επίσης

διαπιστώνεται ότι οι περισσότεροι μαθητές μπορούν να γράψουν σωστά την εντολή αυτή επειδή την έχουν διδαχθεί και εφαρμόσει πολλές φορές μέσα στην τάξη σε πολλά προβλήματα. Παρατηρούμε ότι 19 από τους 33 μαθητές απαντούν στο 4^ο επίπεδο, ενώ 6 μαθητές στο 3^ο επίπεδο. Στο τρίτο ερώτημα έπρεπε οι μαθητές να εκχωρήσουν στην μεταβλητή που υπολογίζει το όνομα του αθλητή με τον ελάχιστο χρόνο την κατάλληλη μεταβλητή. Σύμφωνα με τον πίνακα 7 οι περισσότεροι μαθητές μπορούν να γράφουν σωστά αυτή την εντολή και οι απαντήσεις τους ανήκουν στο 4^ο επίπεδο. Αξιοσημείωτο είναι το γεγονός ότι 8 μαθητές απαντούν στο 1^ο επίπεδο δηλαδή δεν είναι σε θέση να υπολογίσουν ποιος αθλητής έχει τον ελάχιστο χρόνο.

Συνολικά ως προς το επίπεδο Εφαρμόζω για την πρώτη δραστηριότητα έχουμε τα εξής αποτελέσματα:

- Το 14,1% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να εφαρμόσουν τη διαδικασία υπολογισμού ελάχιστης τιμής καθώς και το όνομα αυτού που την έχει σε προκαθορισμένο αριθμό από τυχαίες τιμές. Από αυτούς το 2% δεν μπορεί να υλοποιήσει προκαθορισμένο αριθμό επαναλήψεων με τη δομή ΓΙΑ, το 6% δεν μπορεί να εφαρμόσει την εντολή εκχώρησης τιμής για την ελάχιστη τιμή και το 8,1% δε μπορεί να εφαρμόσει την εντολή εκχώρησης τιμής στο όνομα του ελαχίστου.

- Το 9,1 % των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να εκτελέσουν και να εφαρμόσουν στοιχειώδεις και πολύ απλές λειτουργίες ενός προγράμματος. Το 2% των μαθητών δεν υλοποιεί σωστά την επανάληψη, χρησιμοποιεί λάθος μεταβλητές και τις αντίστοιχες οριακές τιμές, το 4,1% εκχωρεί σε λάθος μεταβλητή την ελάχιστη τιμή, ενώ το 3% εκχωρεί στη σωστή μεταβλητή αριθμητικές τιμές ενώ θα έπρεπε τιμή τύπου χαρακτήρα.

- Το 15,2% των μαθητών απαντά στο 3^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές έχουν τη δυνατότητα να προσεγγίσουν σε ικανοποιητικό βαθμό, την εφαρμογή μιας επαναληπτικής διαδικασίας, καθώς και να χρησιμοποιήσουν τις εντολές εκχώρησης. Το 8,1% δεν υλοποιεί σωστά την επανάληψη με λάθος μεταβλητές ή εναλλακτικά με λάθος οριακές τιμές, επίσης το 6,1% γράφει τη σωστή εντολή εκχώρησης αλλά κάνει συντακτικά λάθη ενώ το 1 % εκχωρεί στη σωστή μεταβλητή λανθασμένη αλφαριθμητική τιμή.

- Το 61,6% των μαθητών εφαρμόζουν με πληρότητα τη διαδικασία υπολογισμού ελαχίστου και οι απαντήσεις τους τοποθετούνται στο 4^ο επίπεδο ΙΑΠ. Από αυτούς το 21,2% υλοποιεί σωστά 20 επαναλήψεις, το 19,2% εκχωρεί την σωστή μεταβλητή στο ελάχιστο καθώς επίσης το 21,2% εκχωρεί τη σωστή μεταβλητή στο όνομα του ελαχίστου χωρίς συντακτικά ή άλλα λάθη.

- Το 5^ο επίπεδο δεν ορίζεται για τη συγκεκριμένη δραστηριότητα.

Σύγκριση τρόπων αξιολόγησης στο επίπεδο Εφαρμόζω

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 14.1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 37.4% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν να εφαρμόσουν τη διαδικασία υπολογισμού της ελάχιστης τιμής και του ατόμου που την έχει για προκαθορισμένο αριθμό από τυχαίες τιμές διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης. Για το προδρομικό επίπεδο η εφαρμογή διαδικασιών, υποπολλαπλασιάζεται με την αξιολόγηση ΙΑΠ συγκριτικά με την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 9,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 4% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις γνώσεις για την εφαρμογή του υπολογισμού μιας ελάχιστης τιμής και της θέσης της υπερδιπλασιάζονται σε ποσοστό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση. Στο επίπεδο επιμέρους κατανόησης η εφαρμογή γνώσεων είναι πολλαπλάσια με την αξιολόγηση ΙΑΠ από ότι με την παραδοσιακή αξιολόγηση.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 15,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 1% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που μπορούν να εφαρμόζουν γνώσεις προγραμματισμού, σε ικανοποιητικό βαθμό, που αφορούν την εύρεση του ελαχίστου και της θέσης του μέσα από μια επαναληπτική διαδικασία 15πλασιάζεται με την αξιολόγηση ΙΑΠ. Στο επίπεδο προσεγγιστικής κατανόησης η εφαρμογή γνώσεων πολλαπλασιάζεται με την ΙΑΠ.

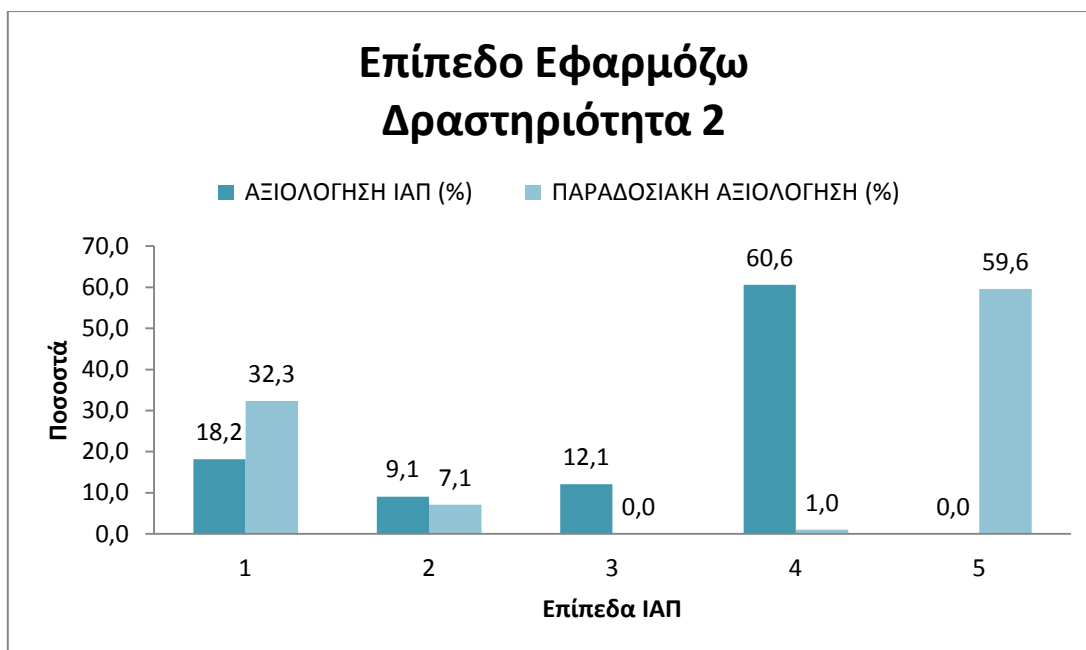
- Στο 4^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 61,6% με την ΙΑΠ και 1% με την παραδοσιακή αξιολόγηση. Ενώ το 5^ο επίπεδο δεν ορίζεται με την αξιολόγηση ΙΑΠ αυτό το ποσοστό στην παραδοσιακή αξιολόγηση είναι 56,6%. Στην συγκεκριμένη δραστηριότητα που αφορά την εφαρμογή γνώσεων στις δομές επανάληψης, στον υπολογισμό ελάχιστης τιμής, καθώς και στις εντολές εκχώρησης διαπιστώνεται ότι περισσότεροι από τους μισούς μαθητές έχουν ένα πάρα πολύ καλό επίπεδο γνώσεων ώστε να απαντούν σωστά στα ερωτήματα της δραστηριότητας. Το ποσοστό των απαντήσεων με την παραδοσιακή αξιολόγηση στο 4^ο και 5^ο επίπεδο αθροιστικά τείνει να προσεγγίσει το αντίστοιχο ποσοστό στο 4^ο επίπεδο με την ΙΑΠ.

Πίνακας 5.7: ΔΡ2 αξιολόγηση ΙΑΠ επίπεδο εφαρμοζω

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΕΦΑΡΜΟΖΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2			ΑΞΙΟΛΟΓΗΣΗ
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	ΙΑΠ (%)
	1	7	6	5
2	2	4	3	9,1
3	2	7	3	12,1
4	22	16	22	60,6
5	0	0	0	0,0

Πίνακας 5.8: ΔΡ2 αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Εφαρμοζω

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΕΦΑΡΜΟΖΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2			ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ (%)
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	
	1	8	12	12
2	3	4	0	7,1
3	0	0	0	0,0
4	0	1	0	1,0
5	22	16	21	59,6



Σχήμα 5.5: ΔΡ2 σύγκριση αποτελεσμάτων επίπεδο Εφαρμόζω

Αξιολόγηση γνώσεων με ΙΑΠ

Στον πίνακα 5.9 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.10 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.5 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Εφαρμόζω της αναθεωρημένης ταξινόμιας Bloom και με τους 2 τρόπους αξιολόγησης.

Η δεύτερη δραστηριότητα που δόθηκε στους μαθητές για το επίπεδο Εφαρμόζω υπολόγιζε το άθροισμα και το πλήθος από τυχαίες θετικές τιμές που δίνονται από το πληκτρολόγιο και είχε τρία ερωτήματα. Στο πρώτο ερώτημα οι μαθητές έπρεπε να αρχικοποιήσουν το άθροισμα πριν την επανάληψη, στο δεύτερο ερώτημα να γράψουν δομή επανάληψης με ΟΣΟ και στο τρίτο ερώτημα να γράψουν την εντολή εκχώρησης υπολογισμού πλήθους.

Από τον πίνακα 5.9 διαπιστώνεται ότι οι μαθητές κατά πλειοψηφία (22 απαντήσεις) αναγνωρίζουν ότι πρέπει να αρχικοποιήσουν το άθροισμα με την τιμή μηδέν πριν την επανάληψη. Οι περισσότεροι μαθητές ως προς το ερώτημα αυτό απαντούν στο 4^ο επίπεδο. Για το δεύτερο ερώτημα οι περισσότεροι από τους μισούς (16 απαντήσεις) ανήκουν στο 4^ο επίπεδο, είναι δηλαδή σε θέση να εφαρμόσουν μια διαδικασία επανάληψης για τυχαίες θετικές τιμές. Ενώ για το τρίτο ερώτημα ισχύουν τα ίδια με το πρώτο, ότι οι περισσότεροι μαθητές (22 απαντήσεις) ανήκουν στο 4^ο επίπεδο και μπορούν να εφαρμόσουν την εντολή εκχώρησης που υπολογίζει το πλήθος τιμών που δίνονται από το πληκτρολόγιο. Παρατηρούμε ότι οι μαθητές χρησιμοποιούν εντολές εκχώρησης

(αρχικοποίηση αθροίσματος, υπολογισμός πλήθους) με μεγαλύτερη ευκολία και άνεση σε σχέση με τις δομές επανάληψης.

Συνολικά ως προς το επίπεδο Εφαρμόζω για την δεύτερη δραστηριότητα έχουμε τα εξής αποτελέσματα:

- Το 18,2% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να εφαρμόσουν τη διαδικασία υπολογισμού αθροίσματος και πλήθους για τυχαίες θετικές τιμές. Από αυτούς το 7,1% δεν μπορεί να εφαρμόσει την εντολή αρχικοποίησης αθροίσματος στη σωστή θέση στο πρόγραμμα, το 6,1% δεν μπορεί υλοποιήσει επανάληψη με ΟΣΟ για θετικούς αριθμούς και το 5% δε μπορεί να εφαρμόσει την εντολή εκχώρησης τιμής που υπολογίζει πλήθος.

- Το 9,1% των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να εκτελέσουν και να εφαρμόσουν στοιχειώδεις και πολύ απλές λειτουργίες ενός προγράμματος. Το 2% των μαθητών εκχωρεί στη σωστή μεταβλητή αλφαριθμητικές τιμές, το 4,1% δεν υλοποιεί σωστά την επανάληψη με λάθος μεταβλητές και τις αντίστοιχες οριακές τιμές στη συνθήκη, ενώ το 3% εκχωρεί στη σωστή μεταβλητή (το πλήθος) μια έκφραση εντελώς λανθασμένα.

- Το 12,1% των μαθητών απαντά στο 3^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές έχουν τη δυνατότητα να προσεγγίσουν σε ικανοποιητικό βαθμό, την εφαρμογή μιας επαναληπτικής διαδικασίας, καθώς και να χρησιμοποιήσουν τις εντολές εκχώρησης. Το 2% εκχωρεί στη σωστή μεταβλητή (άθροισμα) μια λανθασμένη αρχική τιμή, επίσης το 7,1% υλοποιεί την επανάληψη με λάθος οριακές τιμές ή με λάθος μεταβλητή ενώ το 3% γράφει την εντολή εκχώρησης για τη σωστή μεταβλητή, το πλήθος, η οποία περιέχει στο δεξιό μέλος κάποια λογικά λάθη.

- Το 60,6% των μαθητών εφαρμόζουν με πληρότητα τη διαδικασία υπολογισμού αθροίσματος και πλήθους με αποτέλεσμα οι απαντήσεις τους να τοποθετούνται στο 4^ο επίπεδο ΙΑΠ. Από αυτούς το 22,2% αρχικοποιεί σωστά το άθροισμα πριν την επανάληψη, το 16,2% υλοποιεί επανάληψη με ΟΣΟ για τυχαίους θετικούς αριθμούς καθώς επίσης το 22,2% υπολογίζει σωστά το πλήθος με την κατάλληλη εντολή εκχώρησης.

- Το 5^ο επίπεδο δεν ορίζεται για τη συγκεκριμένη δραστηριότητα.

Σύγκριση τρόπων αξιολόγησης στο επίπεδο Εφαρμόζω

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 18,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 32,3% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν να εφαρμόσουν τη διαδικασία υπολογισμού αθροίσματος και πλήθους για τυχαίες θετικές τιμές διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης. Για το προδρομικό επίπεδο η εφαρμογή διαδικασιών, υποπολλαπλασιάζεται με την αξιολόγηση ΙΑΠ συγκριτικά με την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 9,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 4,1% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις γνώσεις για την εφαρμογή του υπολογισμού αθροίσματος και πλήθους τυχαίων θετικών τιμών υπερδιπλασιάζονται σε ποσοστό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση. Στο επίπεδο επιμέρους κατανόησης η εφαρμογή γνώσεων είναι πολλαπλάσια με την αξιολόγηση ΙΑΠ από ότι με την παραδοσιακή αξιολόγηση. Επομένως στο επίπεδο αυτό δίνει καλύτερα αποτελέσματα η ΙΑΠ.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 12,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ κανένας μαθητής δεν απαντά με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές μπορούν να εφαρμόζουν γνώσεις προγραμματισμού, σε ικανοποιητικό βαθμό, που αφορούν χρήση μετρητή και αθροιστή μέσα σε μια επαναληπτική διαδικασία 12πλασιάζεται με την αξιολόγηση ΙΑΠ. Στο επίπεδο προσεγγιστικής κατανόησης η εφαρμογή γνώσεων πολλαπλασιάζεται με την ΙΑΠ.

- Στο 4^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 60,6% με την ΙΑΠ και 1% με την παραδοσιακή αξιολόγηση. Ενώ το 5^ο επίπεδο δεν ορίζεται με την αξιολόγηση ΙΑΠ το αντίστοιχο ποσοστό στην παραδοσιακή αξιολόγηση είναι 59,6%. Στην συγκεκριμένη δραστηριότητα που αφορά την εφαρμογή γνώσεων στις δομές επανάληψης, στον υπολογισμό αθροιστή και μετρητή τυχαίων τιμών που διαβάζονται από το πληκτρολόγιο διαπιστώνεται ότι περισσότεροι από τους μισούς μαθητές έχουν ένα πάρα πολύ καλό επίπεδο γνώσεων ώστε να απαντούν σωστά στα ερωτήματα της δραστηριότητας. Το ποσοστό των απαντήσεων με την παραδοσιακή αξιολόγηση στο 4^ο και 5^ο επίπεδο αθροιστικά συμπίπτει με το αντίστοιχο ποσοστό στο 4^ο επίπεδο με την ΙΑΠ.

5.4 Αποτελέσματα ως προς το Επίπεδο Αναλύω

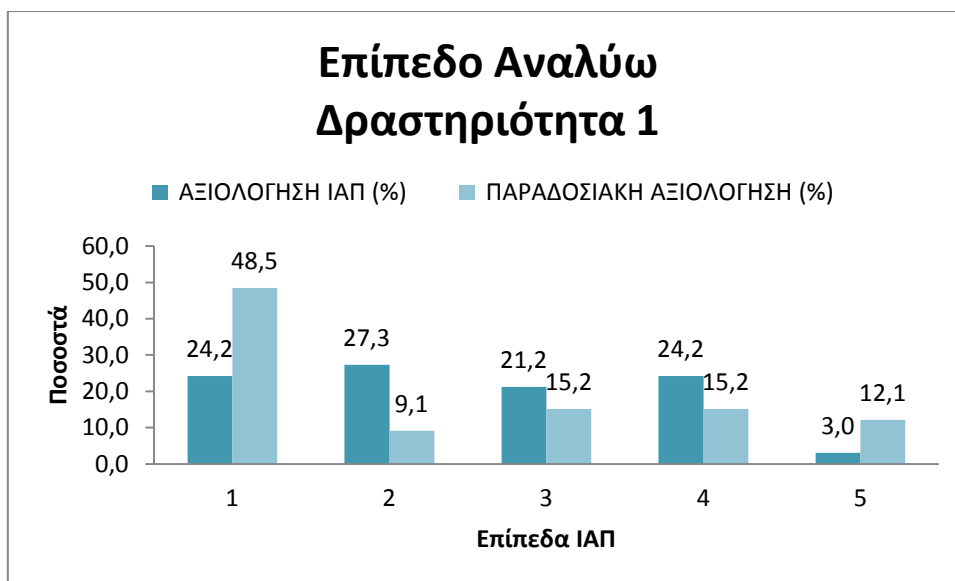
Το επίπεδο «Αναλύω», ορίζεται ως η διάσπαση του προβλήματος στα συστατικά μέρη του και ο καθορισμός του τρόπου με τον οποίο τα μέρη συνδέονται μεταξύ. Αυτή η κατηγορία περιλαμβάνει τη διαφοροποίηση, την οργάνωση και τον προσδιορισμό. Να διασπαστεί μια δραστηριότητα προγραμματισμού στα συστατικά μέρη της (εντολές εκχώρησης, δομές επιλογής, δομές επανάληψης). Να οργανωθούν τα συστατικά μέρη για την επίτευξη ενός γενικού στόχου, να προσδιοριστούν τα σημαντικά ή τα μη σημαντικά στοιχεία του προβλήματος. Ο μαθητής χρειάζεται να μπορεί να διακρίνει, να ταξινομεί και να συσχετίζει τις πληροφορίες που του δίνονται και να αποσυνθέτει ένα πρόβλημα στα μέρη του.

Πίνακας 5.9: ΔΡ1 αξιολόγηση ΙΑΠ επίπεδο Αναλύω

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΑΝΑΛΥΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 1 (αριθμός)	ΑΞΙΟΛΟΓΗΣΗ ΙΑΠ (%)
1	8	24,2
2	9	27,3
3	7	21,2
4	8	24,2
5	1	3,0

Πίνακας 5.10: ΔΡ1 αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Αναλύω

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΑΝΑΛΥΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 1 (αριθμός)	ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ (%)
1	16	48,5
2	3	9,1
3	5	15,2
4	5	15,2
5	4	12,1



Σχήμα 5.6: ΔΡ1 σύγκριση αποτελεσμάτων επίπεδο Αναλύω

Αξιολόγηση γνώσεων με ΙΑΠ

Στον πίνακα 5.11 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.12 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.6 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Αναλύω της αναθεωρημένης ταξινομίας Bloom και με τους δύο τρόπους αξιολόγησης.

Στην πρώτη δραστηριότητα που δόθηκε στους μαθητές τους ζητήθηκε να γράψουν πρόγραμμα που θα διαβάζει θετικούς ακέραιους αριθμούς και θα εμφανίζει το τελευταίο ψηφίο τους, το πλήθος των άρτιων και το ποσοστό των περιττών.

Για την πρώτη δραστηριότητα ως προς το επίπεδο Αναλύω έχουμε τα εξής αποτελέσματα:

- Το 24,2% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να διασπάσουν το πρόβλημα που τους δόθηκε σε υποπροβλήματα τα οποία απαιτούσαν χρήση δομής επανάληψης με άγνωστο αριθμό επαναλήψεων, χρήση δομής επιλογής, υπολογισμό πλήθους υπό όρους.
- Το 27,3% των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να εκτελέσουν και να αναλύσουν ένα πρόβλημα σε ελάχιστες απλές λειτουργίες. Δεν είναι σε θέση να χρησιμοποιήσουν σωστά τη δομή επανάληψης, όμως μπορούν να υπολογίζουν σωστά το τελευταίο ψηφίο ενός ακέραιου αριθμού. Επίσης έχουν τη δυνατότητα να διαχειρίζονται τον υπολογισμό του πλήθους των άρτιων με ορισμένα λάθη ενώ αντίθετα δεν είναι σε θέση να υπολογίσουν το ποσοστό των περιττών επειδή είναι πιο σύνθετη διαδικασία.

- Το 21,2% των μαθητών απαντά στο 3^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές έχουν τη δυνατότητα να προσεγγίσουν σε ικανοποιητικό βαθμό, την ανάλυση ενός προβλήματος που απαιτεί μια επαναληπτική διαδικασία με δομή επιλογής, πλήθος, άθροισμα και ποσοστό. Μπορούν να υλοποιούν σωστά την επανάληψη, υπολογίζουν σωστά το τελευταίο ψηφίο, αλλά αντιμετωπίζουν δυσκολίες κατά τον υπολογισμό των άρτιων ή των περιττών ή το ποσοστό των περιττών τα οποία οφείλονται στη χρήση της δομής επιλογής σε συνδυασμό με τους μετρητές. Επίσης όταν δεν υλοποιούν σωστά την επανάληψη αλλά υπολογίζουν σωστά τα πλήθη και το ποσοστό

- Το 24,2% των μαθητών απαντά στο 4^ο επίπεδο ΙΑΠ, έχουν τη δυνατότητα διάσπασης του προβλήματος σε υποπροβλήματα με πληρότητα. Οι μαθητές αυτοί υλοποιούν σωστά τη διαδικασία επανάληψης για άγνωστο αριθμό επαναλήψεων, υπολογίζουν σωστά το τελευταίο ψηφίο ενός ακέραιου αριθμού κάνοντας εφαρμογή του τελεστή mod, υπολογίζουν το πλήθος των άρτιων και το ποσοστό των περιττών συνδυάζοντας τη δομή επιλογής με την κατάλληλη συνθήκη και τους μετρητές με αποτέλεσμα οι απαντήσεις τους να τοποθετούνται στο 4^ο επίπεδο ΙΑΠ.

- Το 3% των μαθητών απαντά στο 5^ο επίπεδο. Οι μαθητές αυτοί διαθέτουν τη μέγιστη ικανότητα ανάλυσης του προβλήματος υπολογίζοντας σωστά όλα τα ζητούμενα δίνοντας ως απάντηση τη βέλτιστη λύση. Υλοποιούν σωστά την επανάληψη, υπολογίζουν σωστά το τελευταίο ψηφίο, το πλήθος των άρτιων και το ποσοστό των περιττών κάνοντας έλεγχο αν το πλήθος των επαναλήψεων είναι διάφορο ή ίσο με το μηδέν η οποία αποτελεί και τη καλύτερη λύση για το συγκεκριμένο πρόβλημα.

Σύγκριση τρόπων αξιολόγησης για το επίπεδο Αναλύω

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 24,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 48,5% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν να διασπάσουν το πρόβλημα που τους δόθηκε σε υποπροβλήματα τα οποία απαιτούσαν χρήση δομής επανάληψης με άγνωστο αριθμό επαναλήψεων, χρήση δομής επιλογής, υπολογισμό πλήθους υπό όρους, διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης. Για το προδρομικό επίπεδο η ικανότητα ανάλυσης ενός προβλήματος υποδιπλασιάζεται με την αξιολόγηση ΙΑΠ συγκριτικά με την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 27,3% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 9,1% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις ικανότητες να αναλύουν ένα πρόβλημα που απαιτεί μια επαναληπτική διαδικασία με δομή επιλογής, πλήθος και ποσοστό. τριπλασιάζεται σε ποσοστό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση. Στο επίπεδο επιμέρους κατανόησης η ικανότητα διάσπασης είναι πολλαπλάσια με την αξιολόγηση ΙΑΠ από ότι με την παραδοσιακή αξιολόγηση.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 21,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 15,2% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές είναι σε θέση να διασπάσουν ένα πρόβλημα σε επιμέρους προβλήματα καθώς μπορούν να υλοποιούν σωστά την επανάληψη και να υπολογίζουν σωστά το τελευταίο ψηφίο, ενώ δυσκολεύονται κατά τον υπολογισμό των άρτιων ή των περιττών ή το ποσοστό των περιττών τα οποία απαιτούν σε μεγαλύτερο βαθμό αναλυτικές ικανότητες. Στο επίπεδο προσεγγιστικής κατανόησης η ικανότητα ανάλυσης προβλήματος αυξάνεται με την ΙΑΠ.

- Στο 4^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 24,2% με την ΙΑΠ και 15,2% με την παραδοσιακή αξιολόγηση. Στην συγκεκριμένη δραστηριότητα που απαιτεί δυνατότητα διάσπασης του προβλήματος σε υποπροβλήματα με πληρότητα οι μαθητές απαντούν σωστά στα ερωτήματα της δραστηριότητας. Στο συνδυαστικό επίπεδο η ικανότητα ανάλυσης προβλήματος αυξάνεται με την ΙΑΠ συγκριτικά με την παραδοσιακή αξιολόγηση.

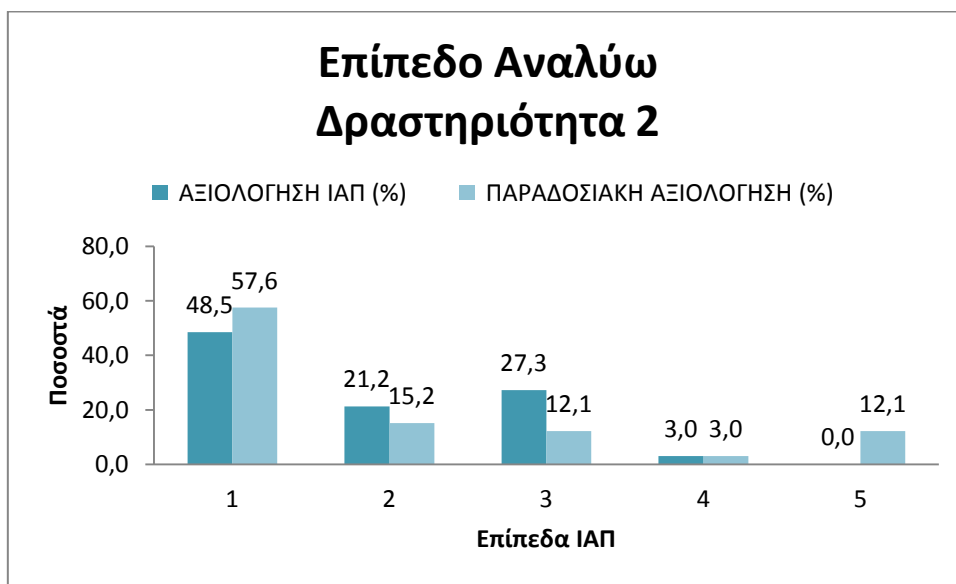
- Στο 5^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 3% με την αξιολόγηση ΙΑΠ ενώ το αντίστοιχο ποσοστό στην παραδοσιακή αξιολόγηση είναι 12,2%. Οι μαθητές που έχουν τη δυνατότητα να δώσουν τη βέλτιστη λύση στο πρόβλημα υποπολλαπλασιάζονται με την αξιολόγηση ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση.

Πίνακας 5.11: ΔΡ2 αξιολόγηση ΙΑΠ επίπεδο Αναλύω

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΑΝΑΛΥΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2 (αριθμός)	ΑΞΙΟΛΟΓΗΣΗ ΙΑΠ (%)
1	16	48,5
2	7	21,2
3	9	27,3
4	1	3,0
5	0	0,0

Πίνακας 5.12: ΔΡ2 αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Αναλύω

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΑΝΑΛΥΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2 (αριθμός)	ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ (%)
1	19	57,6
2	5	15,2
3	4	12,1
4	1	3,0
5	4	12,1



Σχήμα 5.7: ΔΡ2 σύγκριση αποτελεσμάτων επίπεδο Αναλύω

Αξιολόγηση γνώσεων με ΙΑΠ

Στον πίνακα 5.13 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.14 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.7 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Αναλύω της αναθεωρημένης ταξινομίας Bloom και με τους δύο τρόπους αξιολόγησης.

Η δεύτερη δραστηριότητα που δόθηκε στους μαθητές τους ζητούσε να γράψουν πρόγραμμα το οποίο έπρεπε να διαβάζει μέσα σε μια επαναληπτική διαδικασία με προκαθορισμένο αριθμό επαναλήψεων κάνοντας έλεγχο εγκυρότητας των δεδομένων εισόδου και να υπολογίζει πλήθη και ποσοστά υπό όρους.

Για την δεύτερη δραστηριότητα ως προς το επίπεδο Αναλύω έχουμε τα εξής αποτελέσματα:

- Το 48,5% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να διασπάσουν το πρόβλημα που τους δόθηκε σε υποπροβλήματα τα οποία απαιτούσαν χρήση δομής επανάληψης με γνωστό αριθμό επαναλήψεων, έλεγχο εγκυρότητας δεδομένων, χρήση δομής επιλογής, υπολογισμό πλήθους υπό όρους και υπολογισμό ποσοστού.

- Το 21,2% των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να εκτελέσουν και να αναλύσουν ένα πρόβλημα σε ελάχιστες απλές λειτουργίες. Χρησιμοποιούν δομή επανάληψης με ευχέρεια για γνωστό αριθμό επαναλήψεων, μπορούν να κάνουν έλεγχο εγκυρότητας για τα δεδομένα που διαβάζονται από το πληκτρολόγιο, και μπορούν να υπολογίζουν πλήθη αλλά δεν είναι σε θέση να συνδυάζουν δομές επιλογής που ελέγχουν διαφορετικές μεταβλητές.

- Το 27,3% των μαθητών απαντά στο 3^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές έχουν τη δυνατότητα να προσεγγίσουν σε ικανοποιητικό βαθμό, την ανάλυση ενός προβλήματος που απαιτεί μια επαναληπτική διαδικασία με δομή επιλογής, πλήθος, και ποσοστό. Μπορούν να υλοποιούν σωστά την επανάληψη και να υπολογίζουν πλήθη. Αντιμετωπίζουν δυσκολίες στον υπολογισμό ποσοστών, καθώς θα πρέπει να συνδυάσουν υπολογισμό μετρητών με περιορισμούς και το ποσοστό.

- Το 3% των μαθητών έχει τη δυνατότητα διάσπασης του προβλήματος σε υποπροβλήματα με πληρότητα. Οι μαθητές αυτοί υλοποιούν σωστά τη διαδικασία επανάληψης με τη δομή επανάληψης ΓΙΑ, δεν είναι σε θέση να κάνουν σωστό έλεγχο εγκυρότητας, υπολογίζουν σωστά πλήθη και ποσοστά συνδυάζοντας τη δομή επιλογής με την κατάλληλη συνθήκη και τους μετρητές με αποτέλεσμα οι απαντήσεις τους να τοποθετούνται στο 4^ο επίπεδο ΙΑΠ.

- Κανένας μαθητής δεν απαντά στο 5^ο επίπεδο. Δεν υπάρχουν μαθητές που να διαθέτουν ικανότητα ανάλυσης του προβλήματος σε μέγιστο βαθμό ώστε να μπορούν να υπολογίζουν σωστά όλα τα ζητούμενα και να δίνουν ως απάντηση τη βέλτιστη λύση.

Σύγκριση τρόπων αξιολόγησης στο επίπεδο Αναλύω

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 48,5% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 57,6 % με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν να διασπάσουν το πρόβλημα που τους δόθηκε σε υποπροβλήματα τα οποία απαιτούσαν χρήση δομής επανάληψης με γνωστό αριθμό επαναλήψεων, έλεγχο εγκυρότητας δεδομένων εισόδου, χρήση δομής επιλογής, υπολογισμό πλήθους υπό όρους και ποσοστού, δεν διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης. Για το προδρομικό επίπεδο η ικανότητα ανάλυσης ενός προβλήματος ελαττώνεται με την αξιολόγηση ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 21,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 15,2% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται

ότι οι μαθητές που έχουν στοιχειώδεις ικανότητες να αναλύουν ένα πρόβλημα που απαιτεί μια επαναληπτική διαδικασία με δομή επιλογής, πλήθος, και ποσοστό, αυξάνεται σε ποσοστό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 27,3% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 12,1% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές είναι σε θέση να διασπάσουν ένα πρόβλημα σε επιμέρους προβλήματα καθώς μπορούν να υλοποιούν σωστά την επανάληψη, τον έλεγχο των δεδομένων εισόδου που γίνεται με δομή επανάληψης και να υπολογίζουν σωστά πλήθη, ενώ δυσκολεύονται κατά τον υπολογισμό ποσοστού υπό όρους που απαιτεί σε μεγαλύτερο βαθμό αναλυτικές ικανότητες. Στο επίπεδο προσεγγιστικής κατανόησης η ικανότητα ανάλυσης προβλήματος τριπλασιάζεται με την ΙΑΠ.

- Στο 4^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 3% με την ΙΑΠ και 3% με την παραδοσιακή αξιολόγηση. Στην συγκεκριμένη δραστηριότητα που απαιτεί δυνατότητα διάσπασης του προβλήματος σε υποπροβλήματα με πληρότητα οι μαθητές απαντούν σωστά στα ερωτήματα της δραστηριότητας. Στο συνδυαστικό επίπεδο η ικανότητα ανάλυσης προβλήματος ταυτίζεται και στους δύο τρόπους αξιολόγησης.

- Στο 5^ο επίπεδο κανένας μαθητής δεν απαντά με την αξιολόγηση ΙΑΠ ενώ το αντίστοιχο ποσοστό στην παραδοσιακή αξιολόγηση είναι 12,1%. Δεν υπάρχουν μαθητές που να έχουν τη δυνατότητα να δώσουν τη βέλτιστη λύση στο πρόβλημα όπως ορίστηκε με την αξιολόγηση ΙΑΠ.

5.5 Αποτελέσματα ως προς το Επίπεδο Αξιολογώ

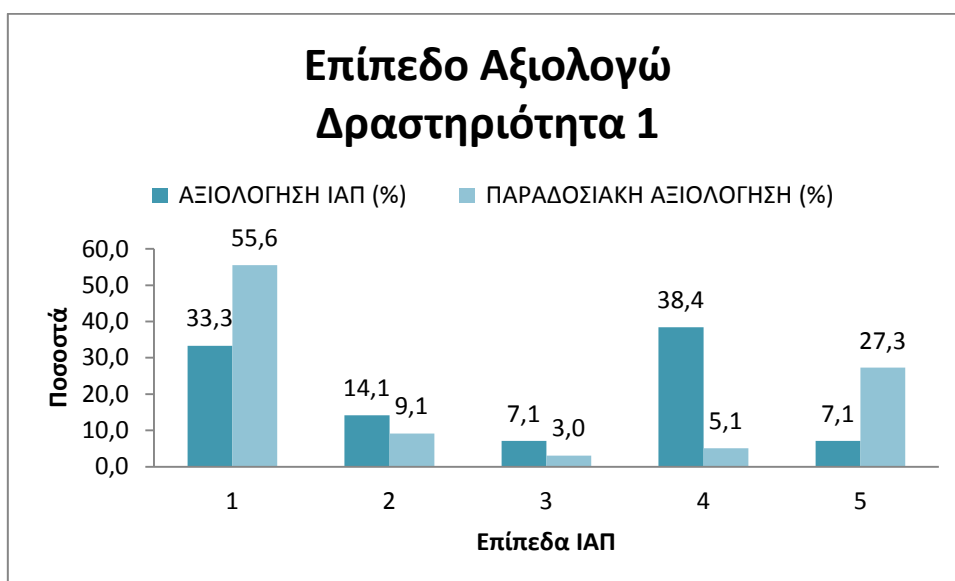
Το επίπεδο «Αξιολογώ», απαιτεί να γίνονται κρίσεις βάσει κριτηρίων και προτύπων. Στην αναθεωρημένη ταξινόμια, αυτή η κατηγορία περιλαμβάνει τον Έλεγχο και την Κριτική. Καθορίζει σε ποιο βαθμό ένα τμήμα προγράμματος ικανοποιεί τις απαιτήσεις του προβλήματος μέσω δοκιμών, κρίνει την απόδοση ενός προγράμματος βασισμένου σε πρότυπα κωδικοποίησης ή κριτήρια απόδοσης.

Πίνακας 5.13: ΔΡ1 αξιολόγηση ΙΑΠ επίπεδο Αξιολογώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΑΞΙΟΛΟΓΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 1			ΑΞΙΟΛΟΓΗΣΗ	
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	ΙΑΠ (%)	
	1	8	9	16	33,3
	2	6	2	6	14,1
3	2	2	3	7,1	
4	17	19	2	38,4	
5	0	1	6	7,1	

Πίνακας 5.14: ΔΡ1 αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Αξιολογώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΑΞΙΟΛΟΓΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 1			ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ	
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	(%)	
	1	17	12	26	55,6
	2	0	7	2	9,1
3	1	0	2	3,0	
4	3	1	1	5,1	
5	12	13	2	27,3	



Σχήμα 5.8: ΔΡ1 σύγκριση αποτελεσμάτων επίπεδο Αξιολογώ

Αξιολόγηση γνώσεων με ΙΑΠ

Στον πίνακα 5.15 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.16 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.8 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Αξιολογώ της αναθεωρημένης ταξινομίας Bloom και με τους δύο τρόπους αξιολόγησης.

Η πρώτη δραστηριότητα που δόθηκε στους μαθητές για το επίπεδο Αξιολογώ υπολόγιζε την απόλυτη τιμή μη μηδενικών αριθμών και είχε τρία ερωτήματα. Στο πρώτο ερώτημα οι μαθητές έπρεπε βρουν τι υπολογίζει το πρόγραμμα, στο δεύτερο ερώτημα να γράψουν την κατάλληλη εντολή ώστε η επανάληψη να εκτελείται για τυχαίους μη μηδενικούς αριθμούς και στο τρίτο ερώτημα να τροποποιήσουν το πρόγραμμα ώστε με λιγότερες εντολές να δίνει τα ίδια αποτελέσματα.

Σχετικά με το πρώτο ερώτημα από τον πίνακα 5.15 διαπιστώνεται ότι οι μισοί μαθητές είναι σε θέση να απαντήσουν σωστά, γράφοντας ότι το πρόγραμμα υπολογίζει την απόλυτη τιμή τυχαίων αριθμών. Επομένως οι απαντήσεις αυτών των μαθητών ανήκουν στο 4^ο επίπεδο (17 απαντήσεις). Σημαντικό είναι επίσης ότι 8 μαθητές (προδρομικό επίπεδο) δεν μπορούν να κρίνουν τι υπολογίζει το πρόγραμμα.

Για το δεύτερο ερώτημα η πλειοψηφία των μαθητών απαντά στο 4^ο επίπεδο (19 απαντήσεις), έχουν την κριτική ικανότητα να εντοπίσουν ποια εντολή λείπει από το πρόγραμμα ώστε να εκτελούνται σωστά οι επαναλήψεις. Επίσης 9 μαθητές δεν μπορούν να βρουν το λάθος, με αποτέλεσμα να

απαντούν στο προδρομικό επίπεδο. Τέλος, για το τρίτο ερώτημα οι μισές απαντήσεις των μαθητών ανήκουν στο 1^ο επίπεδο, δηλαδή οι μαθητές δεν είναι σε θέση να τροποποιήσουν το πρόγραμμα ώστε να δίνει τα σωστά αποτελέσματα με λιγότερες εντολές.

Συνολικά ως προς το επίπεδο Αξιολογώ για την πρώτη δραστηριότητα έχουμε τα εξής αποτελέσματα:

- Το 33,3% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να αναπτύξουν κριτική σκέψη και να απαντήσουν σε ερωτήματα για ένα πρόγραμμα που τους έχει δοθεί σχετικά με τον υπολογισμό της απόλυτης τιμής τυχαίων αριθμών. Από αυτούς το 9% δεν μπορεί να βρει τι υπολογίζει το δοσμένο πρόγραμμα, το 9,1% δεν μπορεί να εντοπίσει το λάθος που υπάρχει στο πρόγραμμα και το 16,2% δε μπορεί να τροποποιήσει το πρόγραμμα ώστε με λιγότερες εντολές να δίνει τα ίδια αποτελέσματα. Παρατηρούμε ότι το ερώτημα με τη μεγαλύτερη δυσκολία είναι το τρίτο. Κατά συνέπεια οι μαθητές δεν είναι σε θέση να όταν τους δοθεί μια λύση να κάνουν τις απαραίτητες αλλαγές ώστε να δώσουν τη βέλτιστη λύση.

- Το 14,1 % των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να αναπτύξουν κριτική σκέψη για στοιχειώδεις και πολύ απλές λειτουργίες ενός προγράμματος. Το 6% των μαθητών θεωρεί ότι το πρόγραμμα υπολογίζει τις τιμές που διαβάζονται, το 2 % εντοπίζει ότι τη σωστή μεταβλητή που αφορά το λάθος αλλά δεν είναι σε θέση να γράψει την κατάλληλη εντολή για τη μεταβλητή αυτή, ενώ το 6,1% αντικαθιστά τη δομή επανάληψης ΟΣΟ με Μέχρις_ότου ή με δομή επιλογής ενώ θα έπρεπε να κάνει αλλαγές στη δομή επιλογής που περιέχεται στο πρόγραμμα.

- Το 7,1% των μαθητών απαντά στο 3^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές έχουν τη δυνατότητα να τροποποιούν, να εντοπίζουν λάθη, και να ερμηνεύουν το πρόγραμμα σε ικανοποιητικό βαθμό. Το 2% θεωρεί ότι το πρόγραμμα υπολογίζει θετικές τιμές που δίνονται από το πληκτρολόγιο, επίσης το 2 % αντιλαμβάνεται και διορθώνει το λάθος γράφοντας τη σωστή εντολή σε λάθος θέση μέσα στο πρόγραμμα ενώ το 3,1% των μαθητών υπολογίζει μόνο την περίπτωση που δίνεται αρνητικός αριθμός ως τιμή μέσα στο πρόγραμμα.

- Το 38,4% των μαθητών αναπτύσσουν την κριτική τους ικανότητα ώστε να εντοπίζουν λάθη και να τα διορθώνουν, να αναγνωρίζουν το ρόλο των μεταβλητών, και να τροποποιούν με βέλτιστο τρόπο το πρόγραμμα με αποτέλεσμα οι απαντήσεις τους να τοποθετούνται στο 4^ο επίπεδο ΙΑΠ. Από αυτούς το 17,2% αντιλαμβάνονται σωστά το πρόβλημα και δίνουν ως απάντηση την απόλυτη τιμή, το 19,2 αντιλαμβάνονται σωστά το πρόβλημα και δίνουν ως απάντηση τη σωστή εντολή στη σωστή θέση μέσα στο πρόγραμμα καθώς επίσης το 2% γράφουν τη σωστή λύση με απλή επιλογή (Αν).

- Το 7,1% των μαθητών απαντά στο 5^ο επίπεδο. Το επίπεδο αυτό δεν ορίζεται για τη συγκεκριμένη δραστηριότητα στο πρώτο ερώτημα. Το 1% των μαθητών αντιλαμβάνεται με απόλυτη πληρότητα το πρόγραμμα, έτσι βρίσκει και διορθώνει το λάθος στο δεύτερο ερώτημα. Οι μαθητές

αυτοί δίνουν ως απάντηση τη σωστή εντολή στη σωστή θέση και εμφανίζουν την απόλυτη τιμή μέσα στην επανάληψη στη σωστή θέση. Στο τρίτο ερώτημα το 6,1% των μαθητών τροποποιούν το πρόγραμμα γράφοντας τη βέλτιστη λύση με χρήση της συνάρτησης της απόλυτης τιμής.

Σύγκριση τρόπων αξιολόγησης στο επίπεδο Αξιολογώ

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 33,3% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 55,6% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν αναπτύξουν την κριτική ικανότητα και την ικανότητα ελέγχου σε ένα δοσμένο πρόγραμμα που υπολογίζει την απόλυτη τιμή τυχαίων αριθμών, διαφοροποιούνται σε μεγάλο βαθμό από τον τρόπο αξιολόγησης. Για το προδρομικό επίπεδο η εφαρμογή διαδικασιών, υποπολλαπλασιάζεται με την αξιολόγηση ΙΑΠ συγκριτικά με την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 14,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 9,1% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις γνώσεις για τον εντοπισμό λαθών και τη διόρθωση, την τροποποίηση του προγράμματος ώστε να δίνει την βέλτιστη λύση, και την ερμηνεία του προγράμματος αυξάνεται σε ποσοστό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση. Στο επίπεδο επιμέρους κατανόησης η κριτική ικανότητα αυξάνεται με την αξιολόγηση ΙΑΠ από ότι με την παραδοσιακή αξιολόγηση.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 7,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 3% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές κρίνουν και να ελέγχουν γνώσεις προγραμματισμού, σε ικανοποιητικό βαθμό, που αφορούν τη διόρθωση λαθών σε μια επαναληπτική διαδικασία και τη βελτιστοποίηση του προγράμματος υπερδιπλασιάζεται με την αξιολόγηση ΙΑΠ. Στο επίπεδο προσεγγιστικής κατανόησης η κριτική ικανότητα των μαθητών σε γνώσεις προγραμματισμού πολλαπλασιάζεται με την ΙΑΠ.

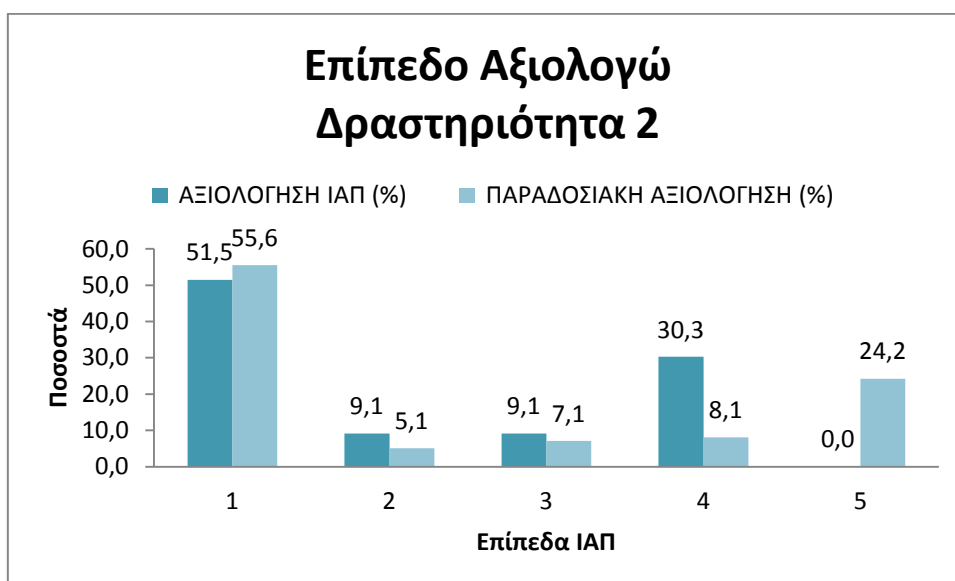
- Στο 4^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 38,4% με την ΙΑΠ και 5,1% με την παραδοσιακή αξιολόγηση. Ενώ το 5^ο επίπεδο δεν ορίζεται για το πρώτο ερώτημα με την αξιολόγηση ΙΑΠ έτσι το ποσοστό είναι 7,1% ενώ το αντίστοιχο ποσοστό στην παραδοσιακή αξιολόγηση είναι 27,3%. Στην συγκεκριμένη δραστηριότητα που αφορά τον εντοπισμό και την διόρθωση λαθών στις δομές επανάληψης καθώς και τη βελτιστοποίηση μιας δομής επιλογής, στον υπολογισμό της απόλυτης τιμής τυχαίων αριθμών που διαβάζονται από το πληκτρολόγιο διαπιστώνεται ότι περισσότεροι μαθητές έχουν ένα πάρα πολύ καλό επίπεδο γνώσεων ώστε να απαντούν σωστά στα ερωτήματα της δραστηριότητας. Το ποσοστό των απαντήσεων με την αξιολόγηση ΙΑΠ στο 4^ο και 5^ο επίπεδο αθροιστικά διαφοροποιείται και είναι πολύ μεγαλύτερο από τα αντίστοιχα ποσοστά συνολικά με την παραδοσιακή αξιολόγηση.

Πίνακας 5.15: ΔΡ2 αξιολόγηση ΙΑΠ επίπεδο Αξιολογώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΑΞΙΟΛΟΓΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2			ΑΞΙΟΛΟΓΗΣΗ
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	ΙΑΠ (%)
	1	11	18	22
2	3	6	0	9,1
3	3	5	1	9,1
4	16	4	10	30,3
5	0	0	0	0,0

Πίνακας 5.16: ΔΡ2 αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Αξιολογώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΑΞΙΟΛΟΓΩ ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2			ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ
	Ερώτημα Α (αριθμός)	Ερώτημα Β (αριθμός)	Ερώτημα Γ (αριθμός)	(%)
	1	14	18	23
2	0	3	2	5,1
3	4	3	0	7,1
4	5	0	3	8,1
5	10	9	5	24,2



Σχήμα 5.9: ΔΡ2 σύγκριση αποτελεσμάτων επίπεδο Αξιολογώ

Αξιολόγηση γνώσεων με ΙΑΠ

Στον πίνακα 5.17 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.18 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.9 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Αξιολογώ της αναθεωρημένης ταξινομίας Bloom και με τους δύο τρόπους αξιολόγησης.

Η δεύτερη δραστηριότητα που δόθηκε στους μαθητές για το επίπεδο Αξιολογώ υπολόγιζε τη μέγιστη 10 αριθμών και είχε τρία ερωτήματα. Στο πρώτο ερώτημα οι μαθητές έπρεπε βρουν ποια μεταβλητή απεικονίζει το μέγιστο, στο δεύτερο ερώτημα να εντοπίσουν το λάθος και να το διορθώσουν και στο τρίτο ερώτημα να τροποποιήσουν το πρόγραμμα ώστε με λιγότερες εντολές να δίνει τα ίδια αποτελέσματα.

Σχετικά με το πρώτο ερώτημα από τον πίνακα 5.17 διαπιστώνεται ότι οι μισοί μαθητές είναι σε θέση να απαντήσουν σωστά, γράφοντας ότι το πρόγραμμα υπολογίζει την μέγιστη από 10 τυχαίους αριθμούς. Επομένως οι απαντήσεις αυτών των μαθητών ανήκουν στο 4^ο επίπεδο (16 απαντήσεις). Σημαντικό είναι επίσης ότι 11 μαθητές (προδρομικό επίπεδο), δηλαδή το 1/3 των μαθητών δεν μπορούν να κρίνουν τι υπολογίζει το πρόγραμμα.

Για το δεύτερο ερώτημα η πλειοψηφία των μαθητών απαντά στο 1^ο επίπεδο (18 απαντήσεις), δεν έχει την κριτική ικανότητα να εντοπίσουν ποια εντολή λείπει από το πρόγραμμα ώστε να μπορεί να εκτελείται και να δίνει αποτελέσματα. Τέλος, για το τρίτο ερώτημα οι περισσότεροι μαθητές ανήκουν στο 1^ο επίπεδο (22 απαντήσεις), δηλαδή οι μαθητές δεν είναι σε θέση να τροποποιήσουν

το πρόγραμμα ώστε να δίνει τα σωστά αποτελέσματα με λιγότερες εντολές αλλά όμως 10 από αυτούς απαντούν στο 4^ο επίπεδο, δηλαδή αντιλαμβάνονται ότι πρέπει να μετατρέψουν τη δομή επανάληψης ΟΣΟ σε ΓΙΑ.

Συνολικά ως προς το επίπεδο Αξιολογώ για την δεύτερη δραστηριότητα έχουμε τα εξής αποτελέσματα:

- Το 51,5% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να αναπτύξουν κριτική σκέψη και να απαντήσουν σε ερωτήματα για ένα πρόγραμμα που τους έχει δοθεί σχετικά με τον υπολογισμό της μέγιστης τιμής 10 τυχαίων αριθμών. Από αυτούς το 11,1% δεν μπορεί να βρει τι υπολογίζει το δοσμένο πρόγραμμα, το 18,2% δεν μπορεί να εντοπίσει το λάθος που υπάρχει στο πρόγραμμα και το 22,2% δε μπορεί να τροποποιήσει το πρόγραμμα ώστε με λιγότερες εντολές να δίνει τα ίδια αποτελέσματα. Παρατηρούμε ότι το ερώτημα με τη μεγαλύτερη δυσκολία είναι το τρίτο. Κατά συνέπεια οι μαθητές δεν είναι σε θέση να όταν τους δοθεί μια λύση να κάνουν τις απαραίτητες αλλαγές ώστε να δώσουν τη βέλτιστη λύση. Κυρίως εντοπίζεται η δυσκολία των μαθητών στο να μετατρέψουν μια δομή επανάληψης σε άλλη.

- Το 9,1% των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να αναπτύξουν κριτική σκέψη για στοιχειώδεις και πολύ απλές λειτουργίες ενός προγράμματος. Το 3% των μαθητών θεωρεί ότι το πρόγραμμα υπολογίζει τιμές που σχετίζονται με τη μεταβλητή που διαβάζεται, το 6,1% εντοπίζει ότι τη σωστή μεταβλητή που αφορά το λάθος αλλά δεν είναι σε θέση να γράψει την κατάλληλη εντολή για αυτή τη μεταβλητή, ενώ κανένας μαθητής δεν αντικαθιστά τη δομή επανάληψης ΟΣΟ με ΓΙΑ, με πολλά συντακτικά λάθη

- Το 30,3% των μαθητών αναπτύσσουν την κριτική τους ικανότητα ώστε να εντοπίζουν λάθη και να τα διορθώνουν, να αναγνωρίζουν το ρόλο των μεταβλητών, και να τροποποιούν με βέλτιστο τρόπο το πρόγραμμα με αποτέλεσμα οι απαντήσεις τους να τοποθετούνται στο 4^ο επίπεδο ΙΑΠ. Από αυτούς το 16,2% αντιλαμβάνονται σωστά το πρόβλημα και δίνει ως απάντηση τη μέγιστη τιμή, το 4 % αντιλαμβάνονται σωστά το πρόβλημα και δίνουν ως απάντηση τη σωστή εντολή στη σωστή θέση μέσα στο πρόγραμμα καθώς επίσης το 10 % γράφει τη μετατροπή του ΟΣΟ σε ΓΙΑ, χωρίς κανένα συντακτικό λάθος.

- Το 5^ο επίπεδο δεν ορίζεται για αυτή τη δραστηριότητα.

Σύγκριση τρόπων αξιολόγησης στο επίπεδο Αξιολογώ

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 51,5% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 55,6% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν αναπτύξουν την κριτική ικανότητα και την ικανότητα ελέγχου σε ένα δοσμένο πρόγραμμα που υπολογίζει την απόλυτη τιμή τυχαίων αριθμών, δεν διαφοροποιούνται από τον τρόπο

αξιολόγησης. Για το προδρομικό επίπεδο η κριτική ικανότητα των μαθητών, τείνει να ταυτίζεται με τους δύο τρόπους αξιολόγησης.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 9,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 5,1% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις γνώσεις για τον εντοπισμό λαθών και τη διόρθωση, την τροποποίηση του προγράμματος ώστε να δίνει την βέλτιστη λύση, και την ερμηνεία του προγράμματος τείνει να διαπλασιαστεί σε ποσοστό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση. Στο επίπεδο επιμέρους κατανόησης η κριτική ικανότητα αυξάνεται με την αξιολόγηση ΙΑΠ από ότι με την παραδοσιακή αξιολόγηση.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 9,1% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 7,1% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που κρίνουν και ελέγχουν γνώσεις προγραμματισμού, σε ικανοποιητικό βαθμό, που αφορούν τη διόρθωση λαθών σε μια επαναληπτική διαδικασία και τη βελτιστοποίηση του προγράμματος αυξάνονται με την αξιολόγηση ΙΑΠ. Στο επίπεδο προσεγγιστικής κατανόησης η κριτική ικανότητα των μαθητών σε γνώσεις προγραμματισμού είναι μεγαλύτερη με την ΙΑΠ συγκρίνοντας την με την παραδοσιακή αξιολόγηση.

- Στο 4^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 30,3% με την ΙΑΠ και 8,1% με την παραδοσιακή αξιολόγηση. Ενώ το 5^ο επίπεδο δεν ορίζεται για την αξιολόγηση ΙΑΠ ενώ το αντίστοιχο ποσοστό στην παραδοσιακή αξιολόγηση είναι 24,2%. Το ποσοστό των απαντήσεων με την αξιολόγηση ΙΑΠ στο 4^ο και 5^ο επίπεδο αθροιστικά δεν διαφοροποιείται σε μεγάλο βαθμό από τα αντίστοιχα συνολικά ποσοστά με την παραδοσιακή αξιολόγηση.

5.6 Αποτελέσματα ως προς το Επίπεδο Δημιουργώ

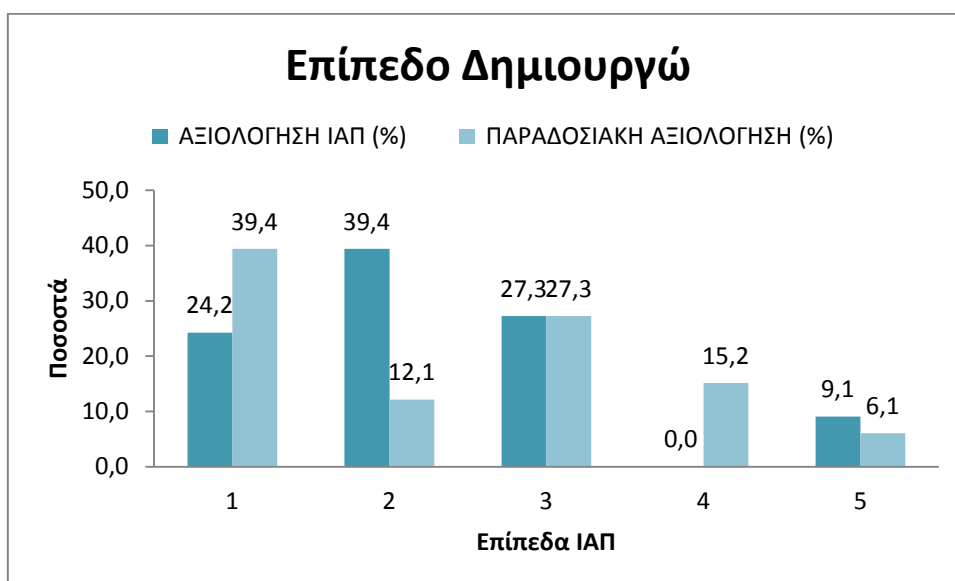
Το επίπεδο «Δημιουργώ» ορίζεται ως η τοποθέτηση στοιχείων μαζί για να σχηματίσουν ένα ενιαίο σύνολο, αναδιοργανώνοντας τα στοιχεία σε μια νέα δομή. Επιδιώκεται με ένα νέο εναλλακτικό αλγόριθμο ή πρόγραμμα. Επιτυγχάνεται με επινόηση μιας εναλλακτικής διαδικασίας ή στρατηγικής για την επίλυση ενός προβλήματος. Η κατηγορία αυτή περιλαμβάνει τη δημιουργία, το σχεδιασμό και την παραγωγή. Επίσης κατατάσσονται σε αυτό το επίπεδο σύνθετες εργασίες προγραμματισμού, οι οποίες μπορεί να περιλαμβάνουν τη διαίρεση της εργασίας σε μικρότερα τμήματα στα οποία μπορούν να εφαρμοστούν γνωστοί αλγόριθμοι και διαδικασίες.

Πίνακας 5.17: Αξιολόγηση ΙΑΠ επίπεδο Δημιουργώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΔΗΜΙΟΥΡΓΩ (αριθμός)	ΑΞΙΟΛΟΓΗΣΗ ΙΑΠ (%)
1	8	24,2
2	13	39,4
3	9	27,3
4	0	0,0
5	3	9,1

Πίνακας 5.18: Αντιστοίχιση βαθμολογιών στα επίπεδα ΙΑΠ επίπεδο Δημιουργώ

ΕΠΙΠΕΔΑ ΙΑΠ	ΕΠΙΠΕΔΟ ΔΗΜΙΟΥΡΓΩ (αριθμός)	ΠΑΡΑΔΟΣΙΑΚΗ ΑΞΙΟΛΟΓΗΣΗ (%)
1	13	39,4
2	4	12,1
3	9	27,3
4	5	15,2
5	2	6,1



Σχήμα 5.10: Σύγκριση αποτελεσμάτων επίπεδο Δημιουργώ

Αξιολόγηση γνώσεων με ΙΑΠ

Στον πίνακα 5.19 κατηγοριοποιούνται οι απαντήσεις των 33 μαθητών που συμμετείχαν στην έρευνα στα επίπεδα ΙΑΠ, ενώ στον πίνακα 5.20 αντιστοιχίζονται οι βαθμολογίες των γραπτών από τους βαθμολογητές σύμφωνα με την κλίμακα που καθορίστηκε η αντιστοίχιση στα επίπεδα ΙΑΠ. Τέλος στο σχήμα 5.10 παρουσιάζονται τα ποσοστά των απαντήσεων των μαθητών στο επίπεδο Δημιουργώ της αναθεωρημένης ταξινομίας Bloom και με τους δύο τρόπους αξιολόγησης.

Η δραστηριότητα που δόθηκε στους μαθητές για το επίπεδο Δημιουργώ απαιτούσε από αυτούς να δημιουργήσουν μια επαναληπτική διαδικασία η οποία θα τερμάτιζε στην περίπτωση που το PIN ενός κινητού ταυτιζόταν με το αρχικό ή στην περίπτωση που εξαντλούσε 3 προσπάθειες εντοπισμού του σωστού PIN. Ανάλογα με το αποτέλεσμα της διαδικασίας θα εμφανίζονταν στην οθόνη του χρήστη το αντίστοιχο μήνυμα επιτυχίας ή αποτυχίας και ο αριθμός των προσπαθειών.

Για το επίπεδο Δημιουργώ έχουμε τα εξής αποτελέσματα:

- Το 24,2% των μαθητών απαντά στο 1^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές δεν μπορούν να διασπάσουν το σύνθετο πρόβλημα που τους δόθηκε σε επιμέρους προβλήματα και κατόπιν να κατασκευάσουν ένα πρόγραμμα που απαιτούσε χρήση δομής επανάληψης με άγνωστο αριθμό επαναλήψεων και σύνθετη συνθήκη καθώς και υπολογισμό πλήθους.
- Το 39,4% των μαθητών απαντά στο 2^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές μπορούν να εκτελέσουν και να παράγουν ένα πρόγραμμα με στοιχειώδεις απλές λειτουργίες. Είναι σε θέση να κάνουν τον κατάλληλο έλεγχο για τη σωστή εμφάνιση μηνυμάτων, χωρίς όμως να μπορούν να χρησιμοποιούν τη δομή επανάληψης.

- Το 27,3% των μαθητών απαντά στο 3^ο επίπεδο ΙΑΠ. Αυτοί οι μαθητές έχουν τη δυνατότητα να προσεγγίσουν σε ικανοποιητικό βαθμό, τη δημιουργία ενός προγράμματος που απαιτεί μια επαναληπτική διαδικασία με δομή επιλογής και υπολογισμό πλήθους. Χρησιμοποιούν δομή επανάληψης με συνθήκη που δεν ελέγχει όλες τις απαιτήσεις του προβλήματος δηλαδή όλες τις οριακές τιμές ώστε να γίνεται σωστός υπολογισμός των προσπαθειών ή δεν γίνεται ο κατάλληλος έλεγχος με δομή επιλογής για τη σωστή εμφάνιση μηνυμάτων.

- Κανένας μαθητής δεν έχει τη δυνατότητα παραγωγής του προγράμματος με πληρότητα. Δεν υπάρχουν μαθητές που να χρησιμοποιούν τη δομή επανάληψης Όσο .. Επανάλαβε με συνθήκη που να πληρεί τις προϋποθέσεις του προβλήματος και να γίνεται σωστός υπολογισμός των προσπαθειών και σωστή εμφάνιση των κατάλληλων μηνυμάτων με αποτέλεσμα οι απαντήσεις τους να τοποθετούνται στο 4^ο επίπεδο ΙΑΠ.

- Το 9,1% των μαθητών απαντούν στο 5^ο επίπεδο. Οι μαθητές αυτοί διαθέτουν ικανότητα διάσπασης του προβλήματος σε μέγιστο βαθμό ώστε να μπορούν να δημιουργούν το κατάλληλο πρόγραμμα και να δίνουν ως απάντηση τη βέλτιστη λύση. Έχουν τη δυνατότητα να χρησιμοποιούν τη δομή επανάληψης Μέχρις_ότου με συνθήκη που να πληρεί τους περιορισμούς του προβλήματος, να γίνεται σωστός υπολογισμός των προσπαθειών και σωστή εμφάνιση των κατάλληλων μηνυμάτων.

Σύγκριση τρόπων αξιολόγησης στο επίπεδο Δημιουργώ

- Στο 1^ο επίπεδο (προδρομικό) οι μαθητές σε ποσοστό 24,2% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 39,4% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που δεν μπορούν να διασπάσουν το πρόβλημα που τους δόθηκε σε υποπροβλήματα και στη συνέχεια να παράγουν σε μια νέα ενιαία δομή ένα πρόγραμμα που απαιτεί δομή επανάληψης με δύο περιορισμούς, χρήση δομής επιλογής και υπολογισμό πλήθους διαφοροποιούνται σε κάποιο βαθμό από τον τρόπο αξιολόγησης. Για το προδρομικό επίπεδο η ικανότητα ανάλυσης ενός προβλήματος ελαττώνεται με την αξιολόγηση ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση.

- Στο 2^ο επίπεδο της επιμέρους κατανόησης οι μαθητές σε ποσοστό 39,4% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 12,1% απαντούν με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές που έχουν στοιχειώδεις ικανότητες να δημιουργούν ένα πρόγραμμα που απαιτεί μια επαναληπτική διαδικασία με δομή επιλογής και πλήθος τριπλασιάζεται σε ποσοστό με την ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση.

- Στο 3^ο επίπεδο οι μαθητές σε ποσοστό 27,3% απαντούν με την αξιολόγηση ΙΑΠ ενώ σε ποσοστό 27,3% με την παραδοσιακή αξιολόγηση. Διαπιστώνεται ότι οι μαθητές είναι σε θέση να διασπάσουν ένα πρόβλημα σε επιμέρους προβλήματα και να κατασκευάσουν το πρόγραμμα, καθώς μπορούν να χρησιμοποιούν σωστά τη δομή επιλογής και τον υπολογισμό μετρητή ενώ

δυσκολεύονται στη χρήση δομής επανάληψης που ελέγχει δύο μεταβλητές καθώς απαιτεί σε μεγαλύτερο βαθμό ικανότητες διάσπασης και σύνθεσης. Στο επίπεδο προσεγγιστικής κατανόησης η ικανότητα δημιουργίας προγράμματος ταυτίζεται και με τους δύο τρόπους αξιολόγησης.

- Στο 4^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 0% με την ΙΑΠ και 15,2% με την παραδοσιακή αξιολόγηση. Στην συγκεκριμένη δραστηριότητα που απαιτεί δυνατότητα διάσπασης του προβλήματος σε υποπροβλήματα και σύνθεση μιας νέας δομής δεν υπάρχουν μαθητές που να απαντούν σε αυτό το επίπεδο με ΙΑΠ. Στο συνδυαστικό επίπεδο η ικανότητα παραγωγής προγράμματος αναπτύσσεται μόνο με την παραδοσιακή αξιολόγηση.

- Στο 5^ο επίπεδο οι μαθητές απαντούν σε ποσοστό 9,1% με την αξιολόγηση ΙΑΠ ενώ το αντίστοιχο ποσοστό στην παραδοσιακή αξιολόγηση είναι 6,1%. Οι μαθητές που έχουν τη δυνατότητα να δώσουν τη βέλτιστη λύση στο πρόβλημα όπως ορίστηκε με την αξιολόγηση ΙΑΠ είναι περισσότεροι με την αξιολόγηση ΙΑΠ σε σχέση με την παραδοσιακή αξιολόγηση.

6. Συμπεράσματα - Συζήτηση

Ακολουθεί η παρουσίαση των συμπερασμάτων ως προς τα ερευνητικά ερωτήματα της έρευνας, πρώτα ως προς το επίπεδο γνώσεων προγραμματισμού με εφαρμογή του μοντέλου της Ιεραρχικής Αξιολόγησης γνώσεων Προγραμματισμού και στη συνέχεια σύγκριση των αποτελεσμάτων του μοντέλου με τα αποτελέσματα από την παραδοσιακή βαθμολόγηση.

Α)Ως προς τις γνώσεις προγραμματισμού

Η παρούσα έρευνα διερευνά τις γνώσεις προγραμματισμού τελειόφοιτων μαθητών δευτεροβάθμιας εκπαίδευσης. Στους μαθητές που συμμετείχαν στην έρευνα δόθηκαν 10 δραστηριότητες ταξινομημένες στα ιεραρχικά επίπεδα της αναθεωρημένης ταξινόμιας Bloom. Στη συνέχεια οι απαντήσεις των μαθητών κατηγοριοποιήθηκαν σύμφωνα με τη μέθοδο της Ιεραρχικής Αξιολόγησης Προγραμματισμού (ΙΑΠ), η οποία βασίζεται στις αρχές της ταξινόμιας SOLO. Αναλύοντας τα αποτελέσματα της έρευνας οδηγούμαστε σε πολύ σημαντικά συμπεράσματα. Τα συμπεράσματα διατυπώνονται για κάθε επίπεδο της ΙΑΠ.

Για το προδρομικό επίπεδο η ανάκληση γνώσεων και η απομνημόνευση αποτελούν δεξιότητες στις οποίες οι μαθητές σημειώνουν τις χαμηλότερες επιδόσεις. Η ανάκληση των τύπων δεδομένων (ακέραιοι, πραγματικοί, χαρακτήρες και λογικοί) που χρησιμοποιούνται σε ένα πρόγραμμα καθώς και η σύνταξη των δομών επανάληψης αποτελούν σημεία δυσκολίας για τους μαθητές, σε αντίθεση με τα διαγράμματα ροής τα οποία θυμούνται οι μαθητές.

Η ικανότητα της εφαρμογής συγκεκριμένων διαδικασιών μέσα σε ένα πρόγραμμα έπεται της ικανότητας απομνημόνευσης. Συγκεκριμένα οι μαθητές που δεν μπορούν να εφαρμόσουν τη διαδικασία υπολογισμού ελάχιστης τιμής καθώς και το όνομα αυτού που την έχει σε προκαθορισμένο αριθμό από τυχαίες τιμές είναι λιγότεροι από τους μαθητές που δεν μπορούν να εφαρμόσουν τη διαδικασία υπολογισμού αθροίσματος και πλήθους για τυχαίες θετικές τιμές. Οι μαθητές που δεν μπορούν να εφαρμόσουν τη δομή επανάληψης Για, η οποία αφορά γνωστό αριθμό επαναλήψεων είναι λιγότεροι από αυτούς που δεν μπορούν να εφαρμόσουν την ΟΣΟ, η οποία χρησιμοποιείται συνήθως για άγνωστο αριθμό επαναλήψεων. Επίσης διαπιστώνεται από τα αποτελέσματα ότι η εφαρμογή της αρχικοποίησης ενός αθροιστή στην κατάλληλη θέση αλλά κυρίως η χρήση μεταβλητής που κρατάει το όνομα της ελαχίστης τιμής αποτελούν δεξιότητες με το μεγαλύτερο βαθμό δυσκολίας για το προδρομικό επίπεδο.

Οι μαθητές σημειώνουν καλύτερες υψηλότερες στη δημιουργία προγράμματος σε σχέση με τη δεξιότητα εφαρμογής. Οι μαθητές δεν μπορούν να διασπάσουν ένα σύνθετο πρόβλημα που τους δόθηκε σε επιμέρους προβλήματα και κατόπιν να κατασκευάσουν ένα πρόγραμμα που απαιτεί χρήση δομής επανάληψης με άγνωστο αριθμό επαναλήψεων και σύνθετη συνθήκη καθώς και υπολογισμό πλήθους.

Οι επιδόσεις των μαθητών κατατάσσουν την ικανότητα κατανόησης του προγράμματος να ακολουθεί τη ικανότητα δημιουργίας για το προδομικό επίπεδο. Αυτό σημαίνει ότι οι μαθητές δεν είναι σε θέση να ερμηνεύσουν τη λειτουργία ενός προγράμματος, να το εκτελέσουν σωστά και να μετατρέψουν σωστά μια δομή επανάληψης σε άλλη αλλά κυρίως να ελέγχουν τις οριακές τιμές σε μια επανάληψη. Για κάποιους μαθητές η εκτέλεση ενός δοσμένου προγράμματος είναι μια ικανότητα προδομική

Η ανάλυση και η διάσπαση του προβλήματος σε υποπροβλήματα είναι μια δεξιότητα που έπεται της κατανόησης του προβλήματος για το προδομικό επίπεδο σύμφωνα με τα αποτελέσματα της έρευνας. Οι μαθητές που δυσκολεύονται με τη χρήση δομής επανάληψης για γνωστό αριθμό επαναλήψεων, έλεγχο εγκυρότητας δεδομένων, χρήση δομής επιλογής, υπολογισμό πλήθους υπό όρους και υπολογισμό ποσοστού είναι περισσότεροι από τους μαθητές που έπρεπε να χρησιμοποιήσουν την ίδια διαδικασία σε αντίστοιχο πρόβλημα με δομή επανάληψης για άγνωστο αριθμό επαναλήψεων, χρήση δομής επιλογής, υπολογισμό πλήθους υπό όρους. Ο έλεγχος εγκυρότητας δεδομένων είναι μια προδομική δεξιότητα για τους μαθητές.

Τέλος η δεξιότητα με το μεγαλύτερο ποσοστό στο προδομικό επίπεδο είναι η αξιολόγηση και η κριτική ικανότητα. Οι μαθητές δεν είναι σε θέση να τροποποιήσουν ένα πρόγραμμα ώστε με λιγότερες εντολές να δίνει τα ίδια αποτελέσματα με αυτό που τους δόθηκε. Κατά συνέπεια οι μαθητές δεν μπορούν να κάνουν τις απαραίτητες αλλαγές σε ένα δοσμένο πρόγραμμα ώστε να δώσουν τη βέλτιστη λύση. Κυρίως εντοπίζεται η δυσκολία των μαθητών στη μετατροπή μιας δομής επανάληψης σε άλλη σε σχέση με την τροποποίηση μιας δομής επιλογής. Ο εντοπισμός λάθους που αφορά την αρχικοποίηση μεταβλητής δυσκολεύει περισσότερο τους μαθητές από ότι η σωστή εκτέλεση μιας επανάληψης με ΟΣΟ. Ως εκ τούτου μπορούμε να συμπεράνουμε ότι για το προδομικό επίπεδο την μεγαλύτερη δυσκολία αντιμετωπίζουν οι μαθητές στο να εκφράζουν κρίσεις για ένα συγκεκριμένο πρόγραμμα και τη μικρότερη να θυμούνται γνώσεις τις οποίες έχουν διδαχθεί.

Οι χαμηλότερες επιδόσεις για το επίπεδο επιμέρους κατανόησης (2^ο επίπεδο ΙΑΠ) σχετίζονται με την ικανότητα των μαθητών να εκτελέσουν και να εφαρμόσουν στοιχειώδεις και πολύ απλές λειτουργίες ενός προγράμματος. Από τα αποτελέσματα διαφαίνεται ότι όταν δίνεται ένα έτοιμο πρόγραμμα, εκχωρούν την ελάχιστη τιμή σε λάθος μεταβλητή καθώς επίσης σε δομή επανάληψης ΟΣΟ χρησιμοποιούν στη συνθήκη λάθος μεταβλητές και τις αντίστοιχες οριακές τιμές, επομένως αυτές οι διαδικασίες δυσκολεύουν τους μαθητές. Ενώ η εφαρμογή επανάληψης με ΓΙΑ και η αρχικοποίηση ενός αθροιστή με αλφαριθμητική τιμή θεωρούνται πιο εύκολες διαδικασίες. Επίσης διαπιστώνονται λάθη ως προς τη σειρά εκτέλεσης των εντολών αφού κάποιοι μαθητές θεωρούν ότι η επανάληψη προηγείται της αρχικοποίησης μιας μεταβλητής.

Κατόπιν ακολουθούν οι επιδόσεις που αφορούν την ικανότητα των μαθητών να κατανοήσουν το πρόβλημα και το πρόγραμμα. Η μετατροπή μιας δομής επανάληψης όχι σε άλλη αλλά σε δομή επιλογής αποτελεί μια διαδικασία περισσότερο κατανοητή από την εκτέλεση περισσότερων ή λιγότερων επαναλήψεων, ενώ οι οριακές τιμές στις συνθήκες αποτελούν σημείο δυσκολίας κατανόησης του προγράμματος. Οι επιδόσεις των μαθητών που αφορούν την ικανότητα να κρίνουν και να αξιολογούν ένα πρόγραμμα ακολουθεί την κατανόηση στο επίπεδο της επιμέρους κατανόησης. Η δεξιότητα να κρίνουν τι υπολογίζει ένα δοσμένο πρόγραμμα αποτελεί σημείο μεγαλύτερης δυσκολίας όταν το ζητούμενο είναι η απόλυτη τιμή ενώ μικρότερης δυσκολίας όταν το ζητούμενο είναι η μέγιστη τιμή γιατί πιθανόν οι μαθητές να έχουν λύσει και να είναι εξοικειωμένοι με αντίστοιχες διαδικασίες επίλυσης προβλημάτων. Όσον αφορά τον εντοπισμό ενός λάθους και τη διόρθωση του, η ικανότητα να αξιολογήσουν λάθη σχετικά με την αρχικοποίηση μεταβλητών είναι μεγαλύτερης δυσκολίας σε σχέση με λάθη που σχετίζονται με τη σωστή εκτέλεση μιας επανάληψης. Σχετικά με την τροποποίηση ενός προγράμματος εντοπίζονται δυσκολίες στο αξιολογήσουν οι μαθητές ότι χρειάζεται να κάνουν αλλαγές στη δομή επιλογής και όχι στη δομή επανάληψης.

Μετά την ικανότητα αξιολόγησης, υψηλότερες επιδόσεις παρατηρούνται στην ικανότητα απομνημόνευσης. Για τους μαθητές μικρότερης δυσκολίας είναι η απομνημόνευση των σχημάτων στα διαγράμματα ροής και η λειτουργία τους αφού γνωρίζουν τα σχήματα και δίνουν μια λάθος εξήγηση για κάποιο σχήμα. Οι περισσότεροι έχουν απομνημονεύσει σωστά τη σύνταξη μόνο μιας δομής επανάληψης. Τέλος οι μαθητές θυμούνται τους περισσότερους τύπους μεταβλητών αλλά όχι όλους.

Η ικανότητα ανάλυσης του προβλήματος έπεται της ικανότητας απομνημόνευσης. Οι μαθητές είναι σε θέση να χρησιμοποιούν δομή επανάληψης με ευχέρεια για γνωστό αριθμό επαναλήψεων, μπορούν να κάνουν έλεγχο εγκυρότητας για τα δεδομένα που διαβάζονται από το πληκτρολόγιο, και μπορούν να υπολογίζουν πλήθη, να χρησιμοποιούν τους τελεστές `div` και `mod` για τον υπολογισμό του τελευταίου ψηφίου καθώς και τον έλεγχο άρτιου ή περιττού αριθμού. Δυσκολεύονται να χρησιμοποιήσουν επανάληψη με άγνωστο πλήθος επαναλήψεων, να συνδυάζουν δομές επιλογής που ελέγχουν διαφορετικές μεταβλητές. Επίσης δυσκολίες εντοπίζονται στον υπολογισμό του ποσοστού, όμως μπορούν να υπολογίζουν σωστά το τελευταίο ψηφίο ενός ακέραιου αριθμού. Άρα η ικανότητα ανάλυσης ενός προβλήματος που περιλαμβάνει συνδυασμό διαδικασιών όπως συνδυασμό δομών επιλογής, υπολογισμό πλήθους υπό όρους και ποσοστού αποτελεί σημείο δυσκολίας για τους μαθητές.

Τέλος η δεξιότητα διάσπασης του προβλήματος και η δημιουργία προγράμματος αποτελεί την μικρότερη δυσκολία για τους μαθητές οι οποίοι είναι σε θέση μόνο να κάνουν απλούς ελέγχους με

δομή επιλογής και να εμφανίζουν τα αντίστοιχα μηνύματα. Άρα μπορούμε να συμπεράνουμε ότι στο επίπεδο της επιμέρους κατανόησης οι μαθητές έχουν τις καλύτερες επιδόσεις στη δημιουργία προγράμματος και τις χειρότερες στην εφαρμογή γνωστών διαδικασιών.

Στο επίπεδο της προσεγγιστικής κατανόησης οι μαθητές παρουσιάζουν τις χαμηλότερες επιδόσεις στην ικανότητα αξιολόγησης. Ελάχιστοι μαθητές δεν είναι σε θέση να κρίνουν τι υπολογίζει το πρόγραμμα και δίνουν απαντήσεις που πλησιάζουν τις σωστές απαντήσεις. Δυσκολία στον εντοπισμό του λάθους παρατηρείται κατά τον προσδιορισμό τη θέσης που θα γίνει η διόρθωση από τους μαθητές ενώ γνωρίζουν την εντολή που απαιτείται να συμπληρώσουν. Μικρή δυσκολία παρουσιάζουν οι μαθητές στο να μετατρέψουν μια δομή επανάληψης σε άλλη με ελάχιστα μικρά συντακτικά λάθη (όπως για παράδειγμα να γράψουν ΓΙΑ Χ ΑΠΟ 1 ΜΕΧΡΙ 10 ΕΠΑΝΑΛΑΒΕ) ενώ αντίθετα διαπιστώνεται μεγαλύτερη δυσκολία στο να τροποποιήσουν τη δομή επιλογής με αποτέλεσμα να μην ελέγχουν όλες τις περιπτώσεις (ελέγχουν μόνο την περίπτωση της αρνητικής τιμής στον υπολογισμό της απόλυτης τιμής)

Τις αμέσως υψηλότερες επιδόσεις παρουσιάζουν οι μαθητές στην ικανότητα εφαρμογής γνωστών διαδικασιών σε ένα πρόγραμμα. Οι μαθητές έχουν τη δυνατότητα να προσεγγίσουν σε ικανοποιητικό βαθμό, την εφαρμογή μιας επαναληπτικής διαδικασίας καθώς και να χρησιμοποιήσουν τις εντολές εκχώρησης. Μικρή δυσκολία διαπιστώνεται στην αρχικοποίηση μεταβλητής καθώς και στην εντολή εκχώρησης μεταβλητής τύπου χαρακτήρα δίνοντας λανθασμένη αλφαριθμητική τιμή (για παράδειγμα $OMIN \leftarrow ' '$). Μεγαλύτερη δυσκολία εντοπίζεται στο να γράψουν εντολές εκχώρησης που υπολογίζουν ελάχιστο ή πλήθος καθώς οι μαθητές κάνουν συντακτικά λάθη (εκχωρούν στη μεταβλητή που διαβάζουν το MIN ενώ πρέπει να γίνει το αντίστροφο) ή λογικά λάθη (για παράδειγμα, υπολογίζουν το πλήθος όπως υπολογίζουν τον μέσο όρο). Επιπρόσθετη δυσκολία αποτελεί η υλοποίηση της επανάληψης είτε πρόκειται για γνωστό είτε για άγνωστο αριθμό επαναλήψεων αφού οι μαθητές δυσκολεύονται να εφαρμόσουν σωστά την διαδικασία της επανάληψης κάνοντας λάθη ή στις μεταβλητές ή στη συνθήκη.

Κατόπιν ως δεξιότητα ακολουθεί η κατανόηση του προβλήματος και κατά συνέπεια του προγράμματος. Μικρότερη δυσκολία εντοπίζεται στην ερμηνεία του προγράμματος καθώς και στην κατανόηση των μεταβλητών. Ενώ οι μεγαλύτερες δυσκολίες των μαθητών αφορούν τον προσδιορισμό μιας συνθήκης όταν γνωρίζουν τις τιμές που εμφανίζονται κατά την επαναληπτική διαδικασία. Επίσης κάποιοι μαθητές δυσκολεύονται να τροποποιήσουν το όριο σε μια συνθήκη ώστε να δώσουν τη βέλτιστη λύση στο πρόβλημα καθώς επίσης δεν είναι σε θέση να εκτελούν σωστά την επανάληψη για τις αρχικές ή τις οριακές τιμές με αποτέλεσμα να υπολογίζουν λιγότερα δεδομένα (επαναλήψεις).

Η ικανότητα ανάλυσης ακολουθεί την ικανότητα κατανόησης. Γενικά διαπιστώνεται ότι οι μαθητές είναι σε θέση να αναλύουν προβλήματα που τους δίνονται και να δημιουργούν προγράμματα που περιέχουν επαναληπτικές διαδικασίες. Σημεία δυσκολίας στην ανάλυση ενός προβλήματος είναι ο υπολογισμός μεταβλητών όπως, το πλήθος με περιορισμούς που απαιτεί δομή επιλογής και η εμφωλευμένη επιλογή. Επίσης ο υπολογισμός του ποσοστού ο οποίος απαιτεί πλήθη, δυσκολεύει τους μαθητές.

Οι ικανότητες της απομνημόνευσης και της δημιουργίας, είναι ικανότητες ισοδύναμες για τους μαθητές και σε αυτές οι μαθητές σημειώνουν τις υψηλότερες επιδόσεις. Για την απομνημόνευση τα σημεία δυσκολίας ιεραρχικά είναι η σύνταξη των τριών δομών επανάληψης, τα σχήματα στο διάγραμμα ροής και οι τύποι των δεδομένων. Σε όλες τις περιπτώσεις οι μαθητές δεν θυμούνται ή μια δομή ή ένα σχήμα ή ένα τύπο δεδομένων ή η περιγραφή γίνεται μέσα από παραδείγματα. Χαρακτηριστική είναι η περίπτωση της εντολής εκχώρησης για τους τύπους των δεδομένων την οποία παρότι οι μαθητές γνωρίζουν, δεν γράφουν παραδείγματα τιμών με τη χρήση αυτής. Για τη δημιουργία προγράμματος οι μαθητές χρησιμοποιούν δομή επανάληψης με συνθήκη που δεν ελέγχει όλες τις απαιτήσεις του προβλήματος δηλαδή όλες τις οριακές τιμές ώστε να γίνεται σωστός υπολογισμός των ζητούμενων ή δεν γίνεται ο κατάλληλος έλεγχος με δομή επιλογής για τη σωστή εμφάνιση μηνυμάτων. Άρα στο επίπεδο της προσεγγιστικής κατανόησης οι επιδόσεις των μαθητών είναι υψηλότερες στην απομνημόνευση γνώσεων και τη δημιουργία προγράμματος και χαμηλότερες στην αξιολόγηση γνώσεων.

Για το συνδυαστικό επίπεδο οι χαμηλότερες επιδόσεις των μαθητών παρατηρούνται στην παραγωγή προγράμματος με πληρότητα. Δεν υπάρχουν μαθητές που να χρησιμοποιούν δομή επανάληψης συνδυάζοντας δύο συνθήκες και λογικό τελεστή. Συμπεραίνουμε ότι δεν έχουν κατακτήσει την ικανότητα να δημιουργούν προγράμματα που απαιτούν σύνθεση και τα οποία δεν έχουν διδαχθεί στην τάξη. Κατόπιν ακολουθεί η ικανότητα ανάλυσης. Οι μαθητές έχουν τη δυνατότητα διάσπασης του προβλήματος σε υποπροβλήματα με πληρότητα, δυσκολεύονται όμως να κάνουν έλεγχο εγκυρότητας δεδομένων καθώς αυτό απαιτεί εμφωλευμένη επανάληψη και δεν είναι εξοικειωμένοι με τη διαδικασία αυτή. Είναι όμως σε θέση να χρησιμοποιούν εμφωλευμένες δομές επιλογής ώστε να υπολογίζουν σωστά πλήθη και ποσοστά. Οι παραπάνω μαθητές έχουν την ευχέρεια να επιλύουν προβλήματα που απαιτούν διαδικασίες επανάληψης με γνωστό ή άγνωστο πλήθος επαναλήψεων.

Στη συνέχεια ακολουθεί η ικανότητα αξιολόγησης. Κατά την αξιολόγηση μεγαλύτερη δυσκολία παρατηρείται στους μαθητές να εντοπίσουν και να διορθώσουν λάθη που αφορούν αρχικοποίηση μεταβλητής και μικρότερη δυσκολία για λάθη σχετικά με τη σύνταξη και τη εκτέλεση μιας δομής επανάληψης. Επίσης όπως φαίνεται από τα αποτελέσματα δυσκολεύονται περισσότερο να

τροποποιήσουν μια δομή επιλογής σε ένα πρόγραμμα παρά μια δομή επανάληψης. Αυτό ίσως οφείλεται στο γεγονός ότι έχουν λύσει πολλές τέτοιες ασκήσεις μέσα στην τάξη.

Οι ικανότητες της απομνημόνευσης και της κατανόησης είναι σχεδόν ισοδύναμες για το συνδυαστικό επίπεδο. Περίπου οι μισοί μαθητές έχουν αναπτυγμένες αυτές τις δεξιότητες. Βέβαια ορισμένοι μαθητές όταν γράφουν τη σύνταξη μιας δομής, γράφουν τη συνθήκη με παράδειγμα πιθανά επειδή είναι πιο κατανοητή έτσι για τους ίδιους. Τα παραδείγματα με εντολές εκχώρησης για τους τύπους δεδομένων αποτελούν σημείο μεγαλύτερης δυσκολίας όπως ήδη έχει αναφερθεί. Σχετικά με την κατανόηση, σημείο μεγαλύτερης δυσκολίας αποτελεί η δημιουργία συνθήκης όταν είναι γνωστό το αποτέλεσμα μιας επαναληπτικής διαδικασίας. Η εκτέλεση μιας επαναληπτικής διαδικασίας και ο υπολογισμός αθροίσματος είναι κατανοητές δεξιότητες για τους μαθητές.

Τις υψηλότερες επιδόσεις παρουσιάζουν οι μαθητές στην ικανότητα εφαρμογής. Σημεία μεγαλύτερης δυσκολίας αποτελούν η σύνταξη συνθήκης με ΟΣΟ, δηλαδή άγνωστο αριθμό επαναλήψεων και τα συντακτικά λάθη στην εντολή εκχώρησης του ελάχιστου. Συμπεραίνουμε ότι οι μαθητές έχουν ευχέρεια στο να υπολογίζουν πλήθη, αθροίσματα και ελάχιστα, μέγιστα σε μια επαναληπτική διαδικασία. Γνωρίζουν τις παραπάνω διαδικασίες και μπορούν με άνεση να τις εφαρμόζουν σε ένα πρόγραμμα. Άρα για το συνδυαστικό επίπεδο οι μαθητές παρουσιάζουν τις καλύτερες επιδόσεις στην εφαρμογή γνωστών διαδικασιών σε ένα πρόγραμμα ενώ τις χειρότερες στην δημιουργία προγράμματος.

Το επίπεδο της εκτεταμένης θεώρησης δεν ορίζεται για τις δραστηριότητες που σχετίζονται με το επίπεδο της εφαρμογής καθώς και για ερωτήματα των επιπέδων απομνημόνευσης, κατανόησης και αξιολόγησης. Οπότε για τις δεξιότητες αυτές δεν μπορούν να εξαχθούν ακριβή και ασφαλή συμπεράσματα σχετικά με το επίπεδο αυτό. Τις χαμηλότερες επιδόσεις παρουσιάζουν οι μαθητές στην ανάλυση ενός προβλήματος σε υποπροβλήματα καθώς δεν είναι σε θέση να λάβουν υπόψη όλους τους περιορισμούς του προβλήματος και να γράψουν έτσι τη βέλτιστη λύση που να ελέγχει όλα τα ενδεχόμενα. Ενώ παρατηρούνται υψηλότερες επιδόσεις κατά τη δημιουργία ενός προγράμματος επειδή γνωρίζουν πολύ καλά όλες τις δομές επανάληψης και έχουν τη δυνατότητα να επιλέγουν και να γράφουν την πλέον κατάλληλη.

B) Ως προς τους τρόπους αξιολόγησης

Η παρούσα εργασία συγκρίνει τις επιδόσεις τελειόφοιτων μαθητών σε εξετάσεις γνώσεων προγραμματισμού με την αξιολόγηση ΙΑΠ καθώς και με την παραδοσιακή αξιολόγηση. Για το σκοπό αυτό δόθηκαν σε έξι έμπειρους βαθμολογητές οι απαντήσεις των μαθητών στις δραστηριότητες της έρευνας. Κάθε γραπτό μαθητή βαθμολογήθηκε από έναν βαθμολογητή. Στη συνέχεια οι

βαθμολογίες αυτές ομαδοποιήθηκαν και κατηγοριοποιήθηκαν στα επίπεδα ΙΑΠ. Από τα αποτελέσματα οδηγούμαστε στα παρακάτω συμπεράσματα.

Στο προδομικό επίπεδο οι μαθητές που δεν μπορούν να ανακαλέσουν γνώσεις προγραμματισμού διαφοροποιούνται κατά μεγάλο βαθμό από τον τρόπο αξιολόγησης καθώς η αξιολόγηση ΙΑΠ δίνει πολύ χαμηλότερες επιδόσεις από την παραδοσιακή αξιολόγηση. Στο επίπεδο κατανόησης, παρότι διαφέρουν τα ποσοστά των επιδόσεων των μαθητών στις δύο δραστηριότητες καταλήγουμε στο ίδιο συμπέρασμα. Οι επιδόσεις των μαθητών υποδιπλασιάζονται με την αξιολόγηση ΙΑΠ σε σύγκριση με την παραδοσιακή αξιολόγηση. Τα διαφορετικά ποσοστά σε κάθε δραστηριότητα οφείλονται στο βαθμό δυσκολίας κατανόησης γνώσεων της συγκεκριμένης δραστηριότητας. Σε ανάλογα συμπεράσματα καταλήγει η έρευνα και στο επίπεδο εφαρμογής, όπου οι επιδόσεις με αξιολόγηση ΙΑΠ υποπολλαπλασιάζονται ανάλογα με τη δυσκολία της κάθε δραστηριότητας. Επίσης η ικανότητα ανάλυσης υποδιπλασιάζεται στην πρώτη δραστηριότητα, ενώ ελαττώνεται στη δεύτερη δραστηριότητα με την αξιολόγηση ΙΑΠ. Η ικανότητα αξιολόγησης υποπολλαπλασιάζεται στη πρώτη δραστηριότητα ενώ οι δύο τρόποι αξιολόγησης τείνουν να ταυτιστούν ως προς τα αποτελέσματα τους στη δεύτερη δραστηριότητα. Τέλος στο επίπεδο δημιουργίας οι επιδόσεις των μαθητών ελαττώνονται με την αξιολόγηση ΙΑΠ.

Σε όλα τα επίπεδα Bloom παρατηρούνται χαμηλότερες επιδόσεις με την ΙΑΠ. Ως εκ τούτου καταλήγουμε στο συμπέρασμα ότι η Ιεραρχική αξιολόγηση Προγραμματισμού στο προδομικό επίπεδο συνολικά δίνει καλύτερα αποτελέσματα από την παραδοσιακή αξιολόγηση καθώς αποτελεί έναν πιο αναλυτικό και στοχοθετημένο τρόπο αξιολόγησης που περιορίζει κατά μεγάλο βαθμό τα ποσοστά επιδόσεων των μαθητών για όλα τα επίπεδα της ταξινομίας Bloom.

Για το επίπεδο της επιμέρους κατανόησης η ανάκληση γνώσεων που ζητήθηκε από τους μαθητές στο πλαίσιο της έρευνας, επειδή η διαφορά μιας ποσοστιαίας μονάδας δεν θεωρείται στατιστικά σημαντική, οδηγεί στο συμπέρασμα ότι οι δύο τρόποι αξιολόγησης ταυτίζονται ως προς τα αποτελέσματα τους. Άρα η ανάκτηση γνώσεων είναι ανεξάρτητη από τον τρόπο αξιολόγησης για το επίπεδο αυτό. Η κατανόηση γνώσεων και η εφαρμογή γνωστών διαδικασιών σε ένα πρόγραμμα είναι πολλαπλάσιες με την αξιολόγηση ΙΑΠ και στις δύο δραστηριότητες. Ενώ η ικανότητα διάσπασης ενός προβλήματος είναι είτε πολλαπλάσια είτε υπερέχει με την αξιολόγηση ΙΑΠ αυτό εξαρτάται κυρίως από τη δυσκολία που παρουσιάζει η δραστηριότητα καθώς και από τους γνώσεις που έχουν κατακτήσει οι μαθητές. Η ικανότητα αξιολόγησης αυξάνεται και στις δύο δραστηριότητες, ενώ στο επίπεδο δημιουργίας οι επιδόσεις των μαθητών πολλαπλασιάζεται με την αξιολόγηση ΙΑΠ.

Σε γενικές γραμμές συμπεραίνουμε ότι στο επίπεδο της επιμέρους κατανόησης οι επιδόσεις των μαθητών με την ΙΑΠ είτε παρουσιάζουν αυξητικές τάσεις είτε πολλαπλασιάζονται σε όλα τα

επίπεδα της ταξινόμιας Bloom, εκτός από την απομνημόνευση γνώσεων που εκεί παρατηρείται ταύτιση. Επομένως παρατηρούμε ότι καθώς αυξάνεται το επίπεδο στην ταξινόμια Bloom αυξάνονται και οι επιδόσεις στην αξιολόγηση ΙΑΠ.

Για το επίπεδο της προσεγγιστικής κατανόησης οι επιδόσεις των μαθητών με την ΙΑΠ κατά την ανάκληση γνώσεων παρουσιάζει μικρή υπεροχή όποτε συμπεραίνουμε ότι η ιεραρχική αξιολόγηση προγραμματισμού τείνει να προσεγγίζει την παραδοσιακή αξιολόγηση. Η κατανόηση γνώσεων πολλαπλασιάζεται με την ΙΑΠ και στις δύο δραστηριότητες. Οι μαθητές που μπορούν να εφαρμόζουν τις γνώσεις προγραμματισμού, σε ικανοποιητικό βαθμό είναι πάρα πολλοί με την αξιολόγηση ΙΑΠ, καθώς κανένας ή ελάχιστοι μαθητές απαντούν με την παραδοσιακή αξιολόγηση και στις δύο δραστηριότητες. Οι μαθητές που είναι σε θέση να διασπάσουν ένα πρόβλημα σε επιμέρους προβλήματα και να αξιολογήσουν ένα πρόγραμμα πλεονεκτούν ή πολλαπλασιάζονται με την ΙΑΠ, αυτό επίσης εξαρτάται από τη δυσκολία της δραστηριότητας. Στο επίπεδο προσεγγιστικής κατανόησης η ικανότητα δημιουργίας προγράμματος ταυτίζεται και με τους δύο τρόπους αξιολόγησης.

Σε γενικές γραμμές για το επίπεδο της προσεγγιστικής κατανόησης οι δύο τρόποι αξιολόγησης ταυτίζονται στη δημιουργία προγραμμάτων, ενώ στις δεξιότητες της απομνημόνευσης, της κατανόησης, της ανάλυσης και της αξιολόγησης υπερέχει η ιεραρχική αξιολόγηση, ενώ στην εφαρμογή γνώσεων παρατηρούνται μεγάλες αποκλίσεις καθώς πλεονεκτεί στον μέγιστο βαθμό ενώ η παραδοσιακή αξιολόγηση σχεδόν μηδενίζεται.

Όταν το επίπεδο της εκτεταμένης θεώρησης ορίζεται για κάποια ερωτήματα της δραστηριότητας του αντίστοιχου επιπέδου Bloom τότε θεωρούμε ότι μαζί με το συνδυαστικό αποτελούν ένα επίπεδο. Αυτό παρατηρήθηκε στην απομνημόνευση γνώσεων όπου διαπιστώθηκε ταύτιση των αποτελεσμάτων και των δύο τρόπων αξιολόγησης. Ανάλογα αποτελέσματα διαπιστώθηκαν και στην περίπτωση που δεν ορίζεται το 5^ο επίπεδο για όλα τα ερωτήματα μιας δραστηριότητας. Αν θεωρήσουμε σαν ένα επίπεδο το συνδυαστικό και το επίπεδο εκτεταμένης θεώρησης τότε στην περίπτωση αυτή η ΙΑΠ πλεονεκτεί με μεγάλες ή μικρές διαφορές έναντι της παραδοσιακής αξιολόγησης ή συμπίπτει ανάλογα με την πολυπλοκότητα της δραστηριότητας.

Το συνδυαστικό επίπεδο και το επίπεδο εκτεταμένης θεώρησης εξετάζονται ξεχωριστά για τις δραστηριότητες της ανάλυσης και της δημιουργίας προγράμματος ως προς τα συμπεράσματα. Στη διάσπαση προβλήματος στην πρώτη δραστηριότητα πλεονεκτεί η ΙΑΠ ενώ στη δεύτερη δραστηριότητα συμπίπτει με την παραδοσιακή αξιολόγηση. Στη δημιουργία προγράμματος η ΙΑΠ μηδενίζεται, δηλαδή παρατηρείται μεγάλη απόκλιση από την παραδοσιακή αξιολόγηση. Το αίτιο πιθανά να οφείλεται στην κλίμακα αντιστοίχισης καθώς και στη δυσκολία του επιπέδου δημιουργίας.

Για το επίπεδο της εκτεταμένης θεώρησης στην ανάλυση προβλήματος παρατηρούνται χαμηλότερες επιδόσεις της ΙΑΠ έναντι της παραδοσιακής αξιολόγησης με μικρές ή μεγάλες αποκλίσεις σε αντίθεση με τη δημιουργία προγράμματος όπου υπερέχει η ΙΑΠ. Ως εκ τούτου δεν μπορούν να εξαχθούν ασφαλή συμπεράσματα για το επίπεδο εκτεταμένης θεώρησης. Κατά την αντιστοίχιση που έγινε για τις βαθμολογίες των γραπτών από τους βαθμολογητές ως 5^ο επίπεδο καθορίστηκε μόνο το 20 ως βαθμός. Αυτό πιθανά να δημιουργεί αυτή την αβεβαιότητα στα αποτελέσματα για το 4^ο και το 5^ο επίπεδο.

Σε γενικές γραμμές από την παρούσα έρευνα εξάγονται πολύ ενδιαφέροντα συμπεράσματα σχετικά με τις δυσκολίες των αρχαρίων προγραμματιστών, την αξιολόγηση των γνώσεων των μαθητών σύμφωνα με τη μέθοδο της Ιεραρχικής Αξιολόγησης Προγραμματισμού (ΙΑΠ) η οποία βασίζεται στην ταξινόμια SOLO καθώς και αν η ΙΑΠ δίνει τα ίδια αποτελέσματα στην αξιολόγηση με την παραδοσιακή μέθοδο βαθμολόγησης.

Σύμφωνα με τα συμπεράσματα της έρευνας επιβεβαιώνονται ευρήματα από προηγούμενη σχετική βιβλιογραφία. Οι μαθητές δυσκολεύονται να μάθουν να χρησιμοποιούν σωστά σε ένα πρόγραμμα τις δομές επανάληψης και να μετατρέπουν μια δομή επανάληψης σε άλλη, και τους μη αριθμητικούς τύπους δεδομένων (χαρακτήρες, λογικοί), επειδή είναι αφηρημένες έννοιες και, ως εκ τούτου, είναι πολύπλοκες στο να κατανοηθούν (Barbosa et al., 2013; Kiran & Moudgalya, 2015; Lahtinen et al., 2005). Επιπρόσθετα πολλοί μαθητές δυσκολεύονται να γράψουν τη συνθήκη στη δομή επιλογής ή επανάληψης (Lahtinen et al., 2005) ενώ είναι σε θέση να εκτελούν προγράμματα τα οποία περιέχουν δομές επαναλήψεις ή δομές επιλογής. Παρερμηνείες εντοπίστηκαν επίσης σχετικά με τη αρχικοποίηση μιας μεταβλητής, με ποια τιμή και σε ποια θέση χρειάζεται να γίνει η αρχικοποίηση για μεταβλητές όπως είναι το πλήθος ή το άθροισμα ή το μέγιστο. (Lahtinen et al., 2005).

Παρανοήσεις συμβαίνουν όταν οι μαθητές με ανεπαρκή κατανόηση μιας έννοιας, κατασκευάζουν τους δικούς τους κανόνες (Sirkiä & Sorva, 2012). Για παράδειγμα η εντολή εκχώρησης είναι γνωστή στους μαθητές και ξέρουν να τη χρησιμοποιούν αλλά όταν τους ζητήθηκε να δώσουν παράδειγμα με εντολή εκχώρησης για τους τύπους δεδομένων οι μαθητές έδωσαν απλά παραδείγματα τιμών χωρίς χρήση της εντολής εκχώρησης. Ένα άλλο εύρημα που δεν έχει εντοπιστεί στη βιβλιογραφία αποτελεί η μετατροπή της δομής επανάληψης σε δομή επιλογής. Αυτό πιθανά οφείλεται στο ότι οι μαθητές βλέποντας μια συνθήκη ελέγχου σε μια διαδικασία δε μπορούν να αντιληφθούν αν η διαδικασία γίνεται πολλές φορές ή μια φορά και έτσι χρησιμοποιούν δομή επανάληψης.

Στη βιβλιογραφία αναφέρεται ότι πολλοί μαθητές δυσκολεύονται να γράψουν τη συνθήκη στη δομή επιλογής ή επανάληψης (Lahtinen et al., 2005), αυτό το συμπέρασμα επιβεβαιώνεται και στην παρούσα έρευνα. Επίσης σε έρευνες αναφέρεται ότι οι μαθητές έχουν καλύτερη κατανόηση των

έννοιών που σχετίζονται με τον «αλγοριθμικό» παράγοντα παρά με τις έννοιες που σχετίζονται με τον «δομικό» παράγοντα (Cabo, 2015) αυτό όμως δεν επιβεβαιώνεται από την συγκεκριμένη έρευνα.

Η διάσπαση θεωρείται ως η πιο δύσκολη δεξιότητα προγραμματισμού για να κατακτηθεί σύμφωνα με τη βιβλιογραφία (Selby, 2015) αυτό οφείλεται στην έλλειψη εμπειρίας και στην ελλιπή κατανόηση του προβλήματος. Οι μαθητές είναι σε θέση να χρησιμοποιούν την ικανότητα της διάσπασης πιο επιτυχημένα σε καταστάσεις όπου ήδη γνωρίζουν τη λύση ή κατανοούν πολύ καλά το πρόβλημα και αυτό το συμπέρασμα διαπιστώνεται και στην παρούσα έρευνα.

Επίσης οι υψηλότερες επιδόσεις καταγράφονται στο συνδυαστικό επίπεδο στην εφαρμογή γνωστών διαδικασιών (πάνω από 60%) σε ένα πρόγραμμα ενώ στο επίπεδο προσεγγιστικής κατανόησης οι μαθητές σημειώνουν τις υψηλότερες επιδόσεις στο επίπεδο δημιουργίας (27,3%). Το εύρημα αυτό είναι σύμφωνο με τις έρευνες των Meerbaum-Salant et al. (2010), αν και η εφαρμογή γνώσεων θεωρείται παραδοσιακά ευκολότερη για τους μαθητές από τη δημιουργία προγράμματος σύμφωνα με τους Lister et al. (2006) και Lopez et al. (2008), αλλά σύμφωνα με τη λογική της ταξινόμησης, η πολυπλοκότητα του προγράμματος, που εκφράζεται στα επίπεδα SOLO (πολυδομικό ή συσχετιστικό), είναι πιο σημαντική. Μια πιθανή εξήγηση είναι ότι οι κατηγορίες δεν είναι πραγματικά διακριτές με αυστηρά όρια, αλλά επιτρέπουν μια συνεχή ανάπτυξη μεταξύ των γνωστικών κατηγοριών. Άλλοι παράγοντες, όπως ο τύπος της δραστηριότητας ή οι έννοιες που περιλαμβάνει, μπορεί να παραβλέψουν την επίδραση της πολυπλοκότητας της SOLO.

Όταν δίνονται δύο δραστηριότητες στο ίδιο επίπεδο bloom, οι μαθητές τείνουν να παρέχουν απαντήσεις στο ίδιο επίπεδο ΙΑΠ για την κατανόηση και την εφαρμογή, λιγότερο για την αξιολόγηση ενώ διαφοροποιούνται στην ανάλυση. Επομένως σχεδόν ταυτίζεται με την έρευνα των Sheard et al. (2008) που κατέληξε στο συμπέρασμα ότι οι σπουδαστές είναι σχετικά συνεπείς στο επίπεδο SOLO των απαντήσεων τους.

Επίσης οι αρχάριοι προγραμματιστές έκαναν διάφορα συντακτικά και λογικά λάθη. Στο επίπεδο επιμέρους κατανόησης επίπεδο, οι αρχάριοι έδειξαν μια κακή αντίληψη της λειτουργίας της επανάληψης, τόσο για απλούς όσο και για εμφωλευμένους βρόχους (Izu et al., 2016). Το ίδιο συμπέρασμα παρατηρήθηκε και στην παρούσα έρευνα.

Μια άλλη άποψη των Petersen et al. (2011) η οποία εξέτασε τις ταξινομίες Bloom και SOLO στην αξιολόγηση των ερωτημάτων σε εξετάσεις, και υποστηρίζει ότι στις δύο ταξινομίες το επίπεδο εξαρτάται από τα συγκεκριμένα παραδείγματα και τις ασκήσεις που καλύπτονται κατά τη διάρκεια της παράδοσης των μαθημάτων επιβεβαιώνεται και στην αξιολόγηση ΙΑΠ.

Στην έρευνα των Izu et al. (2016) παραλείπεται η κατηγορία εκτεταμένη θεώρηση επειδή δεν είναι δυνατόν να προσδιοριστεί από τις απαντήσεις στις εξετάσεις εάν μπορούν να επεκτείνουν τις

τρέχουσες γνώσεις τους σε άλλα θέματα ή πεδία. Καθώς επίσης, επειδή στα μαθήματα προγραμματισμού είναι δύσκολο το περιεχόμενο, δεν υπάρχει σχεδόν καθόλου χώρος σε αθροιστικές αξιολογήσεις για ερωτήσεις σε αυτό το επίπεδο. Στη συγκεκριμένη έρευνα μπορεί να εφαρμοστεί μόνο για τα επίπεδα αναλύω και δημιουργώ της ταξινόμιας Bloom. Στα υπόλοιπα επίπεδα δεν μπορεί να εφαρμοστεί επειδή οι δραστηριότητες είναι πολύ απλές και δεν γίνεται να επεκταθούν. Τα δύο επίπεδα αυτά ίσως θα πρέπει να εξετάζονται ως ένα επίπεδο. Θα μπορούν να εξετάζονται χωριστά μόνο στην περίπτωση της ανάλυσης και της δημιουργίας.

Η Ιεραρχική αξιολόγηση Προγραμματισμού σε σχέση με την παραδοσιακή αξιολόγηση στο προδομικό επίπεδο συνολικά δίνει καλύτερα και πιο αναλυτικά αποτελέσματα αφού μειώνει τις επιδόσεις των μαθητών σε όλα τα επίπεδα της ταξινόμιας Bloom. Ενώ για τα επίπεδα επιμέρους και προσεγγιστικής ταξινόμησης παρουσιάζει ταύτιση ή υπεροχή έναντι της παραδοσιακής αξιολόγησης. Δεν μπορούν να εξαχθούν ασφαλή συμπεράσματα για τα επίπεδα συνδυαστικό και εκτεταμένης θεώρησης.

Σχετικά με την αξιολόγηση γνώσεων προγραμματισμού ο Elnaffar (2016) υποστηρίζει ότι η χρήση των ταξινομιών (Bloom, SOLO) στην αξιολόγηση, είναι υποκειμενική, απαιτεί οι εκπαιδευτικοί να είναι έμπειροι σχετικά με το κατά πόσο αμφισβητούμενες είναι οι ερωτήσεις τους ενώ οι μετρήσεις, είναι μια ποσοτική προσέγγιση όπου βασίζεται σε εργαλεία που αναλύουν την απάντηση δείγματος για τις ερωτήσεις προγραμματισμού και επιστρέφουν μια αριθμητική τιμή που βοηθά στην ταξινόμηση της δυσκολίας του ερωτήματος που εξετάζεται. Ο υποκειμενικός χαρακτήρας της αξιολόγησης υπάρχει είτε πρόκειται για αξιολόγηση με χρήση ταξινομιών είτε για παραδοσιακή βαθμολόγηση καθώς και οι βαθμολογητές ήταν διαφορετικοί και δεν είχαν προσυμφωνήσει πως θα κάνουν την αξιολόγηση.

Οι περιορισμοί της έρευνας αφορούν αρχικά το δείγμα της έρευνας. Ο αριθμός των συμμετεχόντων ήταν μικρός αλλά ήταν μαθητές από το ίδιο σχολείο και είχαν διδαχθεί από το ίδιο εκπαιδευτικό τις ίδιες δραστηριότητες και την διδακτέα ύλη με την ίδια σειρά. Άλλος περιορισμός αφορά τους βαθμολογητές. Οι βαθμολογητές δεν είχαν συναντηθεί να συζητήσουν πως θα βαθμολογήσουν και να διαμορφώσουν κοινό κριτήριο βαθμολόγησης. Τρίτος περιορισμός είναι η αντιστοιχία της κλίμακας βαθμολόγησης με τα επίπεδα ΙΑΠ. Δεν είχε ελεγχτεί με δοκιμή αν ανταποκρίνεται η κλίμακα στα επίπεδα ΙΑΠ. Βέβαια από ότι φαίνεται δεν δημιουργεί προβλήματα η κλίμακα όπως έχει ορισθεί. Διαπιστώθηκε επίσης ότι τα επίπεδα της Ιεραρχικής Αξιολόγησης Προγραμματισμού ενδείκνυται είναι να είναι τέσσερα επειδή δεν ορίζονται τα επίπεδα Bloom σε όλα τα επίπεδα ΙΑΠ καθώς επίσης επειδή ο βαθμός 20 αντιστοιχεί για τους βαθμολογητές και στο τέταρτο και στο πέμπτο επίπεδο.

Η έρευνα αυτή συμβάλλει στην κάλυψη αυτού του χάσματος με την αξιολόγηση των δεξιοτήτων προγραμματισμού τελειόφοιτων μαθητών Λυκείου σε γνώσεις που αφορούν τύπους δεδομένων, εντολές εκχώρησης, δομές επιλογής και δομές επανάληψης. Παρουσιάζει μια αξιολόγηση τόσο των γνώσεων αλλά και των τρόπων αξιολόγησης που μπορούν να εφαρμοστούν σε μαθητές/φοιτητές με μικρή εμπειρία στον προγραμματισμό. Οι μελλοντικές έρευνες προς αυτή την κατεύθυνση μπορεί να περιλαμβάνουν λεπτομερέστερη εξέταση των σφαλμάτων και ακριβέστερη προσαρμογή της Ιεραρχικής αξιολόγησης προγραμματισμού στην αξιολόγηση προγραμμάτων. Η εφαρμογή της Ιεραρχικής αξιολόγησης Προγραμματισμού από τους εκπαιδευτικούς θα μπορούσε να χρησιμοποιηθεί στην αρχή κάθε τετραμήνου με σκοπό να αναγνωρίσουν τα λανθασμένα νοητικά μοντέλα, και κατόπιν να προσαρμόσουν την διδασκαλία του προγραμματισμού ανάλογα με τις ανάγκες των μαθητών ώστε να βελτιωθούν οι δεξιότητες τους. Τέλος, ελπίζουμε να μπορέσουμε να επεκτείνουμε τα αποτελέσματα της έρευνας για την κατασκευή τυποποιημένων τεστ για δεξιότητες προγραμματισμού.

Φαίνεται ότι η παρουσίαση της χρήσης της Ιεραρχικής αξιολόγησης Προγραμματισμού μπορεί: να ενισχύσει τις προοπτικές των εκπαιδευτικών αλλά και των ερευνητών για αλγοριθμικές εκτιμήσεις στον εισαγωγικό προγραμματισμό, να εμβαθύνει και να διευρύνει την αξιολόγηση των λύσεων των μαθητών και να βοηθήσει να εντοπιστούν οι δυσκολίες στο σχεδιασμό προγραμμάτων και τη βελτίωση τους. Καθώς η γνώση για την κατασκευή βέλτιστων προγραμμάτων πάντα θα αποτελεί πρόκληση, το προτεινόμενο μοντέλο αξιολόγησης μπορεί να χρησιμοποιηθεί για να επηρεάσει αποτελεσματικές πρακτικές στην τάξη.

Βιβλιογραφικές αναφορές

- Βακάλη, Α., Γιαννόπουλος, Η., Ιωαννίδης, Ν., Κοίλιας, Χ., Μάλαμας, Κ., Μανωλόπουλος, Ι., & Πολίτης, Π. (1998). *Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον. Βιβλίο Καθηγητή*. Αθήνα: ΟΕΔΒ.
- Κόμης, Β. (2016). Διδακτική - γνωστική ανάλυση περιβαλλόντων προγραμματισμού προσχολικής και πρώτης σχολικής ηλικίας. *Πρακτικά 8ου Συνεδρίου «Διδακτική Της Πληροφορικής»*, 17–25.
- Κόμης, Β. (2001). Μελέτη βασικών εννοιών του Προγραμματισμού στο πλαίσιο μιας Οικοδομηστικής διδακτικής προσέγγισης. *Themes in Education 2*, (τομ. 2-3), 243–270.
- Μπέλλου, Ι. (2003). Ποιοτική αξιολόγηση μαθησιακών αποτελεσμάτων μαθητών μετά την αλληλεπίδραση τους με εκπαιδευτικό λογισμικό. *2ο Συνέδριο Σύρου Στις ΤΠΕ*, 85–95.
- Μπέλλου, Ι., & Μικρόπουλος, Τ. (2008). Μέθοδος για την Ιεραρχική Αξιολόγηση Γνώσεων Προγραμματισμού. *4ο Πανελλήνιο Συνέδριο Διδακτική Της Πληροφορικής*, 111–120.
- Alaoutinen, S. (2010). Effects of learning style and student background on self-assessment and course performance. *Proceedings of the 10th Koli Calling International Conference on Computing Education Research - Koli Calling '10*, 3408(July), 5–12. <http://doi.org/10.1145/1930464.1930465>.
- Alaoutinen, S., & Nikula, U. (2009). Student self-assessment by counting the times of repetition. *Proceedings - 2009 9th IEEE International Conference on Advanced Learning Technologies, ICALT 2009*, (September), 577–578. <http://doi.org/10.1109/ICALT.2009.149>
- Alexandron, G., Armoni, M., Gordon, M., & Harel, D. (2016). Teaching nondeterminism through programming. *Informatics in Education*, 15(3), 1–23. <http://doi.org/10.15388/infedu.2016.01>.
- Alhazbi, S. (2016). Cognition-based adaptive programming tutoring system. *2016 15th International Conference on Information Technology Based Higher Education and Training, ITHET 2016*, 1–4. <http://doi.org/10.1109/ITHET.2016.7760714>.
- Anderson, L. W., Krathwohl, D. R., Airasian, P. W., Cruikshank, K. A., Mayer, R. E., Pintrich, P. R., ... Wittrock, M. C. (2001). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. New York: Longman.
- Barbosa, L., Lira, O. De, & De, C. S. (2013). Pedagogical architectures and web resources in the teaching-learning of programming, 430–433.
- Bennedsen, J., & Caspersen, M. E. (2012). Persistence of elementary programming skills. *Computer Science Education*, 22(2), 81–107. <http://doi.org/10.1080/08993408.2012.692911>.
- Bey, A., Bensebaa, T., & Benselem, H. (2010). EASEL : Evaluation of Algorithmic Skills in an Environment Learning, 4(6), 64–67.

Biggs, J. (1999). Teaching for Quality Learning at University. *Buckingham, Open University Press.*, 165–203. <http://doi.org/10.1097/00005176-200304000-00028>.

Biggs, J. (2014). Constructive alignment in university teaching. *HERDSA Review of Higher Education*, 1, 5–22. <http://doi.org/10.1046/j.1365-2923.1999.00431.x>.

Biggs, J., & Collis, K. (1982). Evaluating the quality of learning: The SOLO taxonomy (Structure of the Observed Learning Outcome). *New York, Academic Press.*

Bloom, B. (1956). *Taxonomy of Educational Objectives. Handbook 1: Cognitive Domain*. New York: David McKay.

Buckley, J., & Exton, C. (2003). Bloom's taxonomy: a framework for assessing programmers' knowledge of software systems. *11th IEEE International Workshop on Program Comprehension*, 165–174. <http://doi.org/10.1109/WPC.2003.1199200>.

Cabo, C. (2015). Quantifying Student Progress through Bloom's Taxonomy Cognitive Categories in Computer Programming Courses. *American Society for Engineering Education Annual Conference & Exposition*.

Castro, F. E. V. G. (2015). Investigating Novice Programmers' Plan Composition Strategies. *Eleventh Annual International Conference on International Computing Education Research*, 249–250. <http://doi.org/10.1145/2787622.2787735>.

Chakraborty, P., Saxena, P., & Katti, C. (2013). A compiler-based toolkit to teach and learn finite automata. *Computer Applications in Engineering Education*, 21(3), 467–474.

Clear, T., Whalley, J. L., Lister, R., Carbone, A., Hu, M., Sheard, J., ... Thompson, E. (2008). Reliably classifying novice programmer exam Responses using the SOLO taxonomy. *Proc. of the 21st Annual NACCQ Conference of the National Advisory Committee on Computing Qualifications (NACCQ 2008)*, (Naccq), 23–30.

Clements, D. H., & Nastasi, B. K. (1999). Metacognition, learning, and educational computer environments. *Information Technology in Childhood Education*, 3–36.

Corney, M., Lister, R., & Teague, D. (2011). Early Relational Reasoning and the Novice Programmer: Swapping as the "Hello World" of Relational Reasoning. *Conferences in Research and Practice in Information Technology Series*, 114, 95–104.

DiSessa, A. (2000). *Changing Minds: Computers, Learning, and Literacy*. MIT Press.

Eckerdal, A. (2009). *Novice Programming Students β€™ Learning of Concepts and Practise*.

Elnaffar, S. (2016). Using software metrics to predict the difficulty of code writing questions. *IEEE Global Engineering Education Conference, EDUCON, 10–13–April(April 2016)*, 513–518. <http://doi.org/10.1109/EDUCON.2016.7474601>

Ferreira Novais, D., Varanda Pereira, M. J., & Rangel Henriques, P. (2016). Profile Detection

Through Source Code Static Analysis. *Drops-Ibn/6014*, 51(9), 1–9.
<http://doi.org/10.4230/OASICS.SLATE.2016.9>

Fessakis, G., Gouli, E., & Mavroudi, E. (2013). Problem solving by 5-6 years old kindergarten children in a computer programming environment: A case study. *Computers and Education*, 63, 87–97. <http://doi.org/10.1016/j.compedu.2012.11.016>.

Fuller, U., Riedesel, C., Thompson, E., Johnson, C. G., Ahoniemi, T., Cukierman, D., ... Thompson, D. M. (2007). Developing a computer science-specific learning taxonomy. *ACM SIGCSE Bulletin*, 39(4), 152. <http://doi.org/10.1145/1345375.1345438>.

Ginat, D., & Menashe, E. (2015). SOLO Taxonomy for assessing novices' algorithmic design. *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE'15)*, 452–457. <http://doi.org/10.1145/2676723.2677311>.

Gluga, R., Kay, J., Lister, R., & Kleitman, S. (2013). Mastering cognitive development theory in computer science education. *Computer Science Education*, 23(November 2014), 24–57. <http://doi.org/10.1080/08993408.2013.768830>.

Gluga, R., Kay, J., Lister, R., Kleitman, S., & Lever, T. (2012a). Coming to terms with Bloom : an online tutorial for teachers of programming fundamentals. *Fourteenth Australasian Computing Education Conference*, 147–156.

Gluga, R., Kay, J., Lister, R., Kleitman, S., & Lever, T. (2012b). Over-confidence and confusion in using bloom for programming fundamentals assessment. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education - SIGCSE '12*, (August 2015), 147. <http://doi.org/10.1145/2157136.2157181>.

Gomes, A., & Mendes, A. J. (2009). Bloom ' s taxonomy based approach to learn basic programming. *EDMEDIA09 - World Conference on Educational Multimedia, Hypermedia & Telecommunications*.

González-Sacristán, C., Molins-Ruano, P., Díez, F., Rodríguez, P., & Sacha, G. M. (2013). Computer-assisted assessment with item classification for programming skills. *Proceedings of the First International Conference on Technological Ecosystem for Enhancing Multiculturality - TEEM '13*, 111–117. <http://doi.org/10.1145/2536536.2536554>.

Govender, I. (2007). Learning to program, learning to teach programming: pre- and in-service teachers' experiences of an object-oriented programming language. *University of South Africa*. Retrieved from <http://dl.acm.org/citation.cfm?id=1329793>.

Hansen, S., & Eddy, E. (2007). Engagement and frustration in programming projects. *38th SIGCSE Technical Symposium on Computer Science Education*, 39(1), 271–275. <http://doi.org/10.1145/1227310.1227407>.

Hattie, J., & Brown, G. T. L. (2004). Assessment Tools for Teaching and Learning Technical Report #43: Cognitive processes in asTTle: The SOLO taxonomy. *University of Auckland*, (August 2004). Retrieved from <http://tinyurl.com/j2epdkq>.

Hernán-Losada, I., Lázaro-Carrascosa, C. A., & Velázquez-Iturbide, J. Á. (2008). An Educative Application Based on Bloom's Taxonomy for the Learning of Inheritance in Oriented-Object Programming. In *Computers and Education* (pp. 157–166). London: Springer London. http://doi.org/10.1007/978-1-84628-929-3_16.

Hernán-Losada, I., Pareja-Flores, C., & Velázquez-Iturbide, J. Á. (2008). Testing-based automatic grading: A proposal from Bloom's taxonomy. *Proceedings - The 8th IEEE International Conference on Advanced Learning Technologies, ICALT 2008*, 847–849. <http://doi.org/10.1109/ICALT.2008.224>.

Herrero, Á., & Baruque, B. (2015). Self-Assessment Web Tool for Java Programming. *Advances in Intelligent Systems and Computing*, 369, 583–592. <http://doi.org/10.1007/978-3-319-19713-5>.

Honig, W. L. (2013). Teaching and assessing programming fundamentals for non majors with visual programming. *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE '13*, 40. <http://doi.org/10.1145/2462476.2462492>.

Hwang, W. Y., Wang, C. Y., Hwang, G. J., Huang, Y. M., & Huang, S. (2008). A web-based programming learning environment to support cognitive development. *Interacting with Computers*, 20(6), 524–534. <http://doi.org/10.1016/j.intcom.2008.07.002>.

Ivanović, M., Xinogalos, S., Pitner, T., & Savić, M. (2015). Different Aspects of Delivering Programming Courses: Multinational Experiences. *Proceedings of the 7th Balkan Conference on Informatics Conference*, 37:1--37:7. <http://doi.org/10.1145/2801081.2801085>.

Izu, C., Weerasinghe, A., & Pope, C. (2016). A Study of Code Design Skills in Novice Programmers using the SOLO taxonomy. *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, 251–259. <http://doi.org/10.1145/2960310.2960324>.

Kaplan, R. M. (2010). Choosing a first programming language. *Proceedings of the 2010 ACM Conference on Information Technology Education - SIGITE '10*, 30, 163. <http://doi.org/10.1145/1867651.1867697>.

Kaushal, R., & Singh, A. (2012). Automated evaluation of programming assignments. *AICERA 2012 - Annual International Conference on Emerging Research Areas: Innovative Practices and Future Trends*, (July 2012). <http://doi.org/10.1109/AICERA.2012.6306707>.

Kiran, E. L. N., & Moudgalya, K. M. (2015). Evaluation of programming competency using student error patterns. *Proceedings - 2015 International Conference on Learning and Teaching in*

Computing and Engineering, LaTiCE 2015, 34–41. <http://doi.org/10.1109/LaTiCE.2015.16>.

Krathwohl, D. R. (2010). A Revision of Bloom ' s Taxonomy :, *41*(September 2013), 37–41. <http://doi.org/10.1207/s15430421tip4104>.

Lahtinen, E., Ala-Mutka, K., & Järvinen, H.-M. (2005). A study of the difficulties of novice programmers. *ACM SIGCSE Bulletin*, *37*(3), 14–18. <http://doi.org/10.1145/1151954.1067453>.

Ling, L., Hengyang, L., & XianYing, H. (2012). Deeply Research on Diversity Evaluation Mode of “Fundamentals of Programming.” *Information Engineering and Applications. Lecture Notes in Electrical Engineering, Vol 154. Springer, 154*, 385–386. <http://doi.org/10.1007/978-1-4471-2386-6>.

Lister, R., Clear, T., Simon, Bouvier, D. J., Carter, P., Eckerdal, A., ... Thompson, E. (2009). Naturally Occurring Data as Research Instrument : Analyzing Examination Responses to Study the Novice Programmer. *ACM SIGCSE Bulletin*, *41*(4), 156–173. <http://doi.org/http://doi.acm.org/10.1145/1709424.1709460>.

Lister, R., Fidge, C., & Teague, D. (2009). Further evidence of a relationship between explaining, tracing and writing skills in introductory programming. *Proceedings of the 14th Annual ACM SIGCSE Conference on Innovation and Technology in Computer Science Education - ITiCSE '09*, (August 2009), 161. <http://doi.org/10.1145/1562877.1562930>.

Lister, R., Simon, B., Thompson, E., Whalley, J. L., & Prasad, C. (2006). Not Seeing the Forest for the Trees : Novice Programmers and the SOLO Taxonomy. *ACM SIGCSE Bulletin*, *38*, 118–122. <http://doi.org/http://doi.acm.org/10.1145/1140124.1140157>.

López, A. R., & García-Peñalvo, F. J. (2016). Relationship of knowledge to learn in programming methodology and evaluation of computational thinking. *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality - TEEM '16*, 73–77. <http://doi.org/10.1145/3012430.3012499>.

Lopez, M., Whalley, J., Robbins, P., & Lister, R. (2008). Relationships between reading, tracing and writing skills in introductory programming. *Proceeding of the Fourth International Workshop on Computing Education Research - ICER '08*, 101–112. <http://doi.org/10.1145/1404520.1404531>.

Ma, L., Ferguson, J., Roper, M., & Wood, M. (2011). Investigating and improving the models of programming concepts held by novice programmersNo Title. *Computer Science Education*, *21*(1), 57–80.

Makihara, E., Igaki, H., Yoshida, N., Fujiwara, K., & Iida, H. (2016). Detecting exploratory programming behaviors for introductory programming exercises. *IEEE International Conference on Program Comprehension, 2016–July*, 1–4. <http://doi.org/10.1109/ICPC.2016.7503731>.

Mallios, N., & Vassilakopoulos, M. G. (2015). Evaluating Students ' Programming Skill

Behaviour and Personalizing Their Computer Learning Environment Using “ the Hour of Code ” Paradigm, 131–135.

Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (2010). Learning computer science concepts with scratch. *ICER '10: Proceedings of the Sixth International Workshop on Computing Education Research*, 69–76. <http://doi.org/10.1145/1839594.1839607>.

Milková, E., & Petránek, K. (2016). Programming courses reflecting students' aptitude testing and implementing learning style preferences research results. *International Journal of Mathematics and Computers in Simulation*, 10, 218–225.

Mühling, A., Hubwieser, P., & Berges, M. (2015). Dimensions of Programming Knowledge. *8th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, 32–44. <http://doi.org/10.1007/978-3-319-25396-1>.

Olsson, M., & Mozelius, P. (2015). Visualization of concepts and algorithms in programming education - A design theoretic multimodal perspective. *Proceedings of the International Conference on E-Learning, ICEL, 2015-Janua(June)*, 257–264. Retrieved from <http://www.scopus.com/inward/record.url?eid=2-s2.0-84940657052&partnerID=tZOtx3y1>.

Ortiz, J. M. B., Frances, J. G., & Martin, J. G. (2008). A learning knowledge framework for a first graduate course in programming. *2008 19th EAEEIE Annual Conference*, (February 2014), 152–156. <http://doi.org/10.1109/EAEEIE.2008.4610177>.

Papavlasopoulou, S., Giannakos, M. N., & Jaccheri, L. (2015). Designing creative programing experiences for 15 years old students. *CEUR Workshop Proceedings*, 1450, 37–44.

Papert, S. (1983). *Mindstorms: Children, computers and powerful ideas. New Ideas in Psychology* (Vol. 1). [http://doi.org/10.1016/0732-118X\(83\)90034-X](http://doi.org/10.1016/0732-118X(83)90034-X).

Park, T. H., Kim, M. C., Chhabra, S., Lee, B., & Forte, A. (2016). Reading Hierarchies in Code : Assessment of a Basic Computational Skill. *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, 302–307. <http://doi.org/10.1145/2899415.2899435>.

Petersen, A., Craig, M., & Zingaro, D. (2011). Reviewing CS1 exam question content. *42nd ACM Technical Symposium on Computer Science Education*, 631–636. <http://doi.org/10.1145/1953163.1953340>.

Philpott, A., Robbins, P., & Whalley, J. L. (2007). Assessing the Steps on the Road to Relational Thinking. *20th Annual Conference of the National Advisory Committee on Computing Qualifications, Nelson, New Zealand.*, (2006), 286. Retrieved from www.naccq.ac.nz.

Poss, R. “kena.” (2014). How good are you at programming? A CEFR-like approach to measure programming proficiency. Retrieved from <http://science.raaphael.poss.name/programming-levels.html>.

Sahoo, N., & Chin, J. (2010). An elaborate education of basic genetic programming using C++. *Computer Applications in Engineering Education*, 18(3), 434–448.

Saroni, N., Aljunid, S. A., Shuhidan, S. M., & Shargabi, A. (2016). An empirical study on program comprehension task classification of novices. *2015 IEEE Conference on E-Learning, E-Management and E-Services, IC3e 2015*, 15–20. <http://doi.org/10.1109/IC3e.2015.7403479>.

Savić, M., Ivanovi, M., Budimac, Z., & Radovanović, M. (2016). Do students' programming skills depend on programming language? *AIP Conference Proceedings*, 1738, 4–7. <http://doi.org/10.1063/1.4952025>.

Seiter, L. (2015). Using SOLO to Classify the Programming Responses of Primary Grade Students. *Proceedings of the 46th ACM Technical Symposium on Computer Science Education - SIGCSE '15*, 540–545. <http://doi.org/10.1145/2676723.2677244>.

Selby, C. C. (2015). Relationships: Computational Thinking, Pedagogy of Programming, and Bloom's Taxonomy. *Proceedings of the Workshop in Primary and Secondary Computing Education*, 80–87. <http://doi.org/10.1145/2818314.2818315>.

Shargabi, A., Aljunid, S. A., Annamalai, M., Shuhidan, S. M., & Zin, A. M. (2016). Tasks that can improve novices' program comprehension. *2015 IEEE Conference on E-Learning, E-Management and E-Services, IC3e 2015*, 32–37. <http://doi.org/10.1109/IC3e.2015.7403482>.

Sheard, J., Carbone, A., Lister, R., Simon, B., Thompson, E., & Whalley, J. L. (2008). Going SOLO to assess novice programmers. *ACM SIGCSE Bulletin*, 40(3), 209. <http://doi.org/10.1145/1597849.1384328>.

Shi, N., Min, Z., & Zhang, P. (2017). Effects of visualizing roles of variables with animation and IDE in novice program construction. *Telematics and Informatics*, 34(5), 743–754. <http://doi.org/10.1016/j.tele.2017.02.005>.

Shuhidan, S., Hamilton, M., & D'Souza, D. (2009). A taxonomic study of novice programming summative assessment. *Conferences in Research and Practice in Information Technology Series*, 95, 147–156.

Sirkiä, T., & Sorva, J. (2012). Exploring Programming Misconceptions An Analysis of Student Mistakes in Visual Program Simulation Exercises. *12th Koli Calling International Conference on Computing Education Research*, 19–28. <http://doi.org/10.1145/2401796.2401799>.

Taheri, S. M., Sasaki, M., & Ngetha, H. T. (2015). Evaluating the effectiveness of problem solving techniques and tools in programming. *Proceedings of the 2015 Science and Information Conference, SAI 2015*, 928–932. <http://doi.org/10.1109/SAI.2015.7237253>.

Taylor, B., & Kaza, S. (2011). Security injections: modules to help students remember, understand, and apply secure coding techniques. *ITiCSE '11 Proceedings of the 16th Annual Joint*

Conference on Innovation and Technology in Computer Science Education, 3–7.
<http://doi.org/10.1145/1999747.1999752>.

Teodosiev, T. K., & Nachev, A. M. (2015). Role of the algorithm in introductory programming courses. *International Journal of Applied Engineering Research*, 10(7), 16491–16504.

Thevathayan, C., & Hamilton, M. (2015). Supporting Diverse Novice Programming Cohorts through Flexible and Incremental Visual Constructivist Pathways. *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE '15*, 296–301.
<http://doi.org/10.1145/2729094.2742609>.

Thompson, E. (2007). Holistic assessment criteria - Applying SOLO to programming projects. *Conferences in Research and Practice in Information Technology Series*, 66, 155–162.

Thompson, E. (2008). How do they understand? Practitioner perceptions of an object-oriented program : a thesis presented in partial fulfilment of ... *Dissertation, Massey University, Palmerston North.*, (May).

Thompson, E., Grove, H., Luxton-reilly, A., Whalley, J. L., & Robbins, P. (2008). Bloom ' s Taxonomy for CS Assessment. *Reproduction*, 78(January). <http://doi.org/10.1145/1345443.1345438>

Thompson, E., Whalley, J., Lister, R., & Simon, B. (2006). Code classification as a learning and assessment exercise for novice programmers. *Annual Conference of the ...*, 291–298. Retrieved from <http://www.citrenz.ac.nz/conferences/2006/papers/291.pdf>

Van Diepen, N. (2005). “Elf redenen waarom programmeren zo moeilijk is (in English: Eleven reasons why programming is so difficult).” *Tinfony*, 14, 105–107.

Whalley, J., Clear, T., Robbins, P., & Thompson, E. (2011). Salient elements in novice solutions to code writing problems. *Conferences in Research and Practice in Information Technology Series*, 114, 37–45.

Whalley, J., Lister, R., Thompson, E., Clear, T., Robbins, P., Ajith Kumar, P. K., & Prasad, C. (2006). An Australasian study of reading and comprehension skills in novice programmers, using the bloom and SOLO taxonomies. *Conferences in Research and Practice in Information Technology Series*, 52(January 2006), 243–252.

Xu, S., & Rajlich, V. (2004). Cognitive process during program debugging. *Proceedings of the Third IEEE International Conference on Cognitive Informatics*, 176–182.
<http://doi.org/10.1109/COGINF.2004.1327473>.

ΠΑΡΑΡΤΗΜΑ

Συνολική Βαθμολογία με παραδοσιακή βαθμολόγηση για τα επίπεδα ΘΥΜΑΜΑΙ και ΚΑΤΑΝΟΩ

ΑΡΙΘΜΟΣ ΓΡΑΠΤΟΥ	ΘΥΜΑΜΑΙ			ΚΑΤΑΝΟΩ ΔΡ1			ΚΑΤΑΝΟΩ ΔΡ2		
	A	B	Γ	A	B	Γ	A	B	Γ
1	10	19	19	15	0	10	0	0	10
2	8	20	19	20	20	18	20	20	10
3	4	20	15	16	20	15	16	0	0
4	16	16	10	20	0	15	16	20	7
5	10	18	20	20	20	20	19	20	18
6	19	20	20	20	20	20	20	20	20
7	20	20	20	20	20	20	20	20	20
8	20	17	15	6	0	20	0	20	5
9	15	20	20	19	0	19	0	0	0
10	5	20	20	20	20	17	10	0	2
11	20	20	12	20	20	20	20	20	0
12	11	19	17	15	20	20	20	0	5
13	14	20	18	16	20	20	16	20	16
14	20	20	20	15	20	20	19	20	15
15	14	7	10	0	0	0	3	0	0
16	20	20	20	0	0	0	0	0	0
17	15	20	20	0	0	2	5	0	5
18	20	12	12	0	0	0	20	20	5
19	8	0	18	0	20	0	0	20	2
20	0	5	18	20	20	0	0	0	0

21	9	2	5		5	0	0		0	0	0
22	0	8	15		20	20	0		2	0	0
23	0	3	15		20	20	2		20	0	0
24	0	10	15		0	0	0		0	0	0
25	14	12	18		20	20	20		15	20	7
26	10	0	12		6	0	15		10	0	0
27	10	0	15		7	0	5		0	0	5
28	18	10	20		10	0	0		0	0	5
29	0	0	20		20	20	0		20	20	0
30	20	20	20		0	0	0		0	7	0
31	20	15	18		20	20	17		0	0	0
32	15	7	16		7	0	20		0	0	5
33	10	10	18		20	20	10		0	20	2

Συνολική Βαθμολογία με παραδοσιακή βαθμολόγηση για τα επίπεδα ΕΦΑΡΜΟΖΩ και ΑΝΑΛΥΩ

ΑΡΙΘΜΟΣ ΓΡΑΠΤΟΥ	ΕΦΑΡΜΟΖΩ ΔΡ1			ΕΦΑΡΜΟΖΩ ΔΡ2			ΑΝΑΛΥΩ ΔΡ1	ΑΝΑΛΥΩ ΔΡ2
	A	B	Γ	A	B	Γ		
1	20	20	20	5	20	20	5	16
2	5	20	20	20	19	20	20	19
3	20	20	0	0	10	20	16	12
4	20	20	0	10	0	20	5	12
5	20	20	20	20	5	20	19	20
6	20	20	0	20	20	20	17	5
7	20	20	20	20	20	20	20	20

8	20	20	19		20	20	20		20		15
9	20	20	20		20	20	20		12		20
10	20	20	10		20	20	0		6		5
11	20	0	20		20	20	20		20		20
12	0	0	0		20	5	0		0		0
13	0	10	2		20	10	2		18		8
14	15	20	20		0	0	0		15		10
15	0	0	0		20	20	0		6		5
16	20	20	20		0	0	0		8		2
17	20	20	20		0	0	0		5		0
18	0	0	0		20	2	0		4		5
19	0	0	0		0	0	0		2		0
20	20	20	0		0	0	0		5		1
21	5	20	20		20	20	20		17		7
22	5	20	20		10	10	20		5		0
23	5	20	20		20	20	20		4		2
24	0	20	0		10	0	20		9		0
25	20	20	2		20	10	20		8		10
26	20	0	0		20	0	20		7		1
27	20	0	0		20	20	20		5		0
28	0	0	0		0	0	0		5		0
29	20	20	20		20	20	20		17		0
30	20	10	10		20	20	20		14		0
31	20	0	20		20	20	0		13		8
32	0	0	20		20	20	20		6		0
33	20	0	20		20	20	20		5		8

Συνολική Βαθμολογία με παραδοσιακή βαθμολόγηση για τα επίπεδα ΑΞΙΟΛΟΓΩ και ΔΗΜΙΟΥΡΓΩ

ΑΡΙΘΜΟΣ ΓΡΑΠΤΟΥ	ΑΞΙΟΛΟΓΩ ΔΡ1			ΑΞΙΟΛΟΓΩ ΔΡ2			ΔΗΜΙΟΥΡΓΩ
	A	B	Γ	A	B	Γ	
1	0	0	0	0	0	0	10
2	20	10	15	18	20	20	18
3	18	2	0	14	20	0	14
4	0	0	0	0	16	18	0
5	5	20	0	20	10	10	5
6	20	20	17	20	20	10	20
7	20	20	20	20	20	20	18
8	19	20	0	20	10	0	15
9	19	19	20	20	10	20	15
10	20	20	0	18	0	5	0
11	20	20	0	18	20	20	19
12	20	0	0	20	0	0	17
13	16	10	0	14	12	0	15
14	20	0	0	20	20	0	15
15	0	3	3	20	20	0	8
16	0	0	0	0	0	0	10
17	0	0	0	0	0	0	15
18	0	20	2	20	2	0	15
19	20	2	10	18	20	0	2
20	20	20	0	0	0	0	0
21	0	10	0	0	0	0	2

22	20	10	0		0	0	0		0
23	0	0	0		0	0	0		0
24	0	10	0		14	16	0		2
25	0	10	0		14	0	18		14
26	0	10	0		0	0	0		2
27	0	20	0		0	0	0		5
28	0	20	0		0	0	5		20
29	20	20	12		0	0	0		2
30	0	0	10		20	20	0		8
31	20	0	0		18	0	20		17
32	0	20	0		0	0	0		15
33	0	20	0		0	0	18		0