University of Ioannina
School of Education
Department of Primary Education

Postgraduate Programme
«Education Sciences»
Field: Natural Sciences in Education

Digital Learning Objects for teaching computer programming

Master Thesis

Paraskevi Topali
A.M. 404

Committee:
Supervisor: **Anastasios Mikropoulos**
Members: **Costas Gavrilakis**
Sara Lorena Villagrá Sobrino

Ioannina 2017

Abstract

The evolving character of learning technology has fostered changes in the way that the learning material is produced, stored, manipulated and experienced. The result was the development of reusable learning content, termed as '*learning object*', which is widespread in the learning community. Learning objects (LOs) can support flexible approaches for learning and teaching, helping students to become more motivated to learn and to take the learning itself into their hands. They are considered as an important teaching tool in many disciplines and especially regarding the STEM, because they visualize their abstract concepts and they provide direct feedback.

Computer programming, key area of Computer Science and a skill of a great importance for today's digital world involves studying different types of abstract concepts that can be difficult for students to understand. As a result, educators search effective and motivate ways to teach its core concepts and programming languages. Despite the strong benefits, a good deal of interest and a number of large-scale projects, the learning object revolution has not really materialized in Computer Science in general and in computer programming especially.

This dissertation revolves around the need to motivate primary students without any experience in learning programming. In response to this need, five LOs have been developed to help Computer Science teachers while dealing with teaching of basic programming concepts and ideas. Driven by the thought of making them as more intuitive and easy handled and the same time quite flexible and reusable, the LOs have been designed and developed into Scratch environment.

As the results suggest, the proposed LOs seem a useful tool for teachers; they address mainly students without any prior experience in programming, they are aligned to the learning goals, they use authentic scenarios and they can boost students' motivation in an innovative way. The evaluation also found shortcomings hinting at path of future improvements.

Keywords: *Learning Objects (LO), programming, Programming structures, Scratch, learning situations, primary students, students' motivation, Computer science education*

Περίληψη

Ο εξελισσόμενος χαρακτήρας των τεχνολογιών μάθησης έχει μεταβάλλει τον τρόπο με τον οποίο το μαθησιακό υλικό παράγεται, αποθηκεύεται και αξιοποιείται. Το αποτέλεσμα ήταν η ανάπτυξη επαναχρησιμοποιούμενου μαθησιακού περιεχομένου, που ονομάζεται «*μαθησιακό αντικείμενο*» και είναι ήδη ευρέως διαδεδομένο στην εκπαιδευτική κοινότητα. Τα μαθησιακά αντικείμενα καθιστούν τη διαδικασία μάθησης πιο ευέλικτη, παρέχοντας στους μαθητές κίνητρο για μάθηση και εμπλέκοντάς τους πιο ενεργά στη διδασκαλία. Θεωρούνται ως ένα σημαντικό διδακτικό εργαλείο σε πολλούς κλάδους και ειδικά στις λεγόμενες STEM, λόγω της οπτικοποίησης αφηρημένων εννοιών και της άμεσης ανατροφοδότησης που παρέχουν.

Ο προγραμματισμός, βασικός άξονας της Πληροφορικής, αφορά τη μελέτη εννοιών που συχνά είναι δυσνόητες για τους μαθητές. Ως αποτέλεσμα, οι εκπαιδευτικοί αναζητούν πιο αποτελεσματικούς και ευχάριστους τρόπους για να διδάξουν αφηρημένες έννοιες και γλώσσες προγραμματισμού. Παρά τα ισχυρά οφέλη και το ενδιαφέρον, ο αριθμός των μαθησιακών αντικειμένων που έχουν αναπτυχθεί για την Πληροφορική, γενικότερα, και τον προγραμματισμό, ειδικότερα, είναι περιορισμένος.

Η παρούσα εργασία αφορά τη βελτίωση της διδασκαλίας βασικών εννοιών του προγραμματισμού σε μαθητές δημοτικού χωρίς προηγούμενες γνώσεις στο αντικείμενο. Για την ενίσχυση του ενδιαφέροντος τους και την παράλληλη υποστήριξη των εκπαιδευτικών, σχεδιάστηκαν και αναπτύχθηκαν πέντε μαθησιακά αντικείμενα στο Scratch, ένα περιβάλλον εύκολο και ευχάριστο στη χρήση που εξασφαλίζει την απαιτούμενη ευελιξία και επαναχρησιμοποίηση των μαθησιακών αντικειμένων.

Βάσει των αποτελεσμάτων, τα προτεινόμενοι μαθησιακά αντικείμενα αναδείχτηκαν ως χρήσιμο εργαλείο για τους εκπαιδευτικούς· απευθύνονται κυρίως σε αρχάριους μαθητές αναφορικά με τις γνώσεις στον προγραμματισμό, ευθυγραμμίζονται με τους μαθησιακούς στόχους, χρησιμοποιούν αυθεντικά σενάρια και είναι καινοτόμα. Η ανατροφοδότηση περιλαμβάνει και προτάσεις για μελλοντικές βελτιώσεις.

Λέξεις-κλειδιά: *Μαθησιακά Αντικείμενα (ΜΑ), προγραμματισμός, δομές προγραμματισμού, Scratch, διαδικασίες μάθησης, μαθητές δημοτικού, κίνητρα μάθησης, Πληροφορική*

Acknowledgements

I could not be happier for having chosen such an interesting theme for my dissertation. I would like to thank all the people who have been part of this stage of my life.

With immense gratitude, I would like to thank wholeheartedly my supervisor, Professor Anastasios Mikropoulos for his perspicacious guidance, his constructive feedback and all the opportunities he offered me during these years; without him this work couldn't have happened. He was always available to help me, inspire me, motivate, encourage and continued support me, despite the inevitable distance.

Additionally, I would like to thank my committee members Professors Costas Gavrilakis and Sara Villagra Sobrino for their guidance, their general interest in my work and their incisive inputs to refine the thesis.

Special thanks to Mr. Anastasios Ladias for his insightful comments regarding my work and his help regarding my LOs. Together with Ms. Ioanna Bellou, they have made everything go smoothly with my research sample.

I am also very grateful to Professor Dimitris Mavridis who has provided critical advice and guided me regardless the distance: Spain-France. Additionally, I would like to thank Professor Sopfia Hantzis for her valuable contribution proof-reading my dissertation.

My long-suffering friends have also been a great support; I would like to thank Ioanna, Elena, Stavri, Dionisia, Vasilis, Petra, Loizos and Chris who have suffered many times my thesis's stories, who were patient with my restricted time schedule and who found creative ways to spend our free time forgetting our University obligations. Special thanks to Giorgos and Erkan for the unforgettable funny moments at the EarthLab & GSIC Lab.

Special thanks to my best friend Emmanouil for being my "critical friend", for supporting and believing in me more than I actually believe in myself.

Last but not least, I would like to thank my family who inspires me and motivates me. Thank you for your great patience during the development of this thesis!

Contents

List of Figures

List of Tables

Introduction

Nowadays, technology permeates every aspect of our daily life changing the way we learn, work and communicate. The evolution from a "*society of information*" to a "*society of knowledge*", confirms that the citizens of the 21st century need to be aware of a variety of new types of literacies to meet the current demands of society (Kellner, 2000). The effective use of computational tools, the design and the implementation of solutions, the creativity and other skills, often collectively termed "*21st Century Skills*," are now in the spotlight (Weintrop & Wilensky, 2015). In the field of education, the stakeholders have to take into account the complexity of the contemporary, technologically enhanced world. The educators should upskill the students by providing the structure and the tools which will enable them to use their own intelligence and knowledge to maximum capacity (Bannan-Ritland, Dabbagh & Murphy, 2002).

The meaningful use of technology to maximize the students' learning experience by *matching the needs of a given set of learners to learn a given content using a given set of learning tools* (Cohen & Nycz, 2006, p. 24) led to the evolution of e-learning and technology enhanced learning. Online learning opportunities, full-time online schools, digital/open educational resources and blended learning opportunities increase educational productivity by promoting learning according to the learner's pace, lower cost of digital learning materials, and flexibility in time and place (Cohen & Nycz, 2006). This evolving character of learning technology has fostered the search for methods and technologies which transform the way in which the learning material is produced, stored, manipulated, and experienced leading to reusable learning content and permitting the learner to take the learning itself into their hands (Downes, 2001; Weller, 2007). As a result, a range of digital resources have been developed. These resources are available via the web, can include text, images, charts, video, audio, data sets, can cover a wide range of subject areas and aim towards specific learning objectives.

A general label that is now applied to these reusable digital learning resources is *learning objects* (LOs) (Goodyear & Retalis, 2010). Understanding what a LO is, is easier than defining it; a LO can be seen as a *knowledge "package"* (Cohel & Nycz, 2006, p. 29), attempting to deliver learning experiences and support the virtual education technologically and pedagogically (Arturo, Jaime, Álvarez, Francisco & García, 2009; Ritzhaupt, 2010), focusing at the same time on reuse and automation of searching, selection and composition of educational contents and activities *(*Sicilia & Sánchez-Alonso, 2006). It is a "*pedagogical resource, a type of computer-based instruction grounded in the object-oriented paradigm, usually delivered over the internet, meaning that any number of people can access and use it simultaneous*" (Begosso, Begosso & Begosso, 2016, p. 2).

In recent years the use of LOs for teaching various thematics is of great interest. Most repositories include LOs that appeal to various disciplines such as Mathematics, Physics, Chemistry, Life Sciences, etc. Most of them have been developed by the initiative of universities and governmental incentives, for example the BBC Learning, Bozeman Science, Harvard Open Learning Initiative, Khan Academy, Learn NC, Merlot, Math Open Reference, MIT Open Courseware, Nobel Prize Education, Smithsonian Education, among others. LOs offer numerous benefits to learners (Downes, 2000; Shank, 2005; de Salas & Ellis, 2005; Shepherd, 2006; Ritzhaupt, 2010; Kay, 2012). The main benefit mentioned is the direct feedback students receive, which is an important factor in meeting the student's desire for success and creating motivation for learning. In fact, they provide an explanation to abstract causes rendering these immediately observable phenomena (Cavus & Ibrahim, 2004; Shank, 2005; Kay, 2012). In general, they:

- promote personalized learning meeting the user's needs who controls the learning process
- are flexible, enriching the learning process in terms of time and space
- provide direct feedback involving users in the learning process
- help users to understand complex ideas and abstract concepts through visualization
- are reusable reducing the cost of developing new educational resources
- provide the possibility of sharing the educational material

Motivation

Computer programming involves a high degree of problem-solving activity and is perceived as an essential skill for today's digital world. It builds skills that can easily be applied to other disciplines and fields such as the ability of communicate one's ideas and thoughts in a relevant context that is extremely important; it cultivates cognitive skills and is the basis for the development of strategic thinking and finding solutions to problems (Papert, 1980; Pirolli & Recker, 1994; European Schoolnet, 2014). Programming has been incorporated in education field, from primary schools until universities. Often enough though, it seems that students find it difficult to deal with its abstract concepts (Jenkins, 2002; Boyle, 2003; Pickard, Chalk & Jones, 2003; Matthiasdottir 2006; Barreto & Benitti, 2012; Rahmat, Shahrani, Latih, Yatim, Zainal & Rahman, 2012; Burbaite, Damasevicius & Stuikys, 2013), with the educators to search ways to teach programming ideas and languages in an effective and motivate manner (Begosso et al., 2016). How to motivate learners in order to enhance the learning of programming is of great importance, so game-based learning, educational robotics and pleasant programming environments has been perceived as effective means for helping learners to construct knowledge.

Quite recently, reusable digital learning resources, the so-called *learning objects,* have been developed and applied in many disciplines in general with many positive contributions. The problem with the LOs which have been designed for programming is that the number encountered is limited, the most of which appeal to secondary and higher education students (Adamchik & Gunawardena, 2003; Boyle, 2003; Pickard et al., 2003; Matthíasdóttir, 2006; Narasimhamurthy & Al Shawkani, 2009; Villalobos, Calderón, & Jiménez 2009; Wu, Qian, Bhattacharya, Guo & Hu, 2011; Burbaite et al., 2013; Jimoyiannis et al, 2013; Matthews, Hin & Choo, 2014; Begosso et al., 2015).

The existence of bibliographic void in the reviewed literature regarding the LOs for programming which address *primary students* led to the preparation of this dissertation. The main goal of this work is ***how to help teachers to motivate and teach the basic programming ideas to primary students without any prior experience in programming, using LOs.***

Objectives:

1. ***To create LOs that meet the general characteristics of Learning Objects.***

The various definitions accompanying the LOs deal with some features, which have become commonly accepted. According to the literature review, the LOs should come along with high levels of *reusability, granuality, discoverability, assesibility, interoperality, adaptability, durability, generativity, manageability*. The challenge is to create LOs that meet these characteristics, based the same time of the definition given by Mikropoulos & Bellou (2016)

2. ***To create LOs that promote the students' motivation***

Motivation has been center of attention among teachers throughout the years because it constitutes the backbone of learning process; keeping students motivated is a key issue if we want them to learn (Jenkins, 2001; Dişlen, 2013). Concerning the programming concepts, find a way to boost students' interest is of great importance since its incomprehensible and abstract character discourages the students.

3. *To create LOs using Scratch.*

We used as a tool, to develop our LOs and to overcome the above challenges, Scratch environment. Scratch is widely widespread in the education community because is attractive to the students, intuitive and easily learned (Armoni, Meerbaum-Salant & Ben-Ari, 2015). Additionally it gathers the characteristics of LOs allowing its projects to be reusable (they can be re-used in various situations such as courses, classes, etc.), of interoperability (they work in different environments), of manageability (they can be easily modified.

The problem with Scratch is that though it gathers the characteristics of the LOs, there were not found many LOs developed in Scratch.

A general conceptual map follows, to visualize the context, the aimed objectives, the original contribution and the evaluation of the current dissertation (Fig. 1).

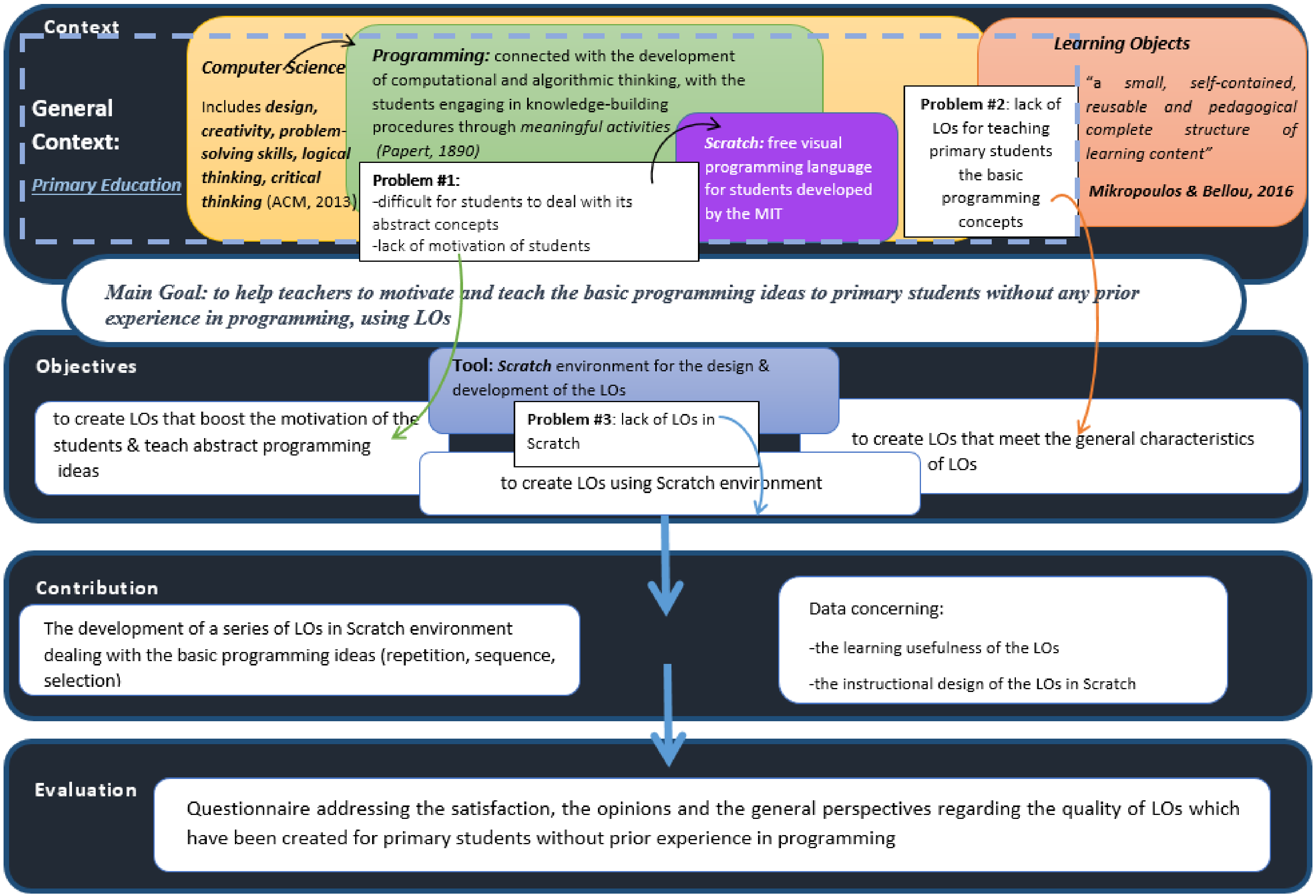


Fig. 1. General schema of the dissertation including its context, the aimed objectives, the original contribution and the evaluation

The instructional design theory that we applied to the Learning Objects we developed was ADDIE (**A**nalyze, **D**esign, **D**evelop, **I**mplement, **E**valuate) (Fig. 2).

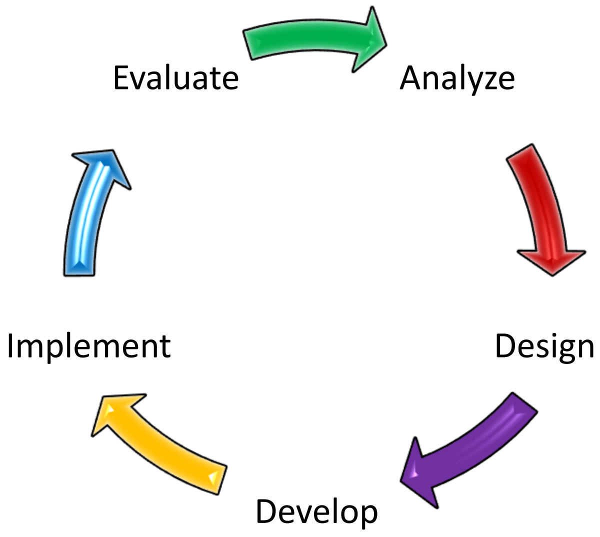


Fig. 2. ADDIE instructional design theory

Concerning each stage, at the phase of the Analysis, we were dealing with the following questions: *who are your learners, what are the learning objectives, what learning style should be applied* etc. The Design phase includes the design of the structure of the LOs. We attempted our created LOs to stand alone, follow a standard instructional format, to be small and have clear learning goals based on the LOs' definition of Mikropoulos & Bellou (2016) we follow. The next phase concerns the digital development of the LOs and the Implementation one the delivery system in which the LOs run; in our case our delivery system was the Scratch environment. Last but not least comes the Evaluation phase where we received valuable feedback for our LOs from Computer Science professors.

The whole process was organized as the next figure indicates (Fig. 3):

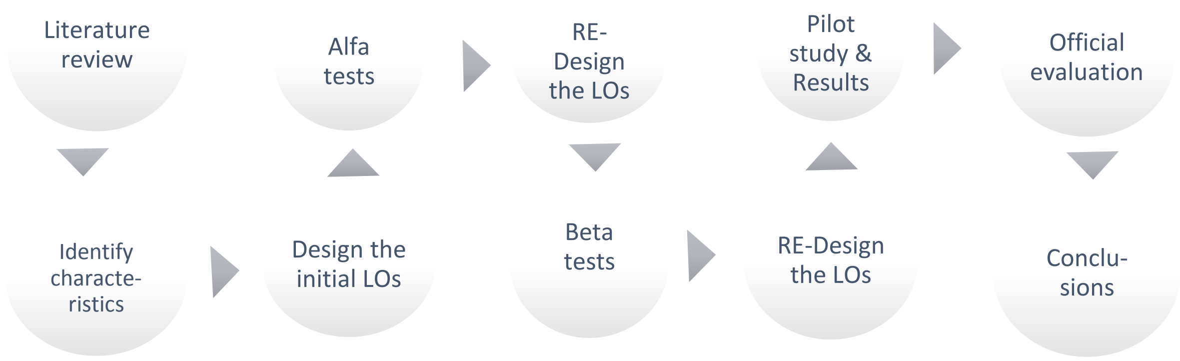


Fig. 3 The envisioned steps of our work

Speaking more precisely, at the beginning a Literature Review was conducted investigating LOs developed to help students understand Programming concepts. Once we finished this step, we tried to identify which are the characteristics of these LOs and in which parameters the creators paid more attention. Then we started the design of our LOs; after Alfa and Beta test and Re-Designs in between, we ended up in a final version of 5 Learning Objects. A pilot study was conducted and then the official evaluation followed among with the valuable feedback of our subjects.

The structure of the rest of this document is as follows:

- The next Chapter is divided into subchapters; the first one presents details about the Computer Science and its importance into the educational context and the second one describes what the Learning Objects are.
- In Chapter 2, a literature review about Learning Objects which have been developed for teaching programming is exposed.
- Chapter 3 dives into our proposal; the architecture and the details of the proposed Learning Objects
- Chapter 4 presents the Methodology followed thorough the dissertation
- To check the viability of the proposed Learning Objects, an evaluation was carried out. The evaluation and the results are exposed in Chapter 5.
- Finally, Chapter 6 contains the conclusions obtained from this work, the limitations and the future work.

Chapter 1

Theoretical Part

1.1. Computer Science as a discipline

1.1.1. Introduction

The scientific and technological advancements have made a big impact on the current lifestyle and the knowledge-based economy. The educators are challenged continually to enhance students' skills and to provide them with dynamic learning experiences that address the needs of learners in today's society (European Schoolnet 2015; The Partnership for 21st Century Learning, 2015; Moyer, 2016). Attempting the alignment of the educational reality with this new information-based and highly technological reality, Sciences were grouped under the umbrella of STEM (Science, Technology, Engineering and Mathematics) (National Science Foundation)[1], seeking the active participation of the students and the cultivation of innovation (CSLNet). As a result, the current learning environments focus on the development of skills which are related to problem solving, creativity and critical thinking (Paige 2009; Moyer, 2016).

In 2007, the OECD Centre for Educational Research and Innovation (CERI) launched the "New Millennium Learners" project exploring the impact of digital technologies on students (OECD, 2009) and concluded that the learning outcomes of the 21st century should include traditional courses and modern contents taught in an interdisciplinary framework following a holistic approach (Dagienė, 2011). The desirable skills suggested by OECD are of social, technological and cognitive aspect and are divided into four categories: a) functioning in socially heterogeneous groups; b) acting autonomously; c) using tools interactively; d) thinking (a "cross-cutting" competency) (Dagienė, 2011). The North Central Regional Education Laboratory (NCREL) identifies a framework of skills grouped into four categories: digital-age literacy, inventive thinking, effective communication, and high productivity (Pacific Policy Research Center, 2010). The ISTE (International Society for Technology in Education) conducted a project asking thousands of teachers and school leaders to identify the most important knowledge and skills for today's digital-age students. The results demonstrated

[1] https://www.nsf.gov/

as crucial concepts those of creativity, imagination and innovation, communication and collaboration, critical and logical thinking, problem solving and the development of efficient and productive technology (ISTE NETS for Students, 2007; UNESCO, 2012). The Partnership for 21st Century and Trilling & Fadel categorized the prized 21st century skills as follows: a) Critical Learning and Innovation skills (creativity and innovation, critical thinking and problem solving, communication and collaboration), b) Information, Media and Technology skills (information literacy, media literacy) and c) Life and Career skills (Trilling & Fadel, 2009; Pacific Policy Research Center, 2010). The European Parliament and the Council (2006) recognized eight key competences for Lifelong Learning which among other digital competence is mentioned. Ferrari (2013) provides a framework (DIGCOMP) where digital competence is summarized as (p. 5-6):

- Information: identify, locate, retrieve, store, organize and analyses digital information, judging its relevance and purpose.
- Communication: communicate in digital environments, share resources through online tools, link with others and collaborate through digital tools, interact with and participate in communities and networks, cross-cultural awareness.
- Content-creation: Create and edit new content (from word processing to images and video); integrate and re-elaborate previous knowledge and content; produce creative expressions, media outputs and programming; deal with and apply intellectual property rights and licenses.
- Safety: personal protection, data protection, digital identity protection, security measures, safe and sustainable use.
- Problem-solving: identify digital needs and resources, make informed decisions on most appropriate digital tools according to the purpose or need, solve conceptual problems through digital means, creatively use technologies, solve technical problems, update own and other's competence.

To sum up, the organizations converge that today's students and tomorrow's employees are asked to think critically and creatively, to possess problem-solving skills, to be comfortable working in groups, to connect with the world around them and to apply the available technology aiming to research and communicate (Paige 2009; Dagienė, 2011). The education sector made efforts to integrate these skills of the 21st century in several courses, the educational policy makers, however, argue that the teaching of Computer Science can greatly improve the abilities of the 21st century students due to the great impact that it has on the real world (Dagienė, 2011).

Computer Science as a discipline includes design, creativity, problem-solving skills, collaboration, logical thinking, critical thinking (ACM, 2013). In a nutshell, it involves skills that can be found in other cognitive areas that reply to authentic problems and respond to social processes (ACM, 2013; Goldberg, Grunwald, & Lewis, 2013). O Jeannette Wing attempting to underline the major contribution of Informatics mentioned the *computational thinking (*Wing, 2006). The students who have developed computational thinking are capable to organize and analyze their data in a logical raw; to represent their data with models and simulations; to think about the same problem from the scope of abstraction and decomposition; to become more creative while dealing with the solution of a problem recognizing that a single solution can be applied to many other seemingly different problems (Wing, 2006; Barr, Harrison & Conery, 2011; Computing at School Working Group, 2012). The same time, while the students are dealing with complex and open-ended problems, they develop their self-confidence and their flexibility (Barr et al., 2011).

The *computational literacy* has become the fourth kind of literacy along with reading, writing and arithmetic (Belshaw, 2011), the knowledge of which makes students able to not only simply apply the new technologies but also to express themselves and apply their ideas through them (Berry, 2013). Resnick & Papert (1995) paralleled the computational literacy with the proficiency in a foreign language; as the fluency in a foreign language not only indicates the use of vocabulary but also the formulation of a complex idea or the telling of an engaging story, analogously computational literacy describes the ability to create and construct using technological tools and not only the knowledge of how to use those tools (Resnick, 2002).

1.1.2. *Computer Science & Information and Communication Technology*

Computer Science or Informatics, along with the technologies it involves, constitutes an important discipline in the knowledge society (CSTA, 2011). The term refers to the science of the study of computers and algorithmic processes, including their principles, equipment (hardware) and software (software), their applications and the impact they have on society (CSTA Curriculum Committee, 2009; Computing at School Working Group, 2012). It is a practical subject that provides insights into a broad range of systems -not only ones which include computers (Computing at School Working Group, 2012; Berry, 2013), and it belongs to STEM approach (Science, Technology, Engineering and Mathematics), to these sciences aim is to educate students by involving them creatively in the learning process (Computing at School Working Group, 2012).

Computer Science as a school subject was introduced to schools in the early 1980s with the teaching content to vary according to the social requirements of each era. Starting with the main axis “teaching about computer” and stepping onto the stage “teaching with computer”, today, in the digital society of the 21st century, Information Technology is seen as the language of technology, as the means by which we can approach and understand the digital world (Dagienė, 2011). Its contribution lies in the development of the computational and the algorithmic thinking by designing and solving problems resulting on the understanding of the human behavior and human limits (Berry, 2013; European Schoolnet, 2015).

According to the Greek Curriculum (ΔΕΠΠΣ, 2011), the purpose of introducing Computer Science to the compulsory education (Primary, Secondary) is " *pupils to gain an original but coherent and comprehensive understanding of the basic computer functions, in a perspective of technological literacy and recognition of the Information Technology and Communication, while developing broader skills of critical thinking, ethics, social behavior and mood for activation and creation on both an individual level and in collaboration with others*" (p. 412).

Alongside with the inclusion of Computer Science, the Information and Communication Technologies (ICT) have been quite recently been introduced as well to the educational processes. Often a confusion is reflected on the curricula between the subject of Computer Science and the ICT. UNESCO in 2002 attempting to differentiate the terms proposes the following definitions:

- Informatics (Computing Science): the science dealing with the design, realization, evaluation, use, and maintenance of information processing systems, including hardware, software, organizational and human aspects, and the industrial, commercial, governmental and political implications of these.
- Information and Communication Technology (ICT): the combination of informatics technology with other, related technologies, specifically communication technology.

Clarifying these two concepts, Computer Science constitutes an autonomous scientific field and a separate discipline, while ICT is a tool of the subject of Informatics; a tool that lets students and teachers to search, explore and express. Computer Science teaches a student how to be an effective author of computational tools (i.e. software), while ICT teaches how to be a thoughtful user of those tools (Computing at School Working Group, 2012; European Schoolnet, 2015). ICT, in short, focuses on the creative and productive use of technology and how to implement computer tools to solve problems and not how to build computer tools (Mikropoulos, 2013).

Education systems worldwide first integrated Computer Science and later ICT. Teaching Computer Science courses with the use of ICT is crucial, as their combination can ensure the formation of learning environments that cultivate skills (cognitive, emotional, and social) which are necessary for the complex knowledge society (UNESCO, 2012).

1.1.3. Computer Science as a school discipline

Computer Science is an established discipline at the collegiate and post-graduate levels (CSTA, 2009). The most educational systems worldwide, introduce Computer Science as an independent discipline in secondary education, while in primary education the teaching of Computer Science principles is foreseen through other courses, e.g. Mathematics, Physics. Exception are the Netherlands, Switzerland, Poland, and Slovakia, where Computer Science is an independent discipline of compulsory attendance. In general, the Computer Science course aims at developing *algorithmic/ computational thinking* with each curriculum to focus more or less on these topics:

- fundamentals of Information Technology,
- solving problems,
- algorithmic thinking,
- algorithms & data structure,
- programming and programming languages,

Concerning the United States, the data vary since it is more difficult for all states to converge on a common curriculum. By 2012 Computer Science courses were elective in secondary education. According to the data of EdSurge Reseach and Code.org, 2013, Chicago began a five-year plan based on which CS becomes a compulsory subject in all schools of the state. In September 2015 the Information was incorporated into all secondary schools in the State of Arkansas and in June 2016, San Francisco extended the teaching of Information Technology from preschool to 12th grade.

Specifically, the following table (Table 1) lists the grades in which Computer Science was first introduced in the curriculum (either as a compulsory or elective subject, or as a concept taughed through other courses) and a short description of the learning outcomes (Guerra, Kuhnt & Blöchliger, 2012 ; Khenner & Semakin, 2014; European Schoolnet, 2015).

Table 1. CS as a school subject among the world

	Country	Grade	Theme details
Europe	***Austria***	Grade 9 (mandatory)	Introduction to software, hardware, operating systems, data privacy, and capabilities of computers.
	Bulgaria	Grade 9 (mandatory)	Programming, algorithmic problem solving and representing information through abstractions (e.g. models and simulations)
	Denmark	Grade 7-10 (inside other lessons) Grade 11-13 (elective)	Grade 7-10: (Physics and chemistry) Knowledge about simple programming and transmission of data, programming languages and skills of programming simple digital solutions (Math) to enhance systematic and abstract thinking with specific guidance
	Finland	Grade 8-9 (elective)	Pupils learn to use different technical devices and programs
	Hungary	Grade 13-14 Grade 15-16	Though Informatics is a stand-alone subject, schools may choose to integrate it for the purposes of lessons. Grade 13-14: Algorithms, Logo or a similar programming language, Basic commands Grade 15-16: Algorithm design and analysis, Problem solving
	Ireland	no national program at primary level Grade 13-15 (elective)	Some primary school teachers may use Scratch programming in the instruction of shape and space in mathematics.
	Germany	Grades 7-8/ 6-12 (mandatory)	*Grades 7 to 10:* Introduction to hardware and software, Terminology and assembly of computer, Foundation of ICT, Solving problems using computers. *Grades 10 to 11*: Basic Concepts of Information Technology, Project work using standard software, Computer science and society, Computer Networks, Structures and algorithms & their implementation, Structured data types, Modeling information table *Grade 12:* advanced programming and fundamentals
	Lithuania	Grade 9-12 (mandatory)	*Grade 9-12*: introduce elements of algorithms and programming.

	Grade 11-12 (elective)	*Grade 11-12*: Information theory, Logic, Algorithms
Montenegro	Grade 6	
Netherlands	K-12 (mandatory)	The curriculum consists of the following themes: Informatics in perspective, Terminology and skills, Systems and their structures, Usage in a context
Poland	Grades 1-12 (mandatory)	
Serbia	Grade 5-8 (elective)	Programming, interactive graphics and graphic design
Slovakia	Grades 1-4 Grades 5-9 Grades 10-13	Information around us, Communication through digital technologies, Procedures, problem solving, algorithmic thinking, Principles of the functioning of digital technologies, computer programming
Slovenia	Grade 7, 8, 9 (elective) Grade 10 (mandatory)	*Grade 7*: Editing Text *Grade 8*: Computer Network *Grade 9*: Multimedia *Grade 10:* Procedures, problem solving, algorithmic thinking, Principles of the functioning of digital technologies, computer programming
Spain	upper secondary education (elective)	Basic Concepts of Information Technology, Project work using standard software, Computer science and society, Computer Networks, Structures and algorithms, Structured data types
Switzerland	Grades 1 to 9 (mandatory)	Curriculum includes an introduction to Algorithm, Software, Hardware, operating systems, Data privacy, and capabilities of computers.
UK (England)	K-12 (mandatory)	Computing is a distinct subject in school curricula but schools are free to teach it as an integrated subject or stand-alone. Teaching as an integrated subject is more common at primary level. The curriculum consists of the: Fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation; analyze problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems; evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems

America	***Canada***	Grades 10-12 (elective)	Understanding computers, introduction to programming & computers and society.
	USA	Grade 9-12 (elective)	Procedures, problem solving, algorithmic thinking, Principles of the functioning of digital technologies, computer programming
Asia	***Russia***	Grade 1-4 (inside other lessons) Grade 5-9 (mandatory)	Practical Use of Computers, Algorithmic thinking, creativity, problem solving, automatics, fundamentals of operating systems, programming languages
	Japan	Grade 7-9 (mandatory)	Practical Use of Computers
	India	Grade 9 (elective)	
Africa	***Israel***	Grade 10-12 (elective)	Algorithmic thinking, creativity, problem solving, automatics, fundamentals of operating systems, programming language C# & Java
	Tunisia	Grade 10-13 (mandatory)	Basic Concepts of Information Technology, Computer science and society, Computer Networks, Structures and algorithms, Structured data types, algorithmic thinking, creativity, problem solving

1.1.3.1. Computer Science & Primary Education in Greece

In Greece, Computer Science as a cognitive object introduced firstly at the secondary education level, while in 1997 the Greek Pedagogical Institute designed the "*Unified Framework of CS Curriculum*" which is referred to the introduction and integration of Computer Science into all stages of the education, as an attempt to include Computer Science in primary education as well. In 2001, according to the Curriculum, Computer Science were taughed in compulsory education as an independent subject in the secondary education and were introduced at the primary education through other subjects[2] (ΔΕΠΠΣ, 2001). During the last decade, workshops were integrated at extended schools for introducing the principles of Computer Science after school. Its official introduction in primary schools as an independent subject begun in 2010 as "*ICT*" (ΠΣ μαθημάτων ΔΣ με ΕΑΕΠ, 2010).

The Greek Pedagogical Institute states as a general purpose of teaching Computer Science in Primary Education "*to familiarize pupils with the basic functions of the computer and to come into contact with its various uses as a supervisory teaching tool, as a cognitive -*

[2] To the curriculum of 2001 is referred to *ICT* with the term "*Computer Science",* where we can observe a confusion between the two terms

investigative tool and as a tool of communication and research, using appropriate software, especially open source software" (ΔΕΠΠΣ, 2011, p. 412).

The general objectives of teaching Computer Science are grouped into the following three areas: a) *Knowledge and Methodology*, b) *Cooperation and Communication* and c) *Science and Technology in everyday life*. The axes of the learning objectives focus on developing technological skills (with the understanding of fundamental concepts of the computer system), cognitive skills (with the use of the computer as a tool of research, communication and learning), problem-solving skills (with use of the two skills above but more critically, innovatively and creatively) and social skills (with the understanding of the social impact of the contemporary digital culture) (Παιδαγωγικό Ινστιτούτο, 2011).

The specific objectives of the introduction of CS in primary school are, the students to:

- be familiarized with the various uses of the computer: a) as supervisory teaching tool, b) as cognitive - investigative tool and c) as a tool of communication and research
- acquire skills and develop the ability to use ICT safely, reflectively towards the available information, with confidence and creativity in order to ensure their full integration in the Society of Knowledge and Information (ΠΣ μαθημάτων ΔΣ με ΕΑΕΠ, 2010)
- to use the digital technology, the communication tools and the network services "*for accessing, managing, integrating, evaluating, creating and communicating information in order to solve problems*"

In the first four grades of the primary education, the curriculum focuses on the three following cognitive axes (ΔΕΠΠΣ, 2011): *I learn the computer*, *I play and learn with the computer*, *I e-communicate*. The aim is the students to get to know the basic parts of the computer and its everyday uses, to learn to use simple applications while playing and to learn to utilize the computer in order to communicate via Internet. In the following two grades the curriculum aims to develop *computational literacy;* the pupils start to edit text and images, to process files, to use and create charts, to learn the principles of programming, to create simple procedures in easy programming environments, to search information via www, to present their works using simple multimedia applications, to communicate with others via e-mail and in general to use computers in realistic situations.

1.1.4. Introducing Computer Programming in educational settings

Following the current era, which seeks the construction of new contents and not the use of ready-made ones, the familiarization of the students with *programming* is placed at the center of attention of the educational policy. In essence, the term programming refers to activities which involve students in creation of small programs and codes (European Schoolnet, 2014).

Computer Programming (or coding) is one of the cognitive axes of Computer Science. It is the process of developing and implementing various sets of instructions to enable a computer to perform a certain task, solve problems and provide human interactivity (European Schoolnet, 2014). It concerns all the procedures and instructions which must be given in a logical raw, like the steps of a cooking recipe, attempting to solve a computer problem (European Schoolnet, 2014). It is directly connected with the development of computational and algorithmic thinking, with the students engaging in knowledge-building procedures through meaningful activities (Papert, 1890).

The introduction of programming in the educational settings occurs at the global level (European Schoolnet, 2015). At European level, according to the European Schoolnet data (2015), 16 European countries have integrated programming in their curriculum before 2015. Finland and Belgium started teaching it in 2015 (Finland has defined programming in the core curricula for 2016) and Poland in 2016 after a general reform of the course of Computer Science. Estonia and Slovakia have already integrated programming planning at all levels of school education, and in 2014 England as well by making it a compulsory subject. France, Poland, Belgium, Finland, Spain, Denmark and Portugal, within the period 2015-2016, integrated it in primary Education- either as a cognitive axis of the course Informatics or ICT or following a cross-curricular approach through other subjects (for example, Mathematics, Physics). At global level, Israel recognizing quite early the value of the "programming", it has made it a compulsory subject at all education stages. In the United States, according to programming advocate Code.org, only one in 10 U.S. schools teach children to code. On December 8, 2014, President Obama became the first US president to write a line of computer code, during the second annual "Hour of Code" underlying the importance of knowing how to code. Marking May 2016, the Education, Culture, Sports, Science and Technology Ministry of Japan has decided to make computer programming a compulsory subject at primary schools in fiscal 2020, followed by middle schools in fiscal 2021 and high schools in fiscal 2022. In the Greek educational system, "programming" referred to upper grades of primary education as a cognitive axis of the course Informatics (DEPPS, 2011).

The recognition of the value of educational programming is not something new, since already in 1965 Wallace Feurzeig and Seymour Papert interested in the development of a programming language (Calao, Moreno-Leon, Correa, & Robles, 2015), created a programming language especially for children. Though for many years programming was in the educational landscape as a cognitive object, around the end of the 1990s the interest in programming fell apart and in the more recent times new visual programming languages, such as Alice, Kodu and especially Scratch and BYOB have reawakened the interest of the educational community (Kafai & Burke, 2013; Calao et al., 2015).

Teaching programming is seen as a long-term solution for the gap between the number of jobs with technological content on one hand, and the people who can fill them. The integration of programming into school curricula attempts to upskill the future workforce and to empower young students by cultivating logical thinking and problem solving skills (Bruce & Freund, 2008; European Schoolnet, 2015). In the 19th and 20th century, the challenge was to understand the natural world and the energy sources and the 21st to understand of data organization and information for the development of new knowledge. Through programming, students develop problem-solving skills, cultivate logical and algorithmic thinking, they become more creative and flexible, they develop imagination, they become familiar with design strategies which are useful in areas irrelevant to Computer Science (Papert, 1980; Bruce & Freund, 2008; Resnick et al., 2009; Sterling & Kittross, 2015). According to Mitch Resnick, proficiency in programming is isobaric with proficiency in foreign language (Resnick, 2002).

1.1.4.1. Computer Programming & Primary Education

In primary education, in particular, the introduction of students to the principles of programming is based on *Block-Programming Environments* (BPEs), such as Scratch, BYOB, Alice (Price, 2015; Weintrop & Wilensky, 2015). The Block-Programming Environments are a variety of visual programming languages that leverage a primitives-as-puzzle pieces metaphor. In these environments the users create codes simply by “*dragging*” and matching blocks of commands, procedure which is similar to the creation of a puzzle (Fig. 4), and then they receive feedback concerning their construction (if what they have created is valid or not). These commands match with each other in meaningful ways and usually are grouped into categories with different characteristics (e.g. different color, a different structure, etc.) regarding their features (Weintrop & Wilensky, 2015). This approach of teaching programming eliminates syntax mistakes and makes children focus their attention on the problem they want to solve and not on the mechanism of programming itself.

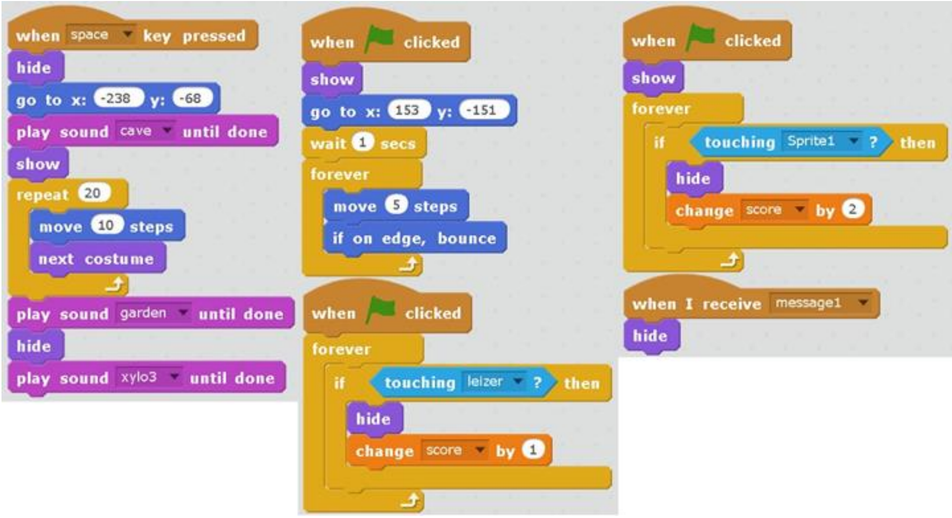


Fig. 4 Three code samples in Scratch environment

These environments seem more attractive for the teachers and the students due to:

i. their pleasant appearance and the simplicity of their commands (Lewis, 2011), with all the tools and commands to have the form of blocks
ii. the fact that they are linked to the interests of students, allowing them to program games, small applications, digital stories (Price, 2015)
iii. the fact that programming is easier with blocks-based programming tools than with the text-based programming ones. BPE eliminate the risk of errors when writing a program and the need to memorize the names of procedures (Price, 2015)
iv. the visualization they offer that makes abstract concepts, such as initialization, variables and synchronization, more understandable (Meerbaum-Salant, Armoni & Ben-Ari, 2011)
v. the feedback and the scaffold students receive during programming (Weintrop & Wilensky, 2015).

The reviewed literature reveals that in primary education, in particular, and in all educational stages, in general, approaching programming concepts is a difficult task because the teacher cannot simply transfer their own knowledge (Fetaji, Loskovska, Fetaji, & Ebibi 2007). Its incomprehensible and abstract character discourages the students (Jenkins, 2002; Boyle, 2003; Pickard, Chalk & Jones, 2003; Matthiasdottir, 2006; Burbaite, Damasevicius, & Stuikys, 2013) with the ones who are involved in the education to seek more efficient teaching ways. As a result, a range of digital resources have been developed with specific learning goals by the name of *learning objects*.

1.2. Learning Objects

1.2.1. Define the term "learning object"

More than a few words have been produced while trying to give a clear picture of what Learning Objects (LO) are all about. Yet confusion is apparent in the literature, as doesn't exist a consistent definition (Sosteric & Hesemeier, 2002). Describing what a LO is, isn't an easy task; although its educational role is well understood in the e-learning community, what it is in its essence is not. As a result, in the literature exists a number of definitions and many are still developing (Stuikys, 2015). The references and the definition efforts lead to confusion, which generally occurs when attempting to match the existing vocabulary with a new technological conception (Shank, 2005). The various definitions represent the interests, the scope, the concerns of its proponents and the existing variety of design techniques in the field of e-learning (Polsani, 2003; Rehak & Mason, 2003). ASTD & SmartForce underline this difficulty in the article "A Field Guide to Learning Objects" stating: "*it may surprise you that no single learning object definition exists within the e-learning industry. Learning objects are different things to different e-learning professionals. In fact, there seems to be as many definitions as there are people to ask*" (ASTD & SmartForce, 2002, p.3).

The concept of LOS is not a recent innovation since they have been on the educational agenda for several years now with attempted definition already from 1998 (Sosteric & Hesemeier, 2002). The way that the educators and those who are involved in the educational technology do define and categorize the changes though, a fact that complicates the acceptance of a unique definition (Bratina, Hayes & Blumsack, 2002). The term "*learning object*" comes from Wayne Hodgins, who coined it in 1994 (2002) and it has its origins from the OOP (object-oriented programming) according to Wiley (2002) and Friesen (2003). It describes an object that is designed for a specific purpose and can be categorized using metadata; this is the basic idea behind learning objects: *small building educational resources that can be reused many times in different learning environments* (Wiley, 2002, p.3).

In the literature the most widely known definitions are those of Wiley (2002) and IEEE (2002), though already in 1998 James L'Allier applies a three-part definition: a learning objective, a unit of instruction that teaches the objective, and a unit of assessment that measures the objective (Wiley, 2002). Learning Object Metadata Working Group of IEEE Learning Technology Standards Committee defines it as a learning object "*as any entity, digital or non-digital, which can be used, reused or reported during technology supported learning*" (2002, p.1) that includes (2005):

a) multimedia content,

b) instructional content,
c) learning objectives,
d) instructional software and software tools,
e) persons, organizations, or events referenced

According to David Wiley, the definition of IEEE is general and vague. He proposes instead the following definition: "*a learning object is any digital resource that can be reused to support learning* (Wiley, 2002, p. 3). He provides a framework narrow enough to include a homogeneous set of things and broad enough to include the estimated 15 terabytes of information available on the Internet (Wiley, 2002). This definition has been considered as well wide as it has been limited only to the exclusion of non-digital resources comparing the definition of IEEE (Polsani, 2003).

Along with these two well-known definitions, in the literature can be found plenty definition, since the authors of each article related to LO tend to provide a definition based on their standards (Wiley, 2002). In 2000 Hodgins and Conner intended to describe LOs resembling them to LEGO: "*small units that can be fitted together any number of ways to produce customized learning experiences*" (Hodgins & Conner, 2000). In 2002 Hamel & Ryan-Jones suggest the following definition: "*learning objects are small but full educational segments teaching content, which can be suitably combined to create larger sections of teaching organization*" (Morgan, 2011). This definition is adopted recently by Morgan (2011) due to its pedagogical and didactic approach. Sosteric & Hesemeier (2002) considering that the context for which the learning object is constructed is of great importance and should be included in the definition, give the following definition: "*a learning object is a digital file (image, movie, etc.) intended to be used for pedagogical purposes, which includes, either internally or via association, suggestions on the appropriate context within which to utilize the object*" (p. 2). The Gunawardena & Adamchik defined LO as "*an integrated module containing the main text, examples, evaluation questions and any other supporting materials*" (Adamchik & Gunawardena, 2003, p. 2). Polsani focusing on the value of reuse describes LOs as "*an independent and self-contained learning content unit, which can be reused in many teaching contexts*" (Polsani, 2003, p.4). Sicilia, Garcia, Sanchez-Alonso & Rodriguez (2004) characterize as learning objects "*digital entities -Internet resources- that represent information digitally encoded and readable by a computer" (p.2093),* while Cochrane in his work follows Wiley's definition and adds "*learning objects are interactive digital resource illustrating one or more concepts*" (Cochrane, 2005, p.33). Cohen & Nycz state that LO "*are types of knowledge objects in the sense that their goal is to provide knowledge in support of an* *associated learning objective"* (Cohen & Nycz, 2006, p. 29). One year later, Chiappe, Segovia & Rincon (2007) give the following definition: "*a digital self-contained and reusable entity, with a clear educational purpose, with at least three internal and editable components: content, learning activities and elements of context. The learning objects must have an external structure of information to facilitate their identification, storage and retrieval: the metadata*" (p. 675). Balatsoukas, Morris & O'Brien note that in general most definitions of learning objects are based on the following features: reusability, cognitive targeting, independence (Balatsoukas, Morris & O'Brien, 2008). The Farha (2009) describes LO "*as digital resource based on multimedia elements which are reusable and can be aggregated with other learning objects to form larger pieces of content*" (p. 2). One the latest definitions, the one that this work follows, is the one of Mikropoulos and Bellou (2016), which defines learning object as "*a small, self-contained, reusable and pedagogical complete structure of learning content*". It consists of content data (e.g. multimedia data) and information objects (concepts) and it is a learning component in a learning environment (Μικρόπουλος & Μπέλλου, 2016).

McGreal (2004) gathered and grouped the definitions of LOs into five categories:

1. anything and everything
2. anything digital, whether it has an educational purpose or not
3. anything that has an educational purpose
4. only digital objects that have a formal educational purpose (Koper, 2003; Sosteric & Hesemeier, 2003; Polsani, 2003)
5. only digital objects that are marked in a specific way for educational purposes

The existence of similar terms referring to LOs make, as well, the acceptance of one general definition harsh; terms which are slightly different but have similar meanings (Mc Grey, 2004; Parrish, 2004). Merrill for instance, uses the term "*knowledge objects*" (2000) defining them as "*a set of fields (containers) for the components of knowledge required to implement a variety of instructional strategies*" (p.1), while the repositories ARIADNE and MERLOT use the terms "*pedagogical documents*" and "*online learning materials*". Gibbons, Nelson & Richards, (2002) use the term "*instructional object*", Frieser (2001) the term "*educational objects*", Koper (2001) the term "*unity of study*", Downes the term "*resources*' (2005), Cohen & Nycz (2006) the terms "*raw media elements of e-learning*" and "*knowledge object*", while in the literature there is also the term "*reusable learning object (RLO)*" (Mow, 2002; Polansi, 2003). Muzio, Heins & Mundell (2002) and Escobar, Reyes & Van Hilst (2014) use the term *e-Iearning objects* and define them as "*small learning component that can be reused several times within a learning context*" (Escobar, Reyes & Van Hilst, 2014, p. 1).

Apart from the different terms used, there have been reports mentioning updated forms of LOs. Thus, Boyle refers to "*compound objects*" (2003) and describes them as "*the new object created by the composition of two or more independent learning objects which provide pedagogical richness not available through simple objects and a significant basis for re-purposing*" (p. 51). Villalobos, Calderón & Jiménez introduce the term "*interactive learning objects* (ILO)" (2009) which are interactive visualization tools that differ from other learning objects in: a) graphical visualization aspect, b) the existence of an interaction objective and c) the importance of the environment where they are used (Villalobos, Calderón & Jiménez, 2009). Morales, Leeder, & Boyle talk about "*generative learning objects* (GLOs)" which are based on the idea of separating the learning design from the surface instantiation of a learning object. This surface separation from the content leads to reusable forms where tutors can load content (template-based approach) (Morales, Leeder & Boyle 2005). In 2013 Burbaite, Stuikys & Damasevicius introduce the "*physical learning objects* (PLO)" with robots as main representative. According to the author, physical learning objects extend the notion of traditional learning objects beyond the virtual domain (e-content, web page) to a physical domain (robot hardware and physical processes that are demonstrated by the hardware) (Burbaite, Stuikys & Damasevicius 2013). Štuikys (2015) on the other side, searching how the discipline of Computer Science can be taught more effectively, refers to "*smart learning objects* (SLO)" which are the upgrading generative learning objects.

1.2.2. Features of Learning Objects

The various definitions accompanying the LOs deal with some features of LOs, which have become commonly accepted unlike with the definitions themselves. Gürer (2013) attempted to gather and describe the features to which the most of the papers are referred (LOM, 2000; Bannan-Ritland, Dabbagh, & Murphy, 2002; Goodyear & Retalis 2010). More precisely:

- ***Reusability:*** refers to the reuse of LO in different contexts. Through reusability, the diminution of the cost and the effort which are needed for development of a resource is attempted (Gürer, 2013). It holds a key position in the education where those who are involved in teaching prefer to use tools, materials, resources and lesson plans which are reusable, sharable and exchangeable (Goodyear & Retalis 2010).

- ***Granularity:*** refers to the size of a LO both in terms of content and functionality, but also in terms of multimedia data (LOM, 2000). The determination of the size of a LO is directly related to its reuse. According to Hodgins, a LO should be so small that can be easily used in different contexts, yet so large to be meaningful for the students (Hodgins, 2004). So far there is not a clear definition of the preferable size of the LO, which often is simply described as a "small" unit. According to Wiley, the learning object has to be just smaller than the whole lesson (ASTD & SmartForce, 2002; Wiley, 2002). The Wisconsin Online Resource Center states that only content units with a duration of 2-15 minutes can be called learning objects (Chitwood & Bunnow 2002, p. 2).
- ***Interoperability:*** refers to the ability of LO to work in different operating systems or operating systems, hardware or browsers and with different material**.**
- ***Accessibility:*** refers to the ability of LO to be accessed anywhere. Accessibility is linked with the appropriate description of the metadata.
- ***Adaptability:*** refers to the ability of LO to be modified in order to align with the needs and requirements of every user.
- ***Discoverability:*** refers to the easy identification of the LO. It is also linked to the appropriate description of the metadata
- ***Durability:*** refers to the sustainability of the LO and its functionality regardless the changes of hardware or updates of the software.
- ***Generativity:*** refers to the possibility of aggregation of the LO automatically in order to meet the individual needs of learners.
- ***Manageability:*** refers to the ability of the LO to be manageable (updated, revised and combined).

1.2.3. Types of Learning Objects

Most learning objects include one or more multimedia elements such as sound, video, animation, graphics, text, and some kind of user interaction (Shank, 2005). Shepherd (2006) distinguishes three types of learning objects: *integrated, informal, practice & review*, so the learning objects can vary from mini-tutorials and mini case studies with supportive information (integrated), to demonstrations / models (informational), to games / simulations and to drill-and-practice exercises (practice & review) (Balci & Inceoglu, 2006). Kay (2012) points out that there are at least two types of learning objects: a) structured learning objects and b) open-ended learning objects. Structured learning objects typically deliver short sequences of information and then test students' knowledge or allow limited practice with the concepts being learned. Open-ended learning objects use a problem-based format where students explore and test what-if scenarios to discover relationships and / or improve understanding of specific concepts (Kay, 2012). The first ones are more directed while the second provide more possibilities.

Most of them have been developed by the initiative of universities and governmental incentives, for example the BBC Learning, Bozeman Science, Harvard Open Learning Initiative, Khan Academy, Learn NC, Merlot, Math Open Reference, MIT Open Courseware, Nobel Prize Education, Smithsonian Education, among others. In order to simplify the task of developing learning objects, the University of Wisconsin Online Resource Center (Wisc-online, pp.220-221, 2001) proposes some guidelines to be followed before creating or using a learning object. Accordingly, a LO:

- Shows clear purpose
- Reflects a specified learning preference
- Supports the competency at the appropriate level
- Helps learners understand the concept
- Can be applied to courses in different subject areas
- Can be applied to different programs of study
- Can be grouped into larger collections of content
- Requires interaction on the part of the learner
- Can stand alone
- Contains all information and materials needed by learners to complete the activity
- Is easy to use for the learner
- Applies appropriate principles of good practice
- Applies appropriate learning object principles

LOs offer numerous benefits both to users and to those who are involved in their development (Downes, 2000; Shank, 2005; de Salas & Ellis, 2005; Balci & Inceoglu, 2006; Ritzhaupt, 2010; Kay, 2012). They:

- ✓ promote personalized learning meeting the user's needs who controls the learning process
- ✓ are flexible, enriching the learning process in terms of time and space
- ✓ provide direct feedback involving users in the learning process

- ✓ help users to understand complex ideas and abstract concepts through visualization
- ✓ are reusable reducing the cost of developing new educational resources
- ✓ provide the possibility of sharing the educational material.

1.2.4. Learning Objects' Repositories

The large number of the LOs, their availability via Web and what has been called "the learning object economy" -the ability to access, share and reuse a variety of learning resources (Downes, 2001), led to the creation of repositories for LOs (Learning Objects Repositories) (Currier, Barton, O'Beirne & Ryan, 2004). Learning Objects Repositories are online libraries that enable storage, handling and sharing of learning objects. Their main purpose is to collect high-quality learning resources preferably of small size (Duncan, 2003) to provide a better quality of learning, by facilitating the sharing and reuse of the educational resources.

According to the end-users needs, a functional repository of LOs has to be characterized by (Lehman, 2007):

- ✓ easy access to learning objects (Lehman, 2007; McGreal, 2007)
- ✓ content compatibility (mainly for repositories which have been created for specific thematic areas or sectors)
- ✓ variety of LO including simple and more advanced ones (McGreal, 2007) in order to meet the requirements of each user
- ✓ ease of sharing the information
- ✓ reusability of learning objects
- ✓ possibility of edit, process and combine the LO
- ✓ ease of accessibility for users with disabilities

Several categories of repositories of LOs are mentioned in the literature (Lehman, 2007; McGreal, 2007; Roy, Sarkar & Ghoseet, 2010). According to McGreal, the learning objects repositories are divided into: a) content repositories, b) linking or metadata repositories, c) hybrid repositories that host content and links to external learning objects (McGreal, 2007). Roy et al. (2010) accept as learning objects repositories digital libraries which include both learning objects and their metadata. Lehman distinguish repositories to a) general, b) discipline specific, and c) commercial/ hybrid (2007). Among the most known repositories of learning objects which either include LO, either metadata and links or both, stand out ARIADNE, Merlot, EdNA, WTCS, MIT OCW, Scientix, NSDL, PhET U. of Colorado and Health

Education Assets Library (Friesen, 2001; Sosteric & Hesemeier, 2002; Currier, Barton, O'Beirne & Ryan, 2004; McGreal, 2007; Roy, Sarkar & Ghoseet, 2010).

ARIADNE is a European digital library developed for the production and storage of reusable learning content. Merlot (Multimedia Educational Resources for Learning and Online Teaching) is also one of the most famous open repositories, designed for teachers and students. It includes online educational resources along with reviews and evaluations. EdNA (Education Network Australia), also provides a basis 20,000 learning objects useful for educational use in Australia. Wisconsin Online Resource Center (WTCS), is a digital repository that contains over 1,000 learning objects for higher education. Health Education Assets Library is a digital library that provides free access to educational resources regarding the Health Sciences. OCW MIT (OpenCourseWare) is a free educational repository of open resources and digital NSDL Library (The National Science, Mathematics, Engineering, and Technology Education Digital Library) have been created to meet the needs of pupils and teachers providing access to learning resources and promoting learning through personal involvement. Scientix and PhET of University of Colorado promote and support a collaboration among STEM offering LO of high quality.

In Greek educational reality, there is Photodentro (PHOTODENTRO LOR), the Greek National Learning Object Repository for primary and secondary education. It constitutes a core part of the Greek Ministry of Education digital infrastructure for educational content for schools hosting small, semantically and functionally autonomous, self-contained and reusable, tagged with educational metadata and open to all educational resources (Megalou & Kaklamanis, 2014).

1.2.5. Learning Objects' Metadata

Gathering the learning objects in repositories is not sufficient itself for their direct identification and their efficient use (Currier et al, 2004); to conduct fruitful researches and to obtain the desired material in a quick and effective way, learning objects are defined with detailed and comprehensive information: *metadata*. The more accurate, consistent and sufficient the metadata are, the more traceability, reusability and interoperability they ensure for the learning objects, while metadata of poor quality make the learning resource virtually invisible in the repository (Currier et al, 2004; Roy et al., 2010).

The term *metadata*, which is formed by the Greek word *meta* (= besides, after) and the Latin word *data*, describes the existence of data, about other data or information about other information (NISO, 2004; Lehman, 2007; Roy et al., 2010). There are keywords and collection descriptions which allow the easy identification of learning objects analogously like the traditional cataloging of libraries where for each book there is an information package for its quick detection and identification; they can describe the whole collections, a simple resource or a component of the resource (Currier et al, 2004; NISO, 2004; Roy et al., 2010). According to IEEE Learning Technologies Standard Committee (2002) the purpose of metadata is to facilitate the "*search, evaluation, acquisition and use"* of learning objects in libraries and repositories. McGreal & Roberts (2001) define metadata as *"tags or descriptions that systematically describe many aspects of a learning object with technological and pedagogical features*" (p. 1). A more complete definition is given by NISO (2004) where metadata is "*structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use, or manage an information resource"* (p. 1).

In the literature several distinctions of metadata can be found. Hodgins (2002), for instance, distinguishes metadata in *objective* and *subjective*. The objective ones describe the physical properties, the date, the author, or the cost of the learning objects while the subjective ones are the more varied and valuable attributes of a learning object, and are determined by the person or group who creates the metadata. They concern the organization of the context, the content and points of view, such as whether or not the LO was effective as component of the learning procedure. NISO (2004) distinguishes three types of metadata: *descriptive*, *structural* and *administrative*. The first ones describe a resource for purposes such as discovery and identification and they can include elements such as title, abstract, author, and keywords. The second ones, in case of a combination of learning objects, include information and details about this combination, for example, how pages are ordered to form chapters. The last ones provide information to help manage a resource, such as when and how it was created, file type and other technical information.

As metadata are of great importance, the processes involved in their creation and management cannot be arbitrary. The creation of metadata is based on internationally recognized open standards. The most commonly used standard for metadata is the LOM (Learning Object Metadata) (IEEE, 2002) that has been established by the IEEE Learning Technologies Standard Committee. The LOM aims to develop accredited standards, recommended practices and guides for learning technology (Roy et al., 2010). It includes more than 60 elements classified into 9 categories (General, Life Cycle, Meta-Metadata, Technical, Educational, Rights, Relation, Annotation, Classification), each one of them containing metadata for various aspects of a LO, including its technical characteristics and rights, as well as educational and instructional features (Solomou, Pierrakeas & Kameas, 2015). Besides LOM, a well-known standard for describing learning objects is the Dublin Core, which is engaged in the development of interoperable online metadata and includes 15 different fields of metadata (Title, Creator, Subject, Description, Publisher, Contributor, Date, Type, Format, Identifier, Source, Language, Relation, Coverage, Rights) (Roy et al., 2010). This standard, however, focuses primarily on metadata concerning the technical aspect of a learning object, in contrast to the LOM which includes educational metadata as well.

Educational metadata are a metadata category, usually in XML coding, which aim to describe a learning object that is used to support the learning process. Educational metadata included information such as (Roy et al, 2010; Solomou et al, 2015):

- Learning Resource Type
- Interactivity Level & Type
- Semantic Density
- Intended End User Role
- Context
- Typical Age Range
- Difficulty
- Typical Learning Time
- Description
- Language

Other known metadata standards which take into account both the technical and the educational data of a learning objects are: a) the IMS Metadata Global Learning Consortium, b) the SCROM Metadata, and c) the CanCore (Roy et al., 2010).

Chapter 2

Literature Review

Recently, the use of learning objects for educational purposes is getting great attention. Most of the repositories host learning objects which appeal to various disciplines such as Mathematics, Physics, Chemistry, Health Sciences, etc. Regarding CS in general and programming in particular, the number of learning objects encountered in repositories is limited.

This paper explores the creation of learning objects for teaching CS and especially for programming. Available literature has been searched on several academic electronic databases: ScienceDirect, Scopus (Elsevier), Google Scholar, Springer, Taylor & Francis, IEEE Xplore, ACM. The search was limited to studies in the English language with the following keywords:

- /*learning objects*/ and /*computer programming*/ OR
- /*learning objects*/ and /*computer science*/ OR
- /*creating learning objects*/ and /*computer science*/ OR
- /*developing learning objects*/ and /*introductory programming concepts*/

The reviewed literature shows that approaching programming concepts is a difficult task and the fact that programming requires direct involvement of the learner, because the teacher cannot simply transfer their own knowledge, makes it even more difficult (Fetaji et al., 2007). Its incomprehensible and abstract character discourages the students of any educational stage to understand and use basic concepts (such as data structures and algorithms) and to create programs that solve concrete problems (Jenkins, 2002; Boyle, 2003; Pickard et al, 2003; Matthiasdottir, 2006; Burbaite et al., 2013). As a result, the ones involved in education seeking more efficient teaching approaches they started developing and using Learning Objects.

Boyle (2003) describes the development of learning objects and their integration into CS courses of higher education. He explains the design principles of LO in order to improve the learning outcomes of Java programming concepts in Computer Science, Mathematics and Communications Technology of the Metropolitan University of London in collaboration with the Bolton Institute in the UK. The 14 LOs developed are characterized as complex learning objects (compound learning objects) and consist of two or more independent LOs linked to form a compound one: a base object, which is a textual description of a single topic in an HTML page, has optional expansions to other objects that deal with that topic in more detail, such as additional text-based descriptions, multimedia examples or simulations, examples of code, or applets showing examples of code in action. (Boyle, 2003). Through a virtual learning environment the easy sharing of the developed LOs is ensured. These LOs are autonomous, reusable and pedagogically rich and they contain multimedia elements (Pickard, Chalk & Jones, 2003). Regarding the results as presented in two articles, the use of these LOs for teaching programming had a positive impact; the students were engaged and they seemed to understand easier the fundamentals of programming noting a significant cognitive improvement (Boyle, Bradley & Chalk, 2004). Pickard et al. underline the positive responses of the students using the LOs (2003).

Adamchik & Gunawardena (2003) focus on how students can learn easily and more efficiently the basics of programming. The authors, while searching for a mechanism to support individualized learning needs, adjusted learning objects in the programming courses they offered at the Carnegie Mellon University. According to them, a LO is defined as *an integrated module containing the core text, code examples, review questions, supplementary material and programming labs*. During their pilot project, they taught the Java language to students who had basic level knowledge of programming using learning objects. Their learning objects are a composition of specific highlighted and annotated content, code examples as well as quick pre-tests to make sure that students possess the skills necessary to understand the material. Additionally, realizing that the students needed also an online place to discuss, they moved a step forward by creating an interactive content display environment (Adaptive Book) where the learners were able to discuss course-related content and the teachers to create and manage learning objects. According to their findings, the students were more concentrated during the learning process, understood the concept, practiced programming skills and generally approached the whole idea with enthusiasm.

Matthíasdóttir (2006) talks about the Codewitz project. Searching an effective way to learn and teach programming, the researchers created interactive learning objects. Under the umbrella of the Codewitz project, six universities participated in order to improve the way the knowledge about programming was delivered. The Codewitz learning objects are web-based standalone visualizations of programming tasks or code examples built for clear specific learning goals; they have an area for input/output from the student, execution that shows step by step what is going on and an area for Memory and Conditions. Many of the objects also have an explanation area. They mainly focus on C++ language and Java presenting details or smaller parts of a program code. Their evaluation of the effectiveness was made by the students of Reykjavik University who used the learning objects as additional material of the teaching programming procedure. As the survey's findings show the students mentioned the usefulness of the learning objects. Nevertheless, a better integration of learning objects is needed since only a part of the students seemed to use extensive material outside the classroom.

Wu, Qian, Bhattacharya & Hu (2011) present the design and development of reusable learning objects to enhance the learning of programming. The LOs were developed attempting to ensure visualization and interactivity during the learning process of programming. The researchers call their LOs as *live LOs* and define them as "*a web-based learning component with specific learning goals and clear programming problem statements, employing learning models to convey knowledge, facilitate understanding and enable just-in-time code practicing and diagnosis*" (Wu et al, 2011, p. 362). The LOs were constructed based on the Interactive Learning Model to present the introductory knowledge of programming. Through a five-stage learning process, the users were practicing their algorithmic thinking and their problem solving strategies. The paper was a work-in-progress paper, but according to researchers, the initial feedback from students was very positive.

Begosso, Begosso & Begosso (2016) are interested in helping university students understand easier programming concepts, with the last years to dedicate to the development of LO to support teaching computer programming concepts to students in early years of Computer Science programs. They have developed and applied LO oriented to supporting the teaching of computer programming concepts. Specifically, in 2015 Begosso, Begosso, Begosso Ribeir & dos Santos developed learning objects to teach Pointers, Data Structures, Binary Trees and Data Classification. Their learning objects have interactive features and work on the basics of algorithms developing, instructions, data and results, and allocation of data in memory. After the conceptual aspects are being presented, a series of questionnaires follow to verify the student's learning level. If the student's understanding was not satisfactory, he or she can return to the previous steps and review the concepts again. The authors conducted two case studies with students without or with low knowledge level of pointers and binary trees and according to the results, the participants were extra motivated to work with the learning objects and they had remarkable levels of success in the assessments.

Matthews, Soon Hin and Ah Choo (2014) investigated the effects of different sized learning objects on programming learning. Driven by the questions if there is any difference between the sizes of LO in improving the programming knowledge or if it be useful to integrate different sizes of LO for programming learning, they developed two different sizes of Learning Objects in order to teach C programming language to students with little or without prior programming knowledge. They divide the learning objects into two categories according to their size: micro learning objects (named *content learning objects and Self-assessment object*) and macro learning objects which are aggregated with several LO. The content LOs (are designed to aid the understanding of abstract programming concepts) and self-assessment LOs (designed to help students to understand the programming syntax and codes) vary from 5-10 minutes while the larger are from 20-30. The content LO has visuals and animation and provides an area where the student can practice and immediate feedback as a way to avoid misconception. The macro LOs consist of a main page and a help page, navigations icons to switch back and forth, content LO and self-assessment LO. According to the findings, the students had a great interest in using the LO as a learning support. About the size, smaller objects where more useful, more preferable with the majority of students and more reusable.

In the two following papers, the authors present the development of LO and describe their features for supporting Computer Science concepts. To begin with, Luna-Ramírez & Jaimez-González (2014*)*, present a set of LOs for supporting structured programming courses. The authors describe learning objects as *a set of digital resources that can be used in different contexts with an educational purpose, which consists of at least three internal elements: content, learning activities and contextualization elements.* The learning objects are created in collaboration with the undergraduate students of the Information Technologies and Systems Faculty and focus on C programming language. Their components are: *description of the problem, solution proposed, flow chart, pseudo code, source code in the C programming language, execution example, executable file, and keywords.* Respectively, Jimoyiannis, Christopoulou, Paliouras, Saridaki, Toukiloglou & Tsakonas (2013), report the development of a variety of LOs aiming at the enrichment of both, the lower secondary education computer science e-textbooks and the Greek National Aggregator of Educational Content. They define a learning object as *a digital element which incorporate features such as hyperlinks, multimedia, interactivity and search ability* and they recognize three basic components: a) learning scenario, b) model and c) instructional overlay. Based on the National Curriculum of ICT, the Computer Literacy and the findings about students' needs, the authors created 157 learning objects for Grades7-9, which are classified in four categories:

- *Visualizations-presentations*: interactive, structured learning objects, which present chunks of multimedia information

- *Micro-activities*: a Java applet which includes a series of default bitmap graphics
- *Micro-lessons*: interactive videos which support a short task or a complete activity, using general purpose software or programming environment
- *Open-ended applications:* a small set of dynamic features which give the users the possibility to make changes and see the result on the screen.

In the following papers the authors present the use of *improved learning objects* for representing abstract ideas of Computer Science. In the first paper (2009) Villalobos, Calderón and Jiménez designed and developed Interactive Learning Objects (ILO) and they investigate how they impact the development of programming skills of the University Los Andes students during their Computer Science course. ILO are described as interactive visualization tools with well-defined learning objectives, used by students inside active learning environments in Computer Science education. According to the findings, general improvement in computer programming course is mentioned; the students' positive attitude about programming courses had increased in more than 21% and their failure in the course declined in 49%. In the second paper (2013) Burbaite, Stuikys, & Damasevicius move from virtual LOs to tangible LOs. During a Computer Science course for high school students, they demonstrate how abstract concepts of Computer Science can be visualized and made more understandable with robotics. The authors introduce the concept of robots as Physical Learning Objects (PLO), defining them as smart learning objects (e.g., a mobile robot) that have sensors and/or actuators to interact with their environment and content (control program) to control their behavior. According to their findings, with PLO they succeeded in student engagement in learning and they developed student abilities to critically analyze and compare different problem solving algorithms (e.g., line-following algorithms in our example).

Concerning the findings, the use of LO contributed to a better understanding of the programming principles by the students; they approached the programming concepts more effectively and they were more interested in CS in general. The majority of the LO created for teaching programming focused extensively on teaching Java programming language and C ++ and they are addressed to secondary and university students. Out of the 12 studies examined, 9 deal with LO for university students and three of them for students of secondary education. There were no studies concerning the use of LO in programming in elementary education. This void in in the reviewed literature regarding the LOs dealing with programming concepts which address primary students led to the preparation of this study.

Regarding the LOs mentioned in the papers, as we can see, the most of the LOs above follow a definition given by their researchers. They share some common characteristics: they include one or more multimedia elements such as sound, video, animation, graphics, text, and some kind of user interaction. Additionally, they attempt to actively involve the user by providing feedback and they are designed to teach having specific learning goal.

Chapter 3

The Proposed Educational Intervention

3.1. Description of the developed LOs

The current paper is based on the definition of Mikropoulos & Bellou (Μικρόπουλος & Μπέλλου, 2016), according to whom learning object is: "*a small, self-contained, reusable and pedagogical complete structure of learning content*".

<u>*How does the proposed definition compare with existing definitions*?</u> As Rehak & Mason state (2003), the definitions about LOs can range from just about anything (e.g., IEEE) to something requiring specific objectives and assessment. The definition of Mikropoulos & Bellou provide is a sense of the internal structure of the learning object; it describes a specific level of organization with specific characteristics. Based on Learnativity Content Model of organization and granularity, they state that a LO consists of content data (e.g. multimedia data) and information objects (concepts) and it is a learning component in a learning environment (Fig. 5).

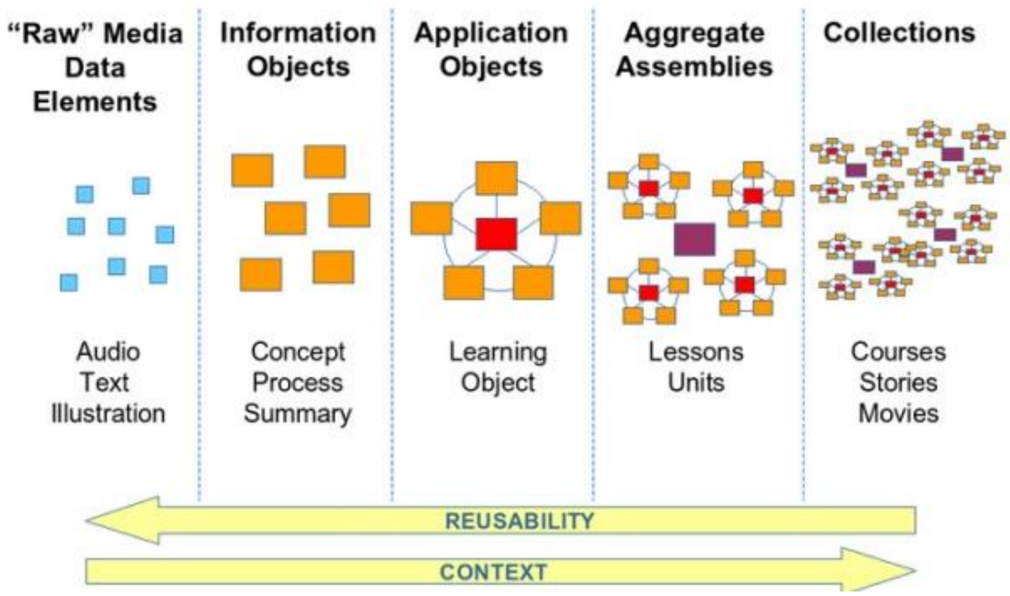


Fig. 5 - Learnativity 2001

<u>*Does the proposed definition support the general characteristics of the LO?*</u> The definition establishes a priori that learning objects are: reusable (*reusability*), small

(*granuality*) and self-contained (*durability*). The LOs which have been created based on this definition are easily accessible via the Web (*accessibility*), function the same way on all common browsers (and they function either online or downloaded for use offline) (*interoperality*), are editable easily able to be easily used even by non-technical users (*manageability*), can be combined with others (*generativity*), can be found easily with the proper keywords (*discoverability*). Last but not least, according to the proposed definition, the LOs are delivered and must be used for learning.

3.2. Learning Theory

The Learning Theory in which the design of Learning Objects was based is *Constructivism*. Constructivism is a widely accepted theory, is based on educational psychology where learning occurs when individuals depending on their experience build on their own personal knowledge; knowledge cannot simply be passed from a teacher to a student, but the students themselves have to build it. Learning is the process of creating knowledge based on existing concepts, ideas and experiences; sometimes by incorporating new information into existing knowledge and other times by placing the conflict and eventually reconstructing a personal individual understanding (Ben-Ari, 2001; Luo, 2005).

Regarding the current Learning Objects, the researcher based on constructivism included real examples which are as close as possible to the learner´s real work situation in order to foster transfer. As far as practices are concerned, problem-based learning was to be emphasized. The feedback given to the users, following the specific learning approach, was meaningful in trying to give hints of what was wrong when errors occur in the program synthesis or to boost their self-esteem in cases of correct program synthesis.

3.3. Learning Objects' programming language

The context of our research is Scratch — a visual programming environment that enables young people to create their own interactive stories, games, and simulations, and then share those creations in an online community with other young programmers from around the world. Scratch was selected, since it is widely widespread in the education community and gathers the characteristics of reusability (LO can be re-used in various situations such as courses, classes, etc.), of interoperability (LO work in different environments), of manageability (easily allows the user -teacher and student- to modify the LO). Additionally, Scratch is one of many programming tools developed to support constructive learning involving the student actively in the educational process. The design and implementation of projects in Scratch promotes students to build their own personal knowledge; students in Scratch are considered as creators and teachers are guides and advisors.

Scratch is a free educational programming language that was developed by the Lifelong Kindergarten Group at the Massachusetts Institute of Technology (MIT). It is designed to be fun and educational. It has the tools for creating interactive stories, games, art, simulations, and more, using block-based programming. Users create programs in Scratch by dragging blocks from a block palette and attach them one after another like a jigsaw puzzle. Structures of multiple blocks are called scripts. This method of programming (building code with blocks) is referred to as “*drag-and-drop programming*”. Scratch is used in schools around the world as a means of introducing basic computer programming to students-especially to younger ones without prior experience on it (Fig. 6). Students are learning with Scratch at all levels (from elementary school to college) and across disciplines (such as math, computer science, language arts, social studies).

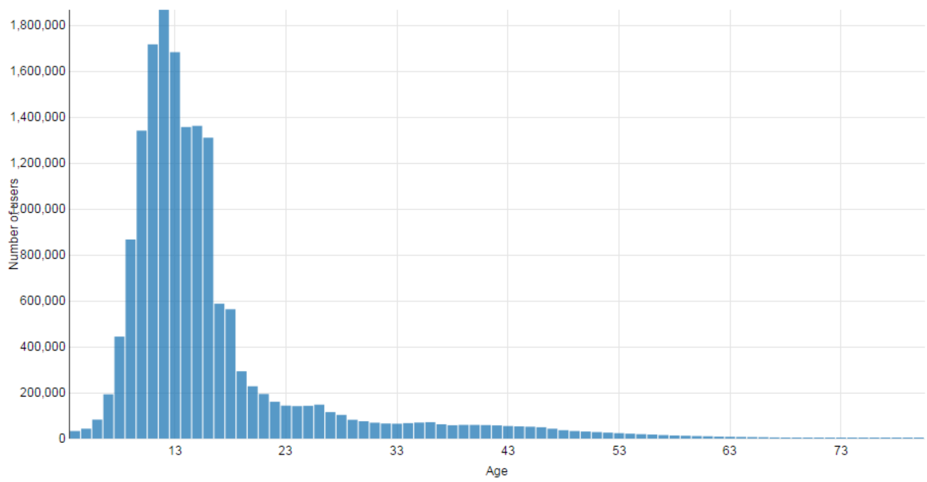


Fig. 6. Age Distribution of New Scratchers

Concerning our proposed intervention, we have identified a set of computational concepts that can be supported by Scratch environment, are common in many programming languages and can be transferred to other programming and non-programming contexts. The created LOs underline the importance of:

- *sequence*: identifying a series of steps for a task
- *loops*: running the same sequence multiple times
- *events:* one thing causing another thing to happen
- *conditionals*: making decisions based on conditions
- *parallelism:* making things happen at the same time

3.4. Description of the current Learning Objects

The user interface of the current Learning Objects attempts to remind the user interface of Scratch (Fig. 7). The researcher programmed a Scratch environment *inside Scratch*, with the figures, the blocks and the main buttons to function in the same way as in "real" Scratch.

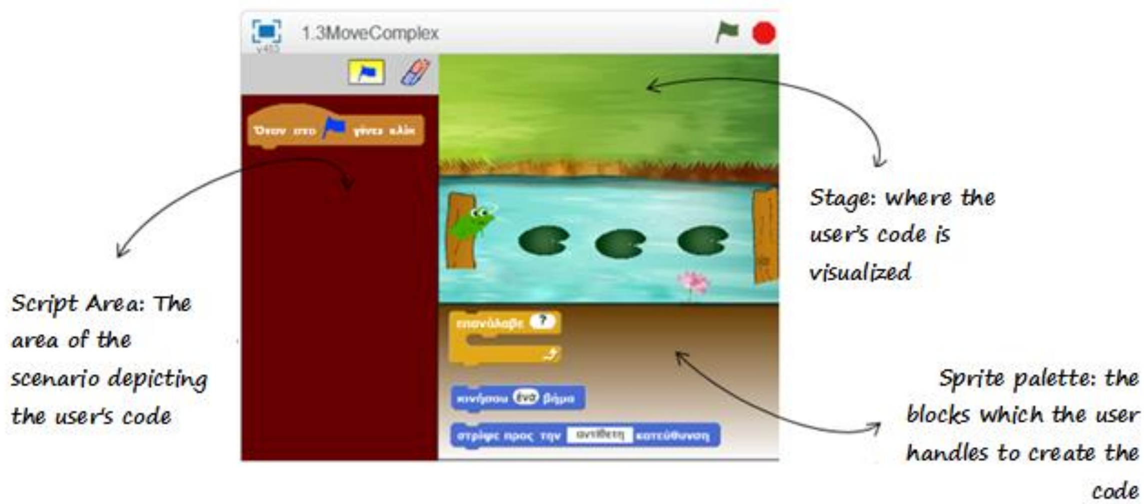


Fig. 7. The interface of the proposed LOs

The users only handle the commands which are at the bottom of the screen and after composing the code, they see the results represented on the screen on the right. The created Learning Objects focus on the structures of *sequence*, of *repetition* (loops), of *selection* (if / else) and of *transmission* (broadcast/ when I receive). Each LO has as main "actor" a frog, which the users have to program correctly in order to perform a particular action of repetition, of selection or/and message transmission and communication between other sprites.

Table 2. Mini Presentation of the proposed LOs

n	Learning Objects	Objective
1	MoveNoRepeat	The aim of the LO is to introduce the idea of the sequence in programming without the use of specific commands.
2	MoveRepeat	The aim of this LO is to underline the importance and the usefulness of the repetition command.
3	MoveComplex	The aim of this LO is to give the students the opportunity to get exercised with the execution of more complex codes applying the repetition command.
4	If_else	The aim of this LO is to help the students to understand that different events are executed in a different way based on the condition of a problem situation.
5	Broadcast_when I receive	The aim of this LO is to make the students feel comfortable with the absent idea of message transmission within Scratch environment.

3.4.1. *MoveNoRepeat*

Detailed Description: The current Learning Object is considered as an initial level of the LOs MoveRepeat and MoveComplex, introducing the importance of the sequence of the steps which are needed to complete a code. The users' goal is to program correctly the frog to reach the wood which is towards it, by jumping on the waterlilies (Fig. 8-left). Specifically, they are asked to click on the block "move one step" as many times as it is needed until they complete the code (Fig. 8-right).

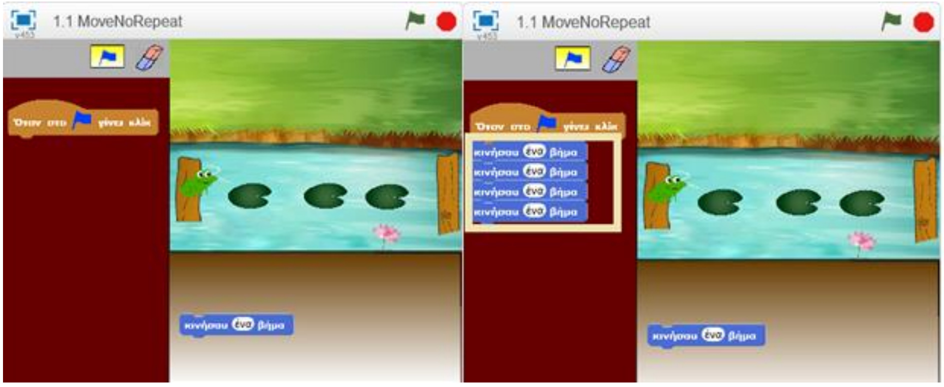


Fig. 8 The interface of the LO "MoveNoRepeat" (1)

When the code is finished, the users have to click the blue flag (at the top of the code), to see the result of the code represented by the frog, which moves as many times as the users have programmed it. While the program is running, a yellow arrow indicates the current point of the code (Fig. 9-top). Eventually, depending on whether the code is correct or not, the frog gives the corresponding feedback (Fig. 9-bottom).

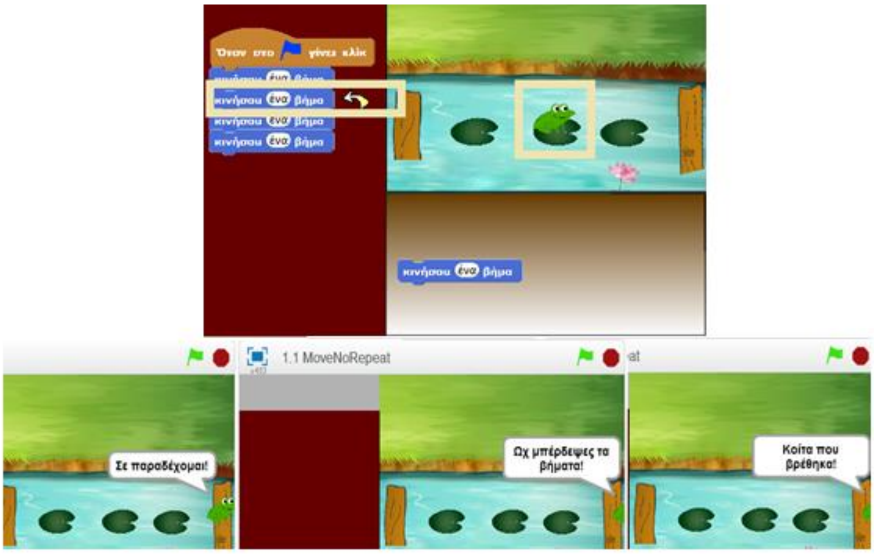


Fig. 9. The interface of the LO "MoveNoRepeat" (2)

Table 3. Metadata of the LO "*MoveNoRepeat*"

General Information	Technical Information	Educational Information
Title: *MoveNoRepeat* Author: Paraskevi Topali Keywords: Scratch, move, sequence Description: *Interactive application which introduces the importance of sequence in Scratch*	Format: *Scratch file (.sb)* Technical Requirements: *run in Scratch 2.0 and online at: https://scratch.mit.edu/*	Intended End User: *learner* Educational Context: *primary* Typical Age Range: *9-12* User Language: *Greek* Interactivity type: *active* Subject Areas: *ICT > programming* Learning Resource Type: *exploration, open activity* Teaching Approach: cognitivist > inquiry learning & problem based learning Educational Objective: to apply and to understand (cognitive), to respond and to participate (affective)

3.4.2. MoveRepeat

Detailed Description: The LO MoveRepeat come as a second level of the first LO, introducing the value of repetition. Its goal is the frog to reach the wood which is towards it, jumping on the waterlilies. The users have to use fruitfully the repeat block. As soon as the users click on the repeat block, automatically they are asked how many times they want to proceed with the iterative energy (Fig. 10. left).

Fig. 10 The interface of the LO “MoveRepeat” (1)

Once the users have completed their code, they see the result represented by the frog, which acts as the users have programmed it. In the case of the command of repetition, a yellow arrow indicates the current point of the code and a red arrow indicates how many times the frog has performed the iterative energy (Fig. 11). As a result, the users visualize how the repetition block works.

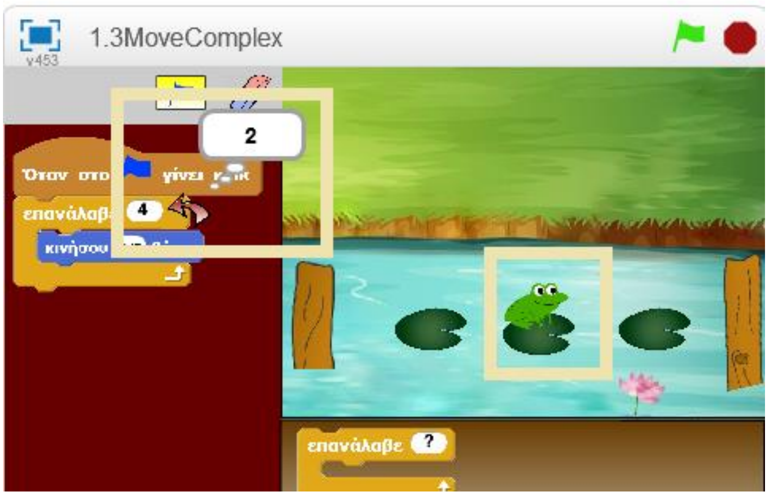


Fig. 11 The interface of the LO “MoveRepeat” (2)

Table 4. Metadata of the LO "*MoveRepeat*"

General Information	Technical Information	Educational Information
Title: *MoveRepeat* Author: Paraskevi Topali Keywords: Scratch, repetition Description: *Interactive application which introduces the value of repetition in Scratch*. The *aim of the LOs is the students to: understand the concept of repetition, explain the usefulness of repetition.*	Format: *Scratch file (.sb)* Technical Requirements: *run in Scratch 2.0 and online at: https://scratch.mit.edu/*	Intended End User: *learner* Educational Context: *primary* Typical Age Range: *9-12* User Language: *Greek* Interactivity type: *active* Subject Areas: *ICT* > *programming* Learning Resource Type: *exploration, open activity* Teaching Approach: cognitivist > inquiry learning & problem based learning Educational Objective: to apply and to understand (cognitive), to respond and to participate (affective)

3.4.3. MoveComplex

Detailed Description: The LO MoveComplex come as a third upgraded level of the previous two LOs. Its goal is the frog to reach the wood which is towards it and then to return back to its initial position, jumping on the water lilies. The users have to use fruitfully the repeat block as many times as it is needed. As soon as the users click on the repeat block, automatically they are asked how many times they want to proceed with the iterative energy (Fig. 12).

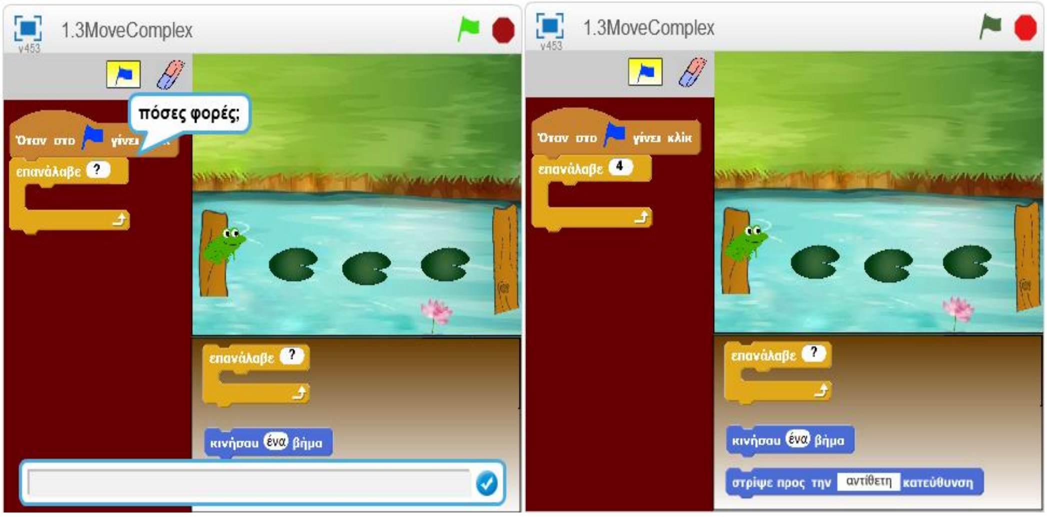


Fig. 12 The interface of the LO “MoveComplex” (1)

Once the users have completed their code, they see the result represented by the frog, which acts as the users have programmed it. In the case of the command of repetition, a yellow arrow indicates the current point of the code and a red arrow indicates how many times the frog has performed the iterative energy (Fig.13). As a result, the users visualize how the repetition block works.

Fig. 13 The interface of the LO “MoveComplex” (2)

Table 5. Metadata of the LO "*MoveComplex*"

General Information	Technical Information	Educational Information
Title: *MoveComplex* Author: Paraskevi Topali Keywords: Scratch, repetition Description: *Interactive application which introduces the value of repetition in Scratch*. The *aim of the LO is the students to: consolidate the functionality of the repeat command in more complex situations, explain the usefulness of repetition*	Format: *Scratch file (.sb)* Technical Requirements: *run in Scratch 2.0 and online at: https://scratch.mit.edu/*	Intended End User: *learner* Educational Context: *primary* Typical Age Range: *9-12* User Language: *Greek* Interactivity type: *active* Subject Areas: *ICT > programming* Learning Resource Type: *exploration, open activity* Teaching Approach: cognitivist > inquiry learning & problem based learning Educational Objective: to apply and to understand (cognitive), to respond and to participate (affective)

3.4.4. If_else

<u>Detailed Description:</u> The users' goal in this learning object is to program the frog to say "*waterlily is disappeared"*, if the lily is missing, otherwise to jump on the lily (Fig. 14 down). The lily is programmed to appear or disappear at random every time that the LO starts, with the frog to perform its code according to which commands were given whether or not the waterlily appears.

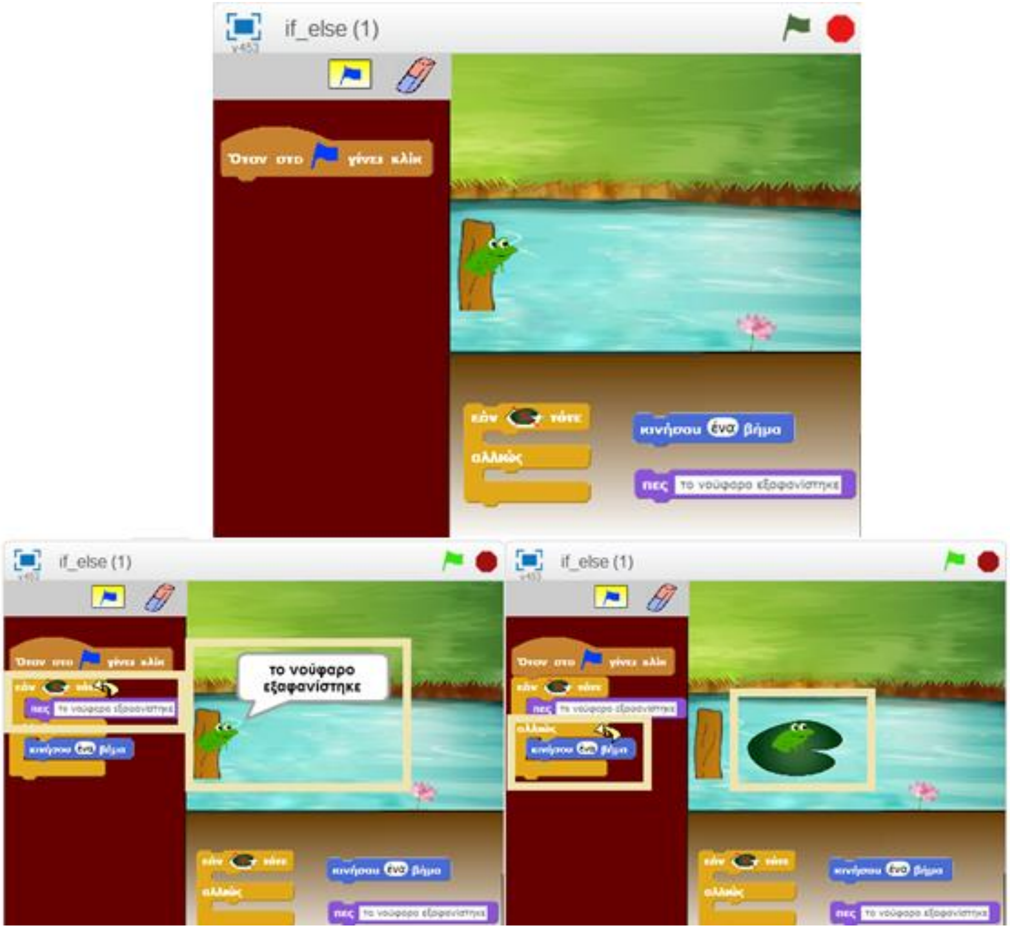


Fig. 14 The interface of the LO "If_else"

In programming the part of "checking conditions" is of great importance. Conditional statements ask questions about the program state to choose from a set of different sequences of commands. This LO aim to show exactly that; depending on the condition of a problem, each time a different piece of code will run.

Table 6. Metadata of the LO "*If_else*"

General Information	Technical Information	Educational Information
Title: *MoveComplex* Author: Paraskevi Topali Keywords: Scratch, if. If/else, selection Description: *Interactive application which introduces the value of selection in Scratch*. The *aim of the LO is the students to: understand the command if / else and how it works, incorporate into their code commands that are executed after specific events*	Format: *Scratch file (.sb)* Technical Requirements: *run in Scratch 2.0 and online at: https://scratch.mit.edu/*	Intended End User: *learner* Educational Context: *primary* Typical Age Range: *9-12* User Language: *Greek* Interactivity type: *active* Subject Areas: *ICT > programming* Learning Resource Type: *exploration, open activity* Teaching Approach: cognitivist > inquiry learning & problem based learning Educational Objective: to apply and to understand (cognitive), to respond and to participate (affective)

3.4.5. Broadcast_when I receive

<u>Detailed Description:</u> The commands "*broadcast*" and "*when I receive*" and its combination is a tricky task for students to understand, especially for those who don't have any prior experience in programming. The users' goal is to program correctly two scenarios, one for each sprite (Fig. 15 -top), resulting in the frog transmitting a message to the fly, which, when it receives it, will disappear (Fig. 15 -bottom).

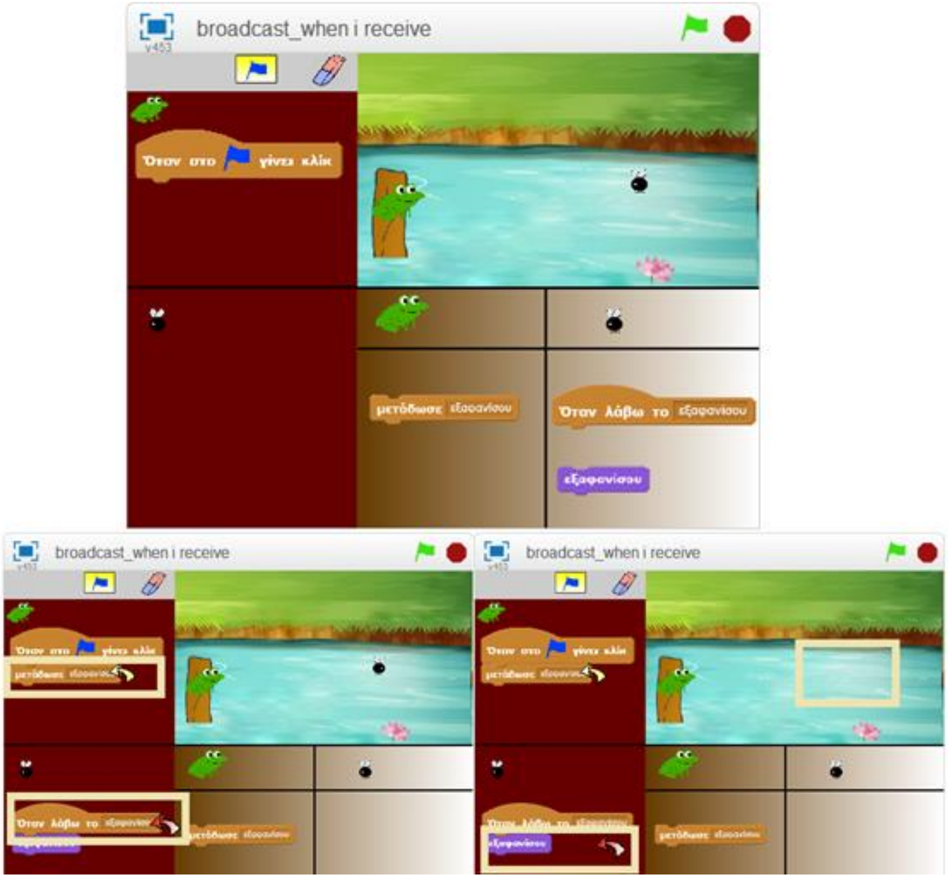


Fig. 15 The interface of the "Broadcast_When I receive"

This LO aim to explain in simple way how the blocks "*broadcast*" and "*when I receive*" work. It makes it clear to the users that these two blocks go always together (broadcast sends a message activating *when I receive* block). The same time visualize their results as an initial attempt to reduce their absent character.

Table 7. Metadata of the LO "*Broadcast_when I receive*"

General Information	**Technical Information**	**Educational Information**
Title: *Broadcast_when I receive* Author: *Paraskevi Topali* Keywords: *Scratch, broadcast, when I receive, transmition* Description: *Interactive application which introduces the sense of transmition in Scratch*. The *aim of the LO is the students to: understand the need for synchronization between objects, consolidate the communication between the scripts.*	Format: *Scratch file (.sb)* Technical Requirements: *run in Scratch 2.0 and online at: https://scratch.mit.edu/*	Intended End User: *learner* Educational Context: *primary* Typical Age Range: *9-12* User Language: *Greek* Interactivity type: *active* Subject Areas: *ICT* > *programming* Learning Resource Type: *exploration, open activity* Teaching Approach: cognitivist > inquiry learning & problem based learning Educational Objective: to apply and to understand (cognitive), to respond and to participate (affective)

Chapter 4

Methodology

4.1. Introduction

This study was designed to obtain educational practitioners' opinions and perspectives regarding the quality of LOs which have been created for primary students without prior experience in programming.

This chapter provides a background to the study, a description of the participants and the applied research design, an explanation of the data collection process and analysis.

4.2. Aim of the study

Due to the existence of bibliographic void in the reviewed literature regarding the LOs for computer science which address primary students led to the preparation of this study. The focus on the design of a number of learning objects for pupils who had no previous programming experience. The author's main priority was to make the developed environment intuitive and easily learned by primary students, within the Scratch environment.

4.3. Research Questions

The research questions of this empirical study:

I. Do the created Learning Objects meet the general characteristics of learning objects?
II. Do the created Learning Objects promote the students' motivation?
III. Can Scratch environment support the development of learning objects?

4.4. Participants

The subjects for this study included educational practitioners coming from different educational stages: primary teachers, professors of secondary and higher education. They were selected systematically to ensure representation of the different educational stages and different points of view. The only limitation in the selection of the participants was that subjects were required to have had experience in teaching Computer Science and in particular applying Scratch in teaching and learning procedure.

4.5. Research Design

The participants were given five Learning Objects to assess and brief instructions explaining how the Learning Objects work. The evaluation of the Learning Objects was done in two phases: first the subjects were tested the created Learning Objects and then they assessed them individually by measuring eight separate qualities of Learning Objects on a scale from one to five. The Learning Object Review Instrument (LORI) developed by Nesbit, Vargo & Belfer (2002), (Nesbit et al., 2002) was used in this research as the assessment tool.

4.5.1. Instrument

The survey questionnaire was structured and organized into 3 main groups:

- questions related to demographic and professional characteristics;
- questions related to participants' perceptions of the created Learning Objects (based on LORI);
- open-end session where the subjects had the opportunity to write comments and provide the researcher with feedback.

The Learning Object Review Instrument was designed in 2002 (Vargo, Nesbit, Belfer & Archambault, 2003), but it has undergone a number of revisions. Originally, the LORI contained ten items for evaluation. In version 4.0 (Vargo et al., 2003), it was revised to include nine (Table 8).

Table 8. LORI Items with Brief Descriptions

1	**Content Quality:** Veracity, accuracy, balanced presentation of ideas, and appropriate level of detail
2	**Learning Goal Alignment**: Alignment among learning goals, activities, assessments, and learner characteristics
3	**Feedback and Adaptation**: Adaptive content or feedback driven by differential learner input or learner modeling
4	**Motivation**: Ability to motivate, and stimulate the interest or curiosity of, an identified population of learners
5	**Presentation Design**: Design of visual and auditory information for enhanced learning and efficient mental processing

6	**Interaction Usability**: Ease of navigation, predictability of the user interface, and the quality of user Interface help features
7	**Accessibility**: Support for learners with disabilities
8	**Reusability**: Ability to port between different courses or learning contexts without modification
9	**Standards Compliance**: Adherence to international standards and specifications

The Learning Objects and the description of the instructions were sent to the participants by email, and the survey questionnaire was online. The subjects responded to the questions and rated items on a Likert scale of 1 to 5.

4.5.2. Configuration of the Learning Objects and the Tool

The development process of the Learning Objects until the final sharing of the sample lasted 4-5 months (Fig 16). In the meantime, there have been changes in the whole design of the Learning Objects. For this research seven Learning Objects were originally created, but after alfa and beta testing only 5 were sent for evaluation. Alfa and beta tests were executed by expert users (specialized in computer science education και computer science teachers) in order to improve the initial LOs.

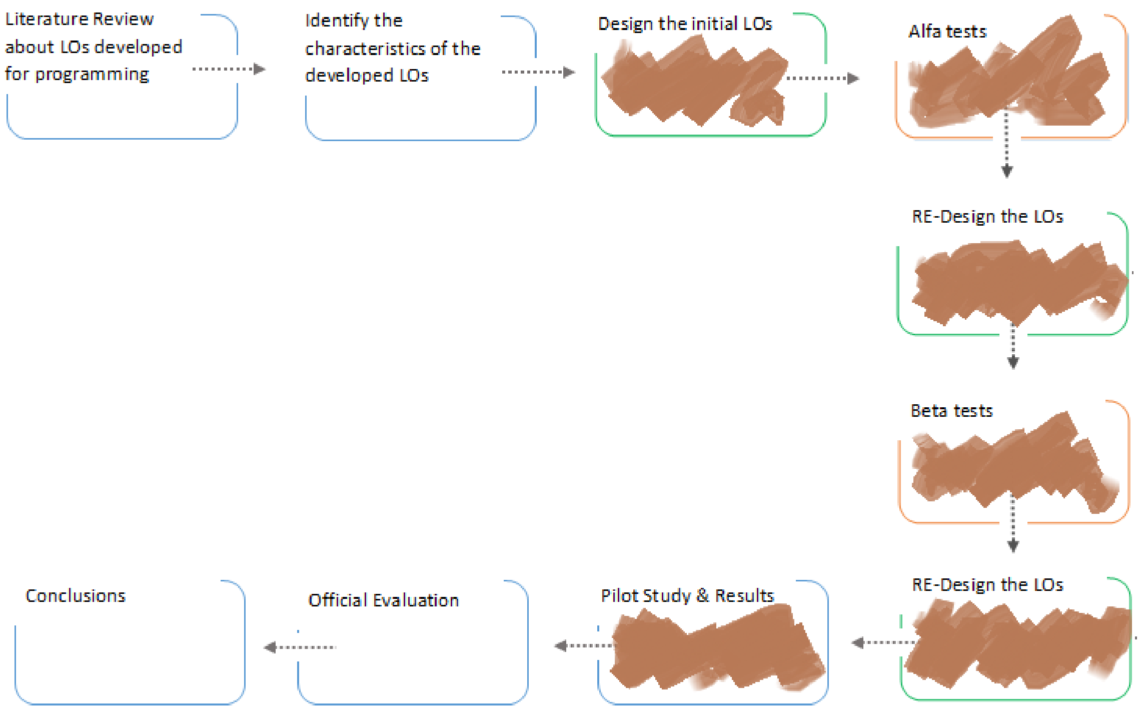


Fig. 16. The envisioned steps of the whole research process

Some adjustments after alfa and beta testing were:

1. the change of the central figure –from a “human” figure to a “frog” figure (Fig. 17)

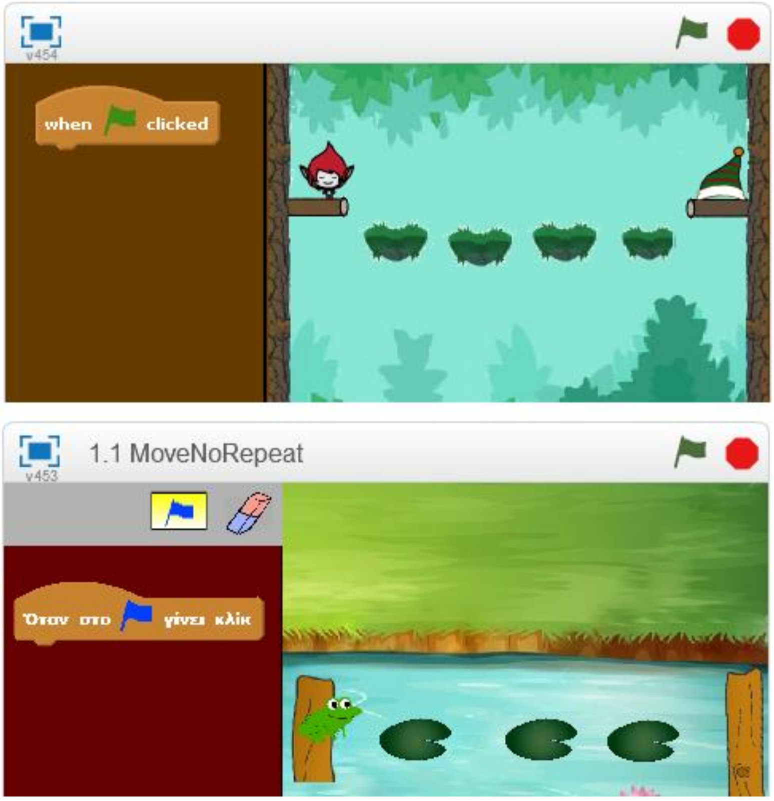


Fig. 17. Some examples regarding the changes of the central figure

2. the change in the numbering of the repeat instruction counter, since initially with every step of the frog the counting was diminished rather than increased, fact that might would possibly be confusing for the users (Fig. 18)

Fig. 18. An example of the change regarding the programming part

3. the change of color -from green to blue- of the button-flag that the user has to click on in order to run his/her code. Initially this button flag was similar to the desktop- flag of Scratch, fact that might/ would be confusing for the users (Fig. 19)

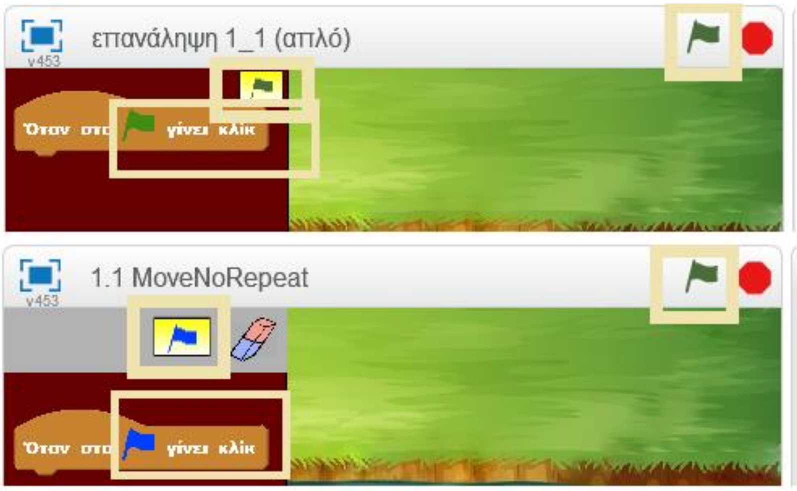


Fig. 19 An example regarding the scripts

Regarding the Tool, some minor changes were made in the wording of the item descriptions of LORI and finally 8 out of 9 item-categories were created, in order to meet the needs of the study. Additionally, each item-category consisted of 2-3 sub-questions.

4.6. Data collection & Data Analysis Strategies

Survey questionnaire was distributed online. Data was collected over a four-week period. Twenty-five completed questionnaires were returned to the researcher.

During the phase of the learning object evaluation process, individual rating was done asynchronously within a period of a few weeks. The participants were provided with the Learning Objects and the online survey questionnaire. They used eight categories for each LO, simply prioritizing their selection on a scale of one to five.

Individual evaluations were analyzed using SPSS for descriptive statistical analysis.

4.7. Pilot Study

Before the evaluation happening, we have chosen to accumulate the preliminary evidence gathered in user trials performed (pilot study) in order to eliminate possible errors and in general to enhance the LOs. A pilot study is a research study conducted before the intended study (Polit, Beck & Hungler, 2001). Pilot studies are usually executed as planned for the intended study, but on a smaller scale. Baker (1994) found that a sample size of 10-20% of a sample size of the actual study is a reasonable number of participants to consider enrolling in a pilot (pp. 182-183). Although a pilot study cannot eliminate all systematic errors or unexpected problems, it reduces the likelihood of making errors.

In our case, we had a smaller sample of 4 people compared to the planned sample size (n=25).

Chapter 5

Results

5.1. Results of the Pilot Study

5.1.1. Data Collection of the Pilot Study

Given the preliminary stage of this first user trials, and for brevity's sake, we will not provide detailed evidence taken during trial. Rather, we will briefly report which were the main findings and conclusions that the evaluators made from them.

The current pilot study included 4 teachers. Two of them were from the field of Computers, Engineers and ICT and the other two from the field of Physics/Math (Table 9**)**.

Table 9. Demographic Data of the Subjects of the Pilot Study

Subject	***Gender***	***Age***	***Study Stage***	***Specialty***	***Years of work***	***Educational Stage***
001	woman	50-59	Master	Physics/Math	>10	High School
002	men	40-49	Bachelor	Computer Science	>10	Primary School
003	woman	50-59	Master	Physics/Math	>10	Secondary School
004	woman	40-49	TEI	Computer Science	>10	Primary School

The pilot participants assessed the Learning Objects and provided answers to the questionnaire (Table 10).

Table 10. Learning Objects assessment

	Questions	***1 poor***	***2 fair***	***3 good***	***4 very good***	***5 excellent***
Content Quality	The content is free of errors and is presented without bias or omissions that could mislead the students.			1	3	

	The content is scientifically correct.		1	2	1
	Graphics highlight the key points and the important ideas with the appropriate level of detail.		1	3	
Learning Goal Alignment	The learning goals are appropriate for the intended students. The learning objects work autonomously and contribute to the achievement of learning goals			2	2
	The learning activities, content and the possible assessments provided by the object align with the declared goals.		1	2	1
	The learning objects work autonomously and contribute to the achievement of the desired learning goals.		1	2	1
Feedback & Adaptation	The learning objects have the ability to tailor instructional messages or activities according to the specific needs or characteristics of each learner.	1		3	
	The learning objects have the ability to simulate or construct phenomena under study in response to differential input from the learner.	1	1	1	1
	The use of the learning objects can be adapted for different students.	2	2	2	
Motivation	The learning objects are highly motivating. Their content is relevant to the personal goals and interests of the intended learners.			3	1
	The object offers choice, true-to-life learning activities, multimedia, interactivity, humor, drama, or game-like challenges. It provides realistic expectations and criteria for success.		2	1	1
	Learners are likely to report an increased interest in the topic after working with the learning objects.		2	2	
	Content contributes to positive learning outcomes. The language		1	2	1

Presentation Design	is clear, comprehensive and error free.				
	Graphics minimize the visual search.		2	2	
	The text is legible.			2	2
	Color, music, and decorative features are aesthetically pleasing and do not interfere with learning goals.		2		2
Interaction Usability	The user interface design implicitly informs learners how to interact with the object, or there are clear instructions guiding use. Navigation through the object is easy, intuitive and free from excessive delay.		1	3	
	The behavior of the user interface is consistent and predictable.	1		2	1
Reusability	The learning object is a stand-alone resource that can be readily transferred to different courses, learning designs and contexts without modification.		3		1
	It operates effectively with a broad range of learners by adapting content or providing adjunctive content such as glossaries and summaries of prerequisite concepts.	1	1	1	1
Accessibility	The learning object is accessible using assistive devices for users with sensory and physical disabilities. It is also accessible via portable devices.	1	2		

Based on the results of the assessment, most subjects seem satisfied with the learning objects. Among the comments of the sample were “*nice work*”, “*fruitful messages when the user makes errors in the code*”, “*some parts of the instructions need more details*”. These findings served to further guide the improvements of some features of the created LOs.

5.2. Results of the Official Evaluation

5.2.1. Survey Questionnaire

5.2.1.1. Demographic data

The final sample of the research included 25 teachers of Computer Science. Out of them, 16 (64%) are women and 9 (36%) men. Their ages range from 30-39 to 50-59 years with the majority being among 40-49 years (36%) (Table 11).

Table 11. Demographic Data of the Subjects (1)

n	***Gender***		***Age***			***Years of Work Experience***		
	man	*woman*	*30-39*	*40-49*	*50-59*	*<6*	*6-10*	*>10*
25	36% (9)	64% (16)	32% (8)	36% (9)	32% (8)	4% (1)	4% (1)	92% (23)

Most of the participants (48%) are of postgraduate level and from the area of Computer Science (60%). Currently, 7 of the participants work in primary schools (28%), while 10 in secondary schools (40%) and 7 in senior high schools (28%) (Table 12).

Table 12. Demographic Data of the Subjects (2)

n	***Educational Level***				***Specialty***			***Educational Stage of work position***			
	Bachelor		Master	PhD	CS	Physics /Math	other	Primary school	Second. school	High school	other
	University	TEI									
25	*36% (9)*	*8% (2)*	*48% (12)*	*8% (2)*	*60% (15)*	*36% (9)*	*4% (1)*	*28% (7)*	*40% (10)*	*28% (7)*	*4% (1)*

5.2.1.2. Instructional Evaluation of the LOs

The questionnaire is divided into 8 categories, each of which contains sub questions. The subjects evaluated each question on a scale of 1(*poor*) to 5 (*excellent*) with the Likert values to be treated as continuous.

5.2.1.2.1. Content Quality

We asked the subjects the following three questions to estimate their general perspective regarding the *content quality* of the created LOs:

- *Q1*: The content is free of errors and is presented without bias or omissions that could mislead the students. Possible differences among cultural and ethnic groups are presented in a balanced and sensitive manner.
- *Q2*: The content is scientifically correct.
- *Q3*: Graphics highlight the key points and the important ideas with the appropriate level of detail.

The results, presented in Table 13, indicate that the subjects were in general satisfied with this section (M= 4,3867, SD=0,66444).

Table 13. Descriptive statistics of content quality

N	Mean	Std. Deviation	*95% Confident Interval*	
			Lower bound	Upper bound
25	**4,3867**	**0,66444**	3,0843976	5

5.2.1.2.2. *Learning Goal Alignment*

We asked the subjects the three following questions to estimate their general perspective regarding the *learning goal alignment* of the created LOs:

- *Q4*: The learning goals are appropriate for the intended students. The learning objects work autonomously and contribute to the achievement of learning goals.
- *Q5*: The learning activities, content and the possible assessments provided by the object align with the declared goals.
- *Q6*: The learning objects work autonomously and contribute to the achievement of the desired learning goals.

The results, presented in Table 14, indicate that the subjects were in general satisfied with this section (M= 4,2667, SD=0,63828).

Table 14. Descriptive statistics of learning goal alignment

N	Mean	Std. Deviation	*95% Confident Interval*	
			Lower bound	Upper bound
25	**4,2667**	**0,63828**	3,0156712	5

5.2.1.2.3. Feedback and Adaptation

We asked the subjects the three following questions to estimate their general perspective regarding the *feedback and adaptation* of the created LOs:

- *Q7*: The learning objects have the ability to tailor instructional messages or activities according to the specific needs or characteristics of each individual learner.
- *Q8*: The learning objects have the ability to simulate or construct phenomena under study in response to differential input from the learner.
- *Q9*: The use of the learning objects can be adapted for different students.

The results, presented in Table 15, reveal that the subjects were positive (M= 3,5867), with some of the educators to be satisfied with this section a lot and some not so much (SD=0,90921).

Table 15. Feedback & Adaptation

N	**Mean**	**Std. Deviation**	***95% Confident Interval***	
			Lower bound	Upper bound
25	**3,5867**	**0,90921**	1,8046484	5

5.2.1.2.4. Motivation

We asked the subjects the three following questions to estimate their general perspective regarding the *motivation* of the created LOs:

- *Q10*: The learning objects are highly motivating. Their content are relevant to the personal goals and interests of the intended learners.
- *Q11:* The object offers choice, true-to-life learning activities, multimedia, interactivity, humor, drama, or game-like challenges. It provides realistic expectations and criteria for success.
- *Q12*: Learners are likely to report an increased interest in the topic after working with the learning objects.

The results, presented in Table 16, reveal that the subjects were more positive in this section than in the previous one (M= 3,8133), with some of the educators to be satisfied with this section a lot and some less (SD=0,88757).

Table 16. Motivation

N	Mean	Std. Deviation	*95% Confident Interval*	
			Lower bound	Upper bound
25	**3,8133**	**0,88757**	2,0736628	5

5.2.1.2.5. Presentation Design

We asked the subjects the four following questions to estimate their general perspective regarding the *presentation design* of the created LOs:

- *Q13*: Content contributes to positive learning outcomes. The language is clear, comprehensive and error free.
- *Q14:* Graphics minimize the visual search.
- *Q15*: The text is legible.
- *Q16*: Color, music, and decorative features are aesthetically pleasing and do not interfere with learning goals.

The results, presented in Table 17, indicate that the subjects were really satisfied with this section (M= 4,29), with some of the educators to be satisfied with this section a lot and some a bit less (SD=0,82184).

Table 17. Presentation Design

N	Mean	Std. Deviation	*95% Confident Interval*	
			Lower bound	Upper bound
25	**4,29**	**0,82184**	2,6791936	5

5.2.1.2.6. Interaction Usability

We asked the subjects the four following questions to estimate their general perspective regarding the *interaction usability* of the created LOs:

- *Q17*: The user interface design implicitly informs learners how to interact with the object, or there are clear instructions guiding use. Navigation through the object is easy, intuitive and free from excessive delay.

- *Q18:* The behavior of the user interface is consistent and predictable.

The results, presented in Table 18, indicate that the subjects were quite positive with this section (M= 4,29), with some of the educators though to be satisfied a lot and some a bit less (SD=0,82184).

Table 18. Interaction Usability

N	Mean	Std. Deviation	*95% Confident Interval*	
			Lower bound	Upper bound
25	**3,96**	**0,96738**	2,0639352	5

5.2.1.2.7. Reusability

We asked the subjects the four following questions to estimate their general perspective regarding the *reusability* of the created LOs:

- *Q19*: The learning object is a stand-alone resource that can be readily transferred to different courses, learning designs and contexts without modification.
- *Q20:* It operates effectively with a broad range of learners by adapting content or providing adjunctive content such as glossaries and summaries of prerequisite concepts.

The results, presented in Table 19, indicate that the subjects were positive with this section (M= 3,64), with some of the educators to be satisfied a lot and some not so much (SD=0,90738).

Table 19. Reusability

N	Mean	Std. Deviation	*95% Confident Interval*	
			Lower bound	Upper bound
25	**3,64**	**0,90738**	1,8615352	5

5.2.1.2.8. Standards Compliance

We asked the subjects the following optional question to estimate their general perspective regarding the *standards compliance* of the created LOs:

- *Q21*: The learning object is accessible using assistive devices for users with sensory and physical disabilities. It is also accessible via portable devices.

The results, presented in Table 20, suggest that the subjects were quite positive with this section (M= 3,9231), with some of the educators to be satisfied a lot and some not so much (SD=0,95407).

Table 20. Standards Compliance

N	Mean	Std. Deviation	*95% Confident Interval*	
			Lower bound	Upper bound
13	3,9231	0,95407	2,0531228	5

In the two sections (Feedback & Adaptation and Reusability) where the means were the lowest (M*f*= 3,5867 & M*r*= 3,64), the researcher examined possible differences in the answers of the subjects because of their age or the educational stage of their work position. Significant differences were not found (Table 21 & Table 22). Concerning reusability, educators of secondary and high school are more positive (M*s*=3,8182 & M*h*=4) than the ones of primary school (M= 3,3571).

Table 21. Feedback & Adaptation lowest means

		N	***Mean***	***SD***
Age	*30-39*	8	3,8333	0,79682
	50-59	8	3,75	1,00396
Educational stage of work position	*Primary school*	7	3,3809	0,67847
	Secondary school	11	3,7576	1,06553
	High school	8	3,5	0,85449

Table 22. Reusability lowest means

		N	***Mean***	***SD***
Age	*30-39*	8	3,9375	0,77632

	50-59	8	3,8125	1,09992
educational stage of work position	*Primary school*	7	3,3571	0,89974
	Secondary school	11	3,8182	0,84477
	High school	8	4	1,08972

5.2.1.3. Feedback of the Subjects

The evaluation of the created LOs is based also on the feedback which the educators provided at the end of the questionnaire answering the optional question "*Please note below your comments and your proposals*". Their responses were divided into two categories: impressions and possible future improvements.

The former expresses their positive impressions while interacting with the LOs. Some of the comments were: "*In general students face problems understanding commands such as the repetition one. These LOs provide a visual perspective of this abstract command and support the educational process*", "*Interesting approach creating LOs with Scratch which is now well known to the educational community", "Pleasant, ideal for younger students", "The LOs motivate the students, boost their interest and are aligned to the learning goals. Very good effort!", "Very effective introduction to the programming structures with a playful way", "It is definitely an excellent and original idea!!".*

The latter category is one of comments which refers to weaknesses and possible improvements. Some of the remarks were: "*It would be preferable when the students make errors at their code, to have the possibility of correcting the wrong part directly and not to have to reboot the whole procedure", "It is very well designed but it responds quite slowly", "I think that in the LO of If_Else_Then, it would be preferable the waterlily to appear and then to disappear. In that way, the execution of the code would be clearer to the students", "Nice work, but it needs some improvements. The commands used by the students shift positions which will confuse the younger students".*

Chapter 6

Conclusions

6.1. Conclusions of the dissertation

The progressive ubiquity of Information and Communication Technologies in our society at all levels is slowly but recklessly transforming our education settings changing the nature of practitioner's labor in the classroom. The emergence of new pedagogical approaches that use these new technologies has further transformed the lecture-oriented mission of the teacher. The evolving character of learning technology has fostered the search for methods and technologies which transform the way in which the learning material is produced, stored, manipulated, and experienced leading in reusable learning content and permitting the learner to take the learning itself into their hands (Downes, 2001; Weller, 2007). As a result, a range of digital resources have been developed termed as *learning objects (LOs)*.

This dissertation set out to ***propose, develop and evaluate LOs for teaching basic programming concepts created for students without any prior experience in it***. To carry this out, we first performed a literature review in the domain of LOs developed for teaching programming. To begin with, we analyzed what the "learning object" term means. Though there is not a clearly stated accepted definition of LOs in literature, a LO in its essence can be seen as a *knowledge* "*package"* (Cohel & Nycz, 2006), attempting to deliver learning experiences and support the virtual education technologically and pedagogically (Arturo et al., 2009; Ritzhaupt, 2010), Ritzhaupt, 2010), focusing the same time on reuse and automation of searching, selection and composition of educational contents and activities (Sicilia & Sánchez-Alonso, 2007). As far as the reviewed literature is concerned, the majority of LOs encountered dealing with programming is limited, the most of which appeal to secondary and higher education students and not to primary education. The existence of this bibliographic void led to the preparation of this study.

The current work is based on Mikropoulos & Bellou, which define (2016) the LOs "*a small, self-contained, reusable and pedagogical complete structure of learning content*" Μικρόπουλος & Μπέλλου, 2016). During the phase of the development, we took into account the conclusions we had from the articles reviewed. In the literature review the LOs mentioned share some common characteristics such as the inclusion of multimedia elements (sound, video, animation, graphics, text, etc.), some kind of user interaction and the direct feedback. The LOs we developed follow the same architecture. The summary of the results is presented here. They have been organized in terms of the following research questions:

I. *Do the created Learning Objects meet the general characteristics of learning objects?*

II. *Are the created Learning Objects an effective teaching tool, promoting the students' motivation?*

III. *Can Scratch environment support the development of learning objects?*

Speaking more precisely:

I. The various definitions accompanying the LOs deal with some features, which have become commonly accepted. According to the literature review, the LOs should come along with high levels of *reusability, granuality, discoverability, assesibility, interoperality, adaptability, durability, generativity, manageability*.

The tool we used in order to evaluate or LOs, assess the most of the features above, the answers of the subjects so, are crucial. The specific LOs have been developed in an environment which supports the features mentioned above, fact which the subjects recognized it. The mean of the majority of the answers is higher than 3,8 out of 5 and only in terms of reusability and adaptability the educators seemed more skeptical (M*r*=3,58 & M*a*=3,64). In general we can conclude that the LOs fulfill the criteria of being considered as "LOs".

II. Motivation has been center of attention among teachers throughout the years because it constitutes the backbone of learning process; keeping students motivated is a key issue if we want them to learn (Jenkins, 2001; Dişlen, 2013). Concerning the programming concepts, find a way to boost students' interest is of great importance since its incomprehensible and abstract character discourages the students (Jenkins, 2002; Boyle, 2003; Pickard et al, 2003; Matthiasdottir 2006; Burbaite et al., 2013).

The developed LOs had showed good evidence of supporting the motivation of students. They have been considered by the subjects highly motivating (M=3,81 out of 5) by "*motivating the students, boosting their interest and being aligned to the learning goals".*

III. Driven by the thought of creating LOs in an environment intuitive and easily learned, we developed five LOs using the Scratch environment. To begin with, Scratch was selected, since it is widely widespread in the education community and gathers the characteristics of *reusability* (LO can be re-used in various situations such courses, classes, etc.), of *interoperability* (LO work in different environments), of *manageability* (easily allows the user -teacher and student- to modify the LO). Additionally, Scratch is one of many programming tools developed to support constructive learning involving the student actively in the educational process.

Despite the limitations accompanying Scratch, the final result of the LOs seemed to satisfy the subjects, fact which can be exacted by the evaluation of the participants and the feedback they provide. To the question about presentation design and content quality, the subjects reacted really positively (M= 4,29, & M= 4,32 out of 5); the graphics highlighted the key points and the important ideas with the appropriate level of detail, the language was clear, comprehensive and error free and color, music, and decorative features are aesthetically pleasing and do not interfere with learning goals. In the feedback section, among the comments were "*Interesting approach creating LOs with Scratch which is now well known to the educational community", "Pleasant, ideal for younger students", "Scratch visualized abstract ideas of programming".* Some of comments like "*the blocks and the sprites are moving if a student drag them, though they should be immovable*" are due to restrictions of Scratch.

Overall, the evaluation of the LOs provided us with certain evidence of the usefulness of the created LOs, as well as illustrating potential problems for its adoption in real practice that should be taken into consideration. Without any doubt, the results in general provided valuable feedback.

6.2. Limitations of the dissertation

First and foremost, the main limitation of this research is the size of the sample (n=25). Though, the general feedback was quite positive, the fact that differences among the ages were not detected is due to the small size and the lack of power to detect statistically significant associations. As a result, we cannot generate our results; they can be as a basis for future investigation and a more systematic and wider empirical research.

Moreover, in terms of statistic, a basic limitation is that we behave to our variables as if they were continuous. The values 1, 2, 3, 4, 5 are conventional and by averaging them (fact that is common with those kind of variables), we assume they are real numbers.

Last but not least, the environment of Scratch itself put some restrictions during the phases of designing and development of the LOs. For instance, Scratch cannot support pictures and images of high quality, so we face problems designing the background of our LOs. Additionally, when a size of a project is big concerning the bytes, the system's operation is slower and some errors may occur while executing a program.

6.3. Future Recommendations

This research has an exploratory character and its findings were really satisfying. Based on the conclusions the following recommendations can be made. To begin with, although the findings of this study are quite positive, longer experiments with larger samples need to be conducted to further investigate the effectiveness of the LOs for teaching programming. Along with that, a research can be carried out having as a sample students without any prior experience in programming in order to test in real contexts parameters such as effectiveness, reusability etc. This will enable us to make a final conclusion about the ability of the LOs to support students in their learning.

Other recommendations regarding to the LOs are related to the areas and trends for future investigation and elaboration. The further development can be connected with the following issues: enhance the reusability of the template and its particular components, sequencing of tasks and activities within tasks, extension of interactivity possibilities and improvements of the template interface.

In addition to the Computer Science courses, this approach could be applied to other courses, too, like in Math or Physics.

References

ACM (2013). Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. In *IEEE Computer Society*. Available at: https://www.acm.org/education/CS2013-final-report.pdf [last access November 2016]

ARIADNE Foundation. Available at: http://www.ariadne-eu.org/content/about [last access June 2016]

ASTD & SmartForce (2002). *A Field Guide to Learning Objects. Learning Circuits. Available at:* http://skat.ihmc.us/rid=1170685164708_899713360_23072/field_guide_to_learning_objects.pdf [last access November 2016]

Adamchik, V., Gunawardena, A. (2003). A learning objects approach to teaching programming. In *Proceedings of the International Conference on Information Technology: Coding and Computing [Computers and Communications], (ITCC 2003), IEEE*, pp. 96–99.

Armoni, M., Meerbaum-Salant, O., Ben-Ari, M. (2015). From Scratch to "real" programming. In *ACM Transactions on Computing Education 14* (4), pp.1-15. DOI: 10.1145/2677087

Baker, T.L. (1994), *Doing Social Research* (2nd Edn.), New York: McGraw-Hill Inc.

Balatsoukas, P., Morris, A., & O'Brien, A. (2008). Learning Objects Update: Review and Critical Approach to Content Aggregation. In *Educational Technology & Society, 11* (2), pp. 119-130. Available at: http://ifets.info/journals/11_2/11.pdf [last access November 2016]

Balci, J. B., Inceoglu, M. (2006). Reusable Learning Objects for Computer Science Students In *Computational Science and Its Applications - ICCSA 2006 Proceedings*, pp. 373-382.

Bannan-Ritland, B., Dabbagh, N., Murphy, K. (2002*). Learning object systems as constructivist learning environments: Related assumptions, theories and applications*. In D. A. Wiley (Ed.), *The instructional use of learning objects*. Bloomington, IN.

Barajas, A.., Muñoz, J., Álvarez F., J., García M., E. (2009). Developing Large Scale Learning Objects for Software Engineering Process Model. In the *Mexican International Conference on Computer Science*, pp. 203-208.

Barr, D., Harrison, J., Conery, L. (2011). Computational Thinking: A Digital Age Skill for Everyone. In *Learning and Leading with Technology*, *38*(6), pp. 20–23.

Begosso, L. C., Begosso, L. R., Begosso, R. H., Ribeiro, A., Santos, R. M. (2015). The use of Learning Objects for teaching Computer Programming. In *IEEE Frontiers in Education Conference,* USA, pp. 786- 791.

Begosso, L. C., Begosso, L. R., Begosso, R. H. (2016). An approach for the use of Learning Objects in teaching Computer Programming concepts. In *IEEE Frontiers in Education Conference, USA*, pp.1-8

Belshaw, D. (2011). *What is digital literacy? A Pragmatic investigation*. Doctoral thesis submitted to the Department of Education at Durham University, United Kingdom. Available at: https://dmlcentral.net/wp-content/uploads/files/doug-belshaw-edd-thesis-final.pdf

Ben-Ari, M. (2001). Constructivism in Computer Science Education. In *Journal of Computers in Mathematics and Science Teaching 20*(1), pp. 45-73

Bratina, T. A., Hayes, D. & Blumsack, S. L. (2002). Preparing Teachers To Use Learning Objects. *Extending the Pedagogy of Threaded-Topic Discussions, 2002*(1).

Berry, M. (2013). *Computing in the national curriculum: A guide for primary teachers*. NAACE. Availale at:
http://www.computingatschool.org.uk/data/uploads/CASPrimaryComputing.pdf [last access April 2017]

Boyle, T. (2003). Design principles for authoring dynamic, reusable learning objects. In *Australian Journal of Educational Technology*, *19*(1), pp. 46-58.

Boyle, T., Bradley, C., Chalk, P. (2004). Improving the teaching of programming using a VLE enhanced with learning objects. In *the 2nd International Conference on Information Technology: Research and Education* (ITRE 2004), IEEE, pp. 74–78.

Bruce, K. Freund, S.N. (2008). Programming languages as part of core computer science. In *SIGPLAN Workshop on Programming Language Curriculum 43*(11), pp. 50-54.

Burbaite, R., Damasevicius, R., Stuikys, V. (2013). Using robots as learning objects for teaching computer science. In *X world conference on computers in education (WCCE'13)*, Torun, Poland, pp. 103-111.

Burbaite, R., Stuikys, V., Damasevicius R. (2013). Educational Robots as Collaborative Learning Objects for Teaching Computer Science. In *System Science and Engineering (ICSSE) 2013 International Conference on*, pp. 211–216.

Calao, L. A., Moreno-Leon, J., Correa, H. E., & Robles, G. (2015). Developing Mathematical Thinking with Scratch. In *Design for Teaching and Learning in a Networked World*, pp. 17-27. Springer International Publishing. Available at: http://jemole.me/replication/2015ectel/CodeMath_Draft.pdf

Cavus, N., Ibrahim, D. (2004). Using Learning Objects to Teach Programming Languages. In *the Creating the Future 3rd FAE International Symposium, European University of Lefke.* Available at: http://files.eric.ed.gov/fulltext/ED503158.pdf

Chiappe, A., Segovia, Y., Rincon, Y. (2007). Toward an instructional design model based on learning objects. In *Educational Technology Research and Development, 55*(6). Boston: Springer, pp. 671-681. DOI: 10.1007/s11423-007-9059-0

Chitwood, K., Bunnow, D. (2002). Learning Objects: Recourses for Learning. In *the 18th Annual Conference on Distance Teaching and Learning*, pp.1-4

Cochrane, T. (2005). Interactive QuickTime: Developing and Evaluating Multimedia Learning Objects to Enhance Both Face-To-Face and Distance E-Learning Environments. In *Interdisciplinary Journal of Knowledge and Learning Objects 1,* pp. 33-54

Cohen, E., Nycz, M. (2006). Learning objects and E-learning: An informing science perspective. In the *Interdisciplinary Journal of Knowledge and Learning Objects, 2,* pp. 23-24.

Computer Science Teachers Association (CSTA). (2009). *A Model Curriculum for K–12 Computer Science Level I Objectives and Outlines*. New York, NY

Computer Science Teachers Association (CSTA) and Association for Computing Machinery (ACM). (2011). *K–12 computer science standards*. New York, NY

Computing at School Working Group (2012). *Computer Science as a school subject: Seizing the opportunity*. Available at: http://www.computingatschool.org.uk/data/uploads/Case%20for%20Computing.pdf

Currier, S., Barton, J., O'Beirne, R., Ryan, B. (2004). Quality assurance for digital learning object repositories: issues for the metadata creation process. In *ALT-J, Research in Learning Technology, 12*(1), pp. 5-20

Dagienė, V. (2011) *Informatics Education for New Millennium Learners*. In Kalaš and R.T. Mittermeir (Eds.) *ISSEP 2011, LNCS 7013,* pp. 9–20, Springer-Verlag Berlin Heidelberg.

de Salas, K., Ellis L. (2006). The Development and Implementation of Learning Objects in a Higher Education Setting. In *Interdisciplinary Journal of Knowledge and Learning Objects, 2,* pp. 1-22. Available at http://ijklo.org/Volume2/v2p001-022deSalas.pdf

Dişlen, G. (2013). The Reasons of Lack of Motivation From The Students' And Teachers' Voices. In *Akademik Sosyal Araştırmalar Dergisi the Journal of Academic Social Science*, pp. *35-45*

Downes, S. (2000*). Learning objects*. Available at: http://www.atl.ualberta.ca/downes/naweb/Learning_Objects.htm

Downes, S. (2001) Learning objects: resources for distance education worldwide. In *International review of research in open and distance learning.*

Downes, S. (2001). Learning objects: Resources for distance education worldwide. In the *International Review of Research in Open and Distance Learning, 2*(1).

Downes, S. (2005). E-Learning 2.0. In *ACM eLearn Magazine*. Available *at: http://elearnmag.acm.org/featured.cfm?aid=1104968*

Duncan, C. (2003). Granularization. In A. Littlejohn, (Ed.), *Reusing Online Resources: A Sustainable Approach to E-learning*, pp. 12-19. Kogan Page, London.

Escobar A. E, Reyes P., Van Hilst M. (2014). Metrics for effectiveness of e-learning objects in software engineering education. In *IEEE SOUTHEASTCON*, pp. 1–5

European Parliament and the Council. (2006). *Recommendation of the European Parliament and of the Council of 18 December 2006 on key competences for lifelong learning*. Official Journal of the European Union, L394/310.

European Schoolnet. 2014. *Computing our future. Computer programming and coding – priorities, school curricula and initiatives across Europe*. Belgium: Brussels [ebook] Available at: http://www.eun.org/publications/detail?publicationID=48 1

European Schoolnet (2015). *Computing our future. Computer programming and coding. Priorities, school curricula and initiatives across Europe*. Belgium: Brussels.

Farha, N. W. (2009). An exploratory study into the efficacy of learning objects. In *The Journal of Educators Online 6*(2). Available at: http://www.thejeo.com/Archives/ Volume6Number2/FarhaPaper.pdf

Ferrari, A. (2013). *DIGCOMP: A Framework for Developing and Understanding Digital Competence in Europe*. Luxembourg: Publications Office of the European Union, JRC-IPTS. Available at: http://publications.jrc.ec.europa.eu/repository/bitstream/JRC83167/lb-na-26035-enn.pdf [last access June 2017]

Fetaji, M., Loskovska, S., Fetaji, B., and Ebibi, M. (2007). Combining virtual learning environment and integrated development environment to enhance eLearning. In *Proceeding of the International Conference on Information Technology Interfaces*, pp. 319-324Croatia

Friesen, N. (2001). What are educational objects? In *Interactive Learning Environments*, *9*(3), pp. 219-230

Friesen, N. (2003). *Three objections to learning objects and E-learning Standards*. In McGreal, R. (Ed.). 2004. *Online Education Using Learning Objects*, pp. 59-70. London: Routledge

Gibbons, A. S., Nelson, J. & Richards, R. (2002). The nature and origin of instructional objects. In D. A. Wiley (Ed.), *The instructional use of learning objects*, pp. 25–58. Bloomington, IN: Agency for Instructional Technology and Association for Educational Communications and Technology.

Goldberg, D., Grunwald, D., Lewis, C. (2013). Addressing 21st Century Skills by Embedding Computer Science in K-12 Classes. In the *SIGCSE'13*, Colorado, USA: ACM.

Goodyear, P., Retalis, S. (2010). Learning, technology and design. In P. Goodyear, S. Retalis (Eds.), *Technology-Enhanced Learning: Design Patterns and Pattern Languages*, pp. 1-27. Rotterdam: Sense Publishers.

Guerra, V., Kuhnt, B., Blöchliger, I. (2012). *Informatics at school –Worldwide An international exploratory study about informatics as a subject at different school levels.* Hasler Foundation, University of Zurich

Gürer, M. D. (2013). *Utilization Of Learning Objects In Social Studies Lesson: Achievement, Attitude And Engagement.* Doctoral Thesis submitted to the Department of Computer Education and Instructional Technology, Middle East Technical University. Turkey

Hamel, C. J., Ryan-Jones, D. (2002). Designing instruction with learning objects. In *International Journal of Educational Technology, 3*(1).

Hodgins, H. W. & Conner, M. (2000). Everything You Ever Wanted to Know About Learning Standards but Were Afraid to Ask. In *Learning in the New Economy e-Magazine (LiNE Zine).* Available at: *http://www.linezine.com/2.1/features/wheyewtkls.htm*

Hodgins, H.W. (2002). The future of learning objects. In D. A. Wiley (Ed.), *The instructional use of learning objects*. Bloomington, IN: Association for Educational Communications and Technology. Available at http://reusability.org/read/ [last access September 2016]

Hodgins, H. W. (2004). The future is learning objects. In *Proceedings of 2002 ECI Conference on e-Technologies n Engineering Education: Learning Outcomes Providing Future Possibilities.*

IEEE Learning Technology Standards Committee (LTSC). (2002). *IEEE LOM 1484.12.1-2002 Draft Standard for Learning Object Metadata*. NJ: Institute of Electrical and Electronics Engineers. Available at: https://ieee-sa.imeetcentral.com/ltsc/ [last access June 2017]

IEEE Learning Technology Standards Committee (LTSC). (2005). *The Learning Object Metadata Standard*. Piscataway, NJ: Institute of Electrical and Electronics Engineers http://ieeeltsc.org/wg12LOM/lomDescription

International Society for Technology in Education (ISTE). (2007). *National Educational Technology Standards for Students*, Second Edition.

Jenkins, T. (2001). The motivation of students of programming. In *Proceedings of ITiCSE 2001: The 6th annual conference on innovation and technology in computer science education*, pp. 53–56.

Jenkins, T. (2002.) On the difficulty of learning to program. In *3rd annual Conference of LTSN Centre for Information and Computer Sciences*.

Jimoyiannis, A., Christopoulou, E., Paliouras, A., Petsos, A., Saridaki, A., Toukiloglou, P., Tsakonas, P. (2013). Design and development of learning objects for lower secondary education in Greece: The case of computer science e-books. *In Proceedings of EDULEARN13 Conference*, pp. 41-49

Kafai, Y. B., Burke, Q. (2013). Computer programming goes back to school. In *Phi Delta Kappan*, *95*(1), 61.

Kay, H. (2012). Examining factors that influence the effectiveness of learning objects in mathematics classrooms. In *Canadian Journal of Science, Mathematics and Technology Education, 12*(4), pp. 350-366.

Khenner, E., Semakin, I. (2014). School subject informatics (computer science) in Russia: Educational relevant areas. In the *ACM Transactions on Computing Education 14* (2). DOI: http://dx.doi.org/10.1145/2602489

Kellner, D. (2000) New Technologies/New Literacies: Reconstructing education for the new millennium. In *Teaching Education, 11* (3), pp. 245-265. DOI: 10.1080/713698975.

Koper, R. (2003). Combining reusable learning resources and services with pedagogical purposeful units of learning. In *Reusing Online Resources*, ed A Littlejohn, pp. 46 - 59, Kogan Page, London.

Lehman, R. (2007). Learning object repositories. In *New Directions for Adult and Continuing Education*, *113*, pp. 57–66. DOI: 10.1002/ace.247

Lewis, C. 2011. Is pair programming more effective than other forms of collaboration for young students? In *Computer Science Education 21*(2), pp. 105-134

LOM (2000). *LOM working draft v4.1*. Available: http://ltsc.ieee.org/doc/wg12/LOMv4.1.htm [last access June 2016]

Luna-Ramírez, W. A., Jaimez-González, C. R. (2014). Supporting Structured Programming Courses through a Set of Learning Objects. In the *International Conference on Information Society (i-Society 2014),* pp. 122-126

Luo, D. (2005). Using constructivism as a teaching model for computer science. In *The China Papers,* pp. 36-40

Matthews, R., Hin, H.S. & Choo, K.A. (2014). Learning Object To Enhance Introductory Programming Understanding: Does The Size Really Matter? *In the Turkish Online Journal of Educational Technology 13* (1), pp.174-183 Available at: http://files.eric.ed.gov/fulltext/EJ1018178.pdf.

Matthiasdottir, A. (2006). Usefulness of learning objects in computer science learning. In *Proceedings of Codewitz Open Conference Methods, Materials and Tools for Programming Education*, Tampere, Finland.

McGreal, R., & Roberts, T. (2001). A primer on metadata for learning objects: Fostering an interoperable environment. *E-learning, 2*(10), pp. 26-29.

McGreal, R. (2004). Learning objects: A practical definition. In *International Journal of Instructional Technology and Distance Learning (IJITDL), 9*(1). Available at: http://www.itdl.org/journal/sep_04/article02.htm [last access April 2017]

McGreal, R. (2007). *A typology of learning object repositories*. Auspace. Available at: http://hdl.handle.net/2149/1078 [last access April 2017]

Meerbaum-Salant, O., Armoni, M., Ben-Ari, M. (2011). Habits of programming in Scratch. In *Proceedings of the Sixteenth SIGCSE Conference on Innovation and Technology in Computer Science Education, Darmstadt, Germany*, pp. 168–172

Megalou, E., Kaklamanis C. (2014). Photodentro LOR, the Greek National Learning Object Repository. In *Proceedings of INTED2014*. Publisher: IATED, pp. 309-319. Available at: http://dschool.edu.gr/p61cti/promotion/publications/

MERLOT: Multimedia Educational Resources for Learning and Online Teaching. Available: http://merlot.org/ [last access October 2016]

Merrill, M.D. (2000). Knowledge objects and mentalmodels. In D.A. Wiley (Ed.),*The instructional use of learning objects*. Bloomington IN: AECT.

Mikropoulos, A. T., Bellou, I. (2013). Educational Robotics as Mindtools. In *Themes in Science & Technology Education, 6*(1), pp. 5-14

Morales, R., Leeder, D. and Boyle, T. (2005). A case in the design of generative learning objects (GLOs): applied statistical methods. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2005,* pp. 2091–2097. Chesapeake, VA: AACE

Morgan, P. J. (2011). *Instructional Design Theory for Learning Objects: ADDIE. [Online]*. Available at: https://pamelajmorgan.org/2011/07/20/instructional-design-theory-for-los-addie/_[last access May 2017]

Moyer, L. (2016). *Engaging Students in 21st Century Skills through Non-Formal Learning*. Doctoral thesis submitted to the faculty of Virginia Polytechnic Institute and State University, USA.

Mow, J. (2002). *Learning Objects and Instructional Design, The Herridge Group*. Available at: http://www.herridgegroup.com/pdfs/Learning Objects & Instructional Design.pdf

Muzio, J. A., Heins, T. & Mundell, R. (2002). Experiences with reusable E-learning objects; from theory to practice. In *The Internet and Higher Education*, *5*, pp. 21-34.

Narasimhamurthy, U., Al Shawkani, K. (2009). Teaching of programming languages: an introduction to dynamic learning objects. In *the International Workshop on Technology for Education, 2009, (T4E'09,*) IEEE, pp. 114–115.

Nesbit, J.C., Belfer, K., & Vargo, J. (2002). A convergent participation model for evaluation of learning objects. In *Canadian Journal of Learning and Technology*, *28* (2), pp. 105-120.

NCREL & Metiri Group. (2003). *enGauge 21st century skills: Literacy in the digital age*. Naperville, IL: NCREL & Metiri Group. Available at: https://www.gpo.gov/fdsys/pkg/ERIC-ED463753/pdf/ERIC-ED463753.pdf

NISO (2004), *Understanding Metadata*, National Information Standards Organisation, NISO Press, Ann Arbor, MI.

OECD (2009). New Millennium Learners Initial findings on the effects of digital technologies on school-age learners. In the *International Conference "Learning in the 21st Century: Research, Innovation and Policy*". Paris: OECD publications

Pacific Policy Research Center (2010). 21st century skills for students and teachers. Honolulu: Kamehameha Schools, Research & Evaluation.

Paige, J. 2009. The 21st century skills movement. In the *Educational Leadership, 9* (67).

Papert, S. (1980). Mindstorms: Children, Computers and Powerful Ideas. New York: Basic Books.

Papert, S., M. Resnick. (1995). Technological Fluency and the Representation of Knowledge. Proposal to the National Science Foundation. MIT Media Laboratory

Partnership for 21st Century Skills. (2006). *Framework for 21st century learning*. http://www.p21.org/documents/ProfDev.pdf [last access June 2017]

Parrish, P. (2004). The trouble with learning objects. In *Educational Technology Research and Development*, *52* (1), 49-67

Pickard, P., Chalk, P. and Jones, R. (2003). Creating and employing on-line dynamic learning objects for an introductory programming module. In *25th Conference on Information Technology Interfaces*, pp. 16-19

Pirolli, P., Recker, M. (1994). Learning strategies and transfer in the domain of programming. In *Cognition and Instruction, 12*(3), pp. 235–275

Polit, D. F., Beck, C. T. and Hungler, B. P. (2001). *Essentials of Nursing Research: Methods, Appraisal and Utilization* (5th edition), Philadelphia: Lippincott Williams & Wilkins.

Polsani, P. R. (2003). Use and abuse of reusable learning objects. In *Journal of Digital Information, 3*(4).

Prensky, M. (2001). Digital Natives, Digital Immigrants Part 1. In *the Horizon, 9* (5), pp. 1 – 6 DOI: 10.1108/10748120110424816

Price, T. W. (2015). Integrating Intelligent Feedback into Block Programming Environments. In *Proceedings of the eleventh annual International Cocerence on International Computing Education Research (ICER' 15)*, pp. 275-276. Omaha, Nebraska, USA. DOI: 10.1145/2787622.2787748

Rahmat M., Shahrani S., Latih R., Yatim N. F. M., Zainal N. F. A., Rahman R. A. (2012). Major Problems in Basic Programming that Influence Student Performance. In *Procedia– Social and Behavioral Sciences, 59*, pp. 287–296. DOI: 10.1016/j.sbspro.2012.09.277

Rehak, D., Mason, R. (2003). Keeping the learning in learning objects. In A. Littlejohn (Ed.), *Reusing online resources: A sustainable approach to eLearning*. London7 Kogan.

Resnick, M. (2002). *Rethinking Learning in the Digital Age*. In G. Kirkman (Ed.), *The global information technology report: Readiness for the networked word*. Oxford, UK: Oxford University Press.

Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E., Brennan, K., et al. (2009). Scratch: Programming for all. In *Communications of the ACM., 52*(11), pp. 60-67

Ritzhaupt, A. D. (2010). Learning object systems and strategy: A description and discussion. In *Interdisciplinary Journal of E-Learning and Learning Objects, 6*, pp. 217 –238.

Roy, D., Sarkar, S., Ghose, S. (2010). A Comparative Study of Learning Object Metadata, Learning Material Repositories, Metadata Annotation Tool. In M. Joshi, H. Boley, R Akerkar *(Eds.) TMRF e-Book Advances in Semantic Computing, 2*, pp. 103 – 126

Shank, D. J. (2005). The emergence of learning objects: The reference librarian's role. In *Research Strategies*, *19*(2003), pp. 193–203.

Sicilia M. Á, García, E., Sánchez, S., Rodríguez, E. (2004). Describing learning object types in ontological structures: Towards specialized pedagogical selection. In Proceedings of *World Conference on Educational Multimedia, Hypermedia and Telecommunications ED-MEDIA'04*, pp. 2093–2097

Sicilia, M. A., Sánchez-Alonso, S. (2006). Learning Objects and Learning Designs: Reusable Learning Resources for Virtual Learning Environments. In S. Dasgupta (Ed.) *Encyclopedia of Virtual Communities and Technologies*, pp. 405-408.

Solomou, G., Pierrakeas, C., & Kameas, A. (2015). Characterization of Educational Resources in e-Learning Systems Using an Educational Metadata Profile. In *Educational Technology & Society, 18* (4), pp. 246–260.

Sosteric, M., Hesemeier, S. (2002). When is a Learning Object not an Object: A first step towards a theory of learning objects? In *International Review of Research in Open and Distance Learning 3*(2), pp. 1-16

Sterling, C., Kittross, J. (2015). The Golden Age of Programming. In D. Crowley, P. Heyer, (Eds.): *Communication in History: Technology, Culture, Society*. New York: Longman Publishers USA.

Štuikys, V. (2015). *Smart Learning Objects for Smart Education in Computer Science: Theory, Methodology and Robot-Based Implementation*; Springer: Berlin, Germany.

The Partnership for 21st Century Learning (P21). (2015). *P21 Framework Definitions document*. Available at: http://www.p21.org/storage/documents/docs/P21_Framework_Definitions_New_Logo_2015.pdf

Trilling, B., Fadel. C. (2009). *21st century learning skills*. San Francisco, CA: John Wiley & Sons.

UNESCO (2000). *Informatics for Primary Education. Recommendations*. IITE, Moscow.

UNESCO (2012). *ICT in Primary Education Analytical survey*. Moscow: The Russian Federation.

Vargo, J., Nesbit, J., Belfer, K. Archambault, A. (2003). Learning object evaluation: Computer mediated collaboration and inter-rater reliability. In *International Journal of Computers and Applications 25*(3).

Villalobos, J., Calderón, N., Jiménez, C. (2009). Developing Programming Skills by Using Interactive Learning Objects. In *Proceedings of the 14th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE)*, France.

Weintrop, D., Wilensky, U. (2015). To Block or not to Block, That is the Question: Students' Perceptions of Blocks-based Programming. In *Proceedings of the 14th International Conference on Interaction Design and Children,* pp. 199-208. New York, NY, USA: ACM. DOI: 10.1145/2771839.2771860

Weller, M. (2007). Learning Objects, Learning Design, and Adoption through Succession. In the *Journal of Computing in Higher Education*, *19*(1), pp. 26-47.

Wiley, D. A. (2002). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In D.A Wiley (Ed.), *The Instructional Use of Learning Objects: Available at: https://eclass.teicrete.gr/modules/document/file.php/TP271/Additional%20material/LO%26instructunalDesign%28Wiley%29.pdf*

Wing, J. M. (2006). Computational thinking. In *Communications of the ACM, 49*(3), pp. 33–35

Wisconsin Online Resource Center. Available: http://wisconline.com [last access April 2017]

Wu, B., Qian, K., Bhattacharya, P., Guo, M., Hu, W. (2011). Live programming learning objects on cloud. In *Proceedings of the 11th IEEE International Conference on Advanced Learning Technologies, (ICALT 2011)*, pp. 362-363. DOI: 10.1109/ICALT.2011.113

ΔΕΠΠΣ (2001). Διαθεματικό Ενιαίο Πλαίσιο Προγραμμάτων Σπουδών και Αναλυτικά Προγράμματα Σπουδών Δημοτικού- Πληροφορικής, ΦΕΚ τ. Β΄ 1366, 1373, 1374, 1375, 1376/18-10-2001.

ΔΕΠΠΣ (2011). Διαθεματικό Ενιαίο Πλαίσιο Προγραμμάτων Σπουδών και Αναλυτικά Προγράμματα Σπουδών Δημοτικού. Διαθέσιμο: http://ebooks.edu.gr/info/cps/18deppsaps_Pliroforikis.pdf [τελευταία πρόσβαση: Ιούνιος 2017]

Πρόγραμμα Σπουδών μαθημάτων Δημοτικού Σχολείου με ΕΑΕΠ (2010). *Διδασκαλία−πρόγραμμα σπουδών των νέων διδακτικών αντικειμένων που θα εισαχθούν στα ολοήμερα δημοτικά σχολεία που θα λειτουργήσουν με Ενιαίο Αναμορφωμένο Εκπαιδευτικό Πρόγραμμα (ΕΑΕΠ)− επανεξέταση & επικαιροποίηση των Αναλυτικών Προγραμμάτων και οδηγιών για τα διδακτικά αντικείμενα του ολοήμερου προγράμματος, Υπουργείο Παιδείας δια Βίου Μάθησης και Θρησκευμάτων*, απ. Φ. 12/879/88413 /Γ1, ΦΕΚ 1139 28-7-2010. Διαθέσιμο: http://www.pi-schools.gr/news/FEK-1139B-PS.pdf [τελευταία πρόσβαση: Ιούνιος 2017]

Παιδαγωγικό Ινστιτούτο. (2011). *Πρόγραμμα Σπουδών για τις ΤΠΕ στην Εκπαίδευση*. ΕΣΠΑ 2007-13\Ε.Π. Ε&ΔΒΜ\Α.Π. 1-2-3 «ΝΕΟ ΣΧΟΛΕΙΟ (Σχολείο 21ου αιώνα) – Νέο Πρόγραμμα Σπουδών, Οριζόντια Πράξη» MIS: 295450. Available at: http://ebooks.edu.gr/info/newps/%CE%A0%CE%BB%CE%B7%CF%81%CE%BF%CF%86%CE%BF%CF%81%CE%B9%CE%BA%CE%AE%20%CE%BA%CE%B1%CE%B9%20%CE%9D%CE%AD%CE%B5%CF%82%20%CE%A4%CE%B5%CF%87%CE%BD%CE%BF%CE%BB%CE%BF%CE%B3%CE%AF%CE%B5%CF%82/%CE%A4%CE%A0%CE%95%20%CE%94%CE%B7%CE%BC%CE%BF%CF%84%CE%B9%CE%BA%CF%8C.pdf [τελευταία πρόσβαση: Ιούνιος 2017]

Μικρόπουλος, Τ. Α., Μπέλλου, Ι. (2016). Είναι όλα μαθησιακά αντικείμενα; Στο Α. Τζιμογιάννης (επ.) *Πανελλήνιο Συνέδριο Ψηφιακό Εκπαιδευτικό Υλικό και Ηλεκτρονική Μάθηση 2.0*, σ. 12. Διαθέσιμο: http://eprl.korinthos.uop.gr/eloer2016/?page_id=214

Appendix

A.1 Introductory Letter

Αγαπητές και αγαπητοί εκπαιδευτικοί,

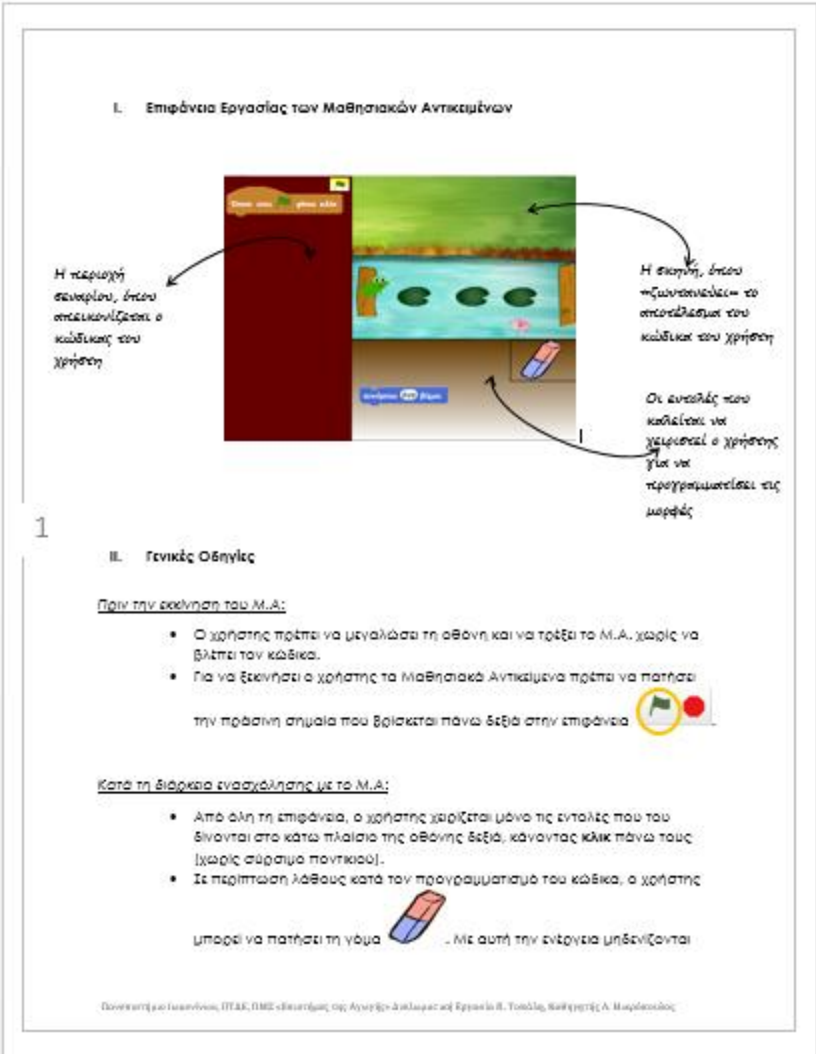
Είμαι μεταπτυχιακή φοιτήτρια του Παιδαγωγικού Τμήματος Δημοτικής Εκπαίδευσης του Πανεπιστημίου Ιωαννίνων στην κατεύθυνση των Φυσικών Επιστημών. Στο πλαίσιο της διπλωματικής μου εργασίας, αναφορικά με την αξιοποίηση των Ψηφιακών Μαθησιακών Αντικειμένων στον προγραμματισμό, διερευνώ αν τα Μαθησιακά Αντικείμενα που δημιούργησα ανταποκρίνονται στις ανάγκες μαθητών Δημοτικού που διδάσκονται προγραμματισμό.

Τα Ψηφιακά Μαθησιακά Αντικείμενα (ΨΜΑ) είναι μια σχετικά πρόσφατη τάση στο χώρο της ηλεκτρονικής μάθησης (e-learning) και χρησιμοποιούνται ευρέως για τη δημιουργία διαδικτυακού εκπαιδευτικού περιεχομένου. Βασίζονται στην ιδέα πως ο εκπαιδευτικός δημιουργεί μικρές μαθησιακές μονάδες, που λειτουργούν αυτόνομα ή μπορούν να συνδυαστούν για την υποστήριξη της εκπαιδευτικής διαδικασίας. Στα κύρια χαρακτηριστικά τους συγκαταλέγονται η επαναχρησιμοποίηση (το ΨΜΑ μπορεί να χρησιμοποιηθεί σε διάφορες περιπτώσεις όπως μαθήματα, τάξεις, τεχνικές, κλπ.), η διαλειτουργικότητα (το ΨΜΑ λειτουργεί σε διάφορα περιβάλλοντα), η διαχειρισιμότητα (το ΨΜΑ μπορεί να τροποποιηθεί).

Τα συγκεκριμένα ΨΜΑ έχουν δημιουργηθεί στο SCRATCH και όχι με άλλα εργαλεία όπως Adobe Flash ή HTML5 καθώς το SCRATCH συγκεντρώνει τα προαναφερθέντα χαρακτηριστικά και επιτρέπει εύκολα στο χρήστη (εκπαιδευτικό αλλά και μαθητή) να το τροποποιεί.

Σας ευχαριστώ για τη συμμετοχή σας στην έρευνα.

B.1 LOs Guidelines



I. **Επιφάνεια Εργασίας των Μαθησιακών Αντικειμένων**

1

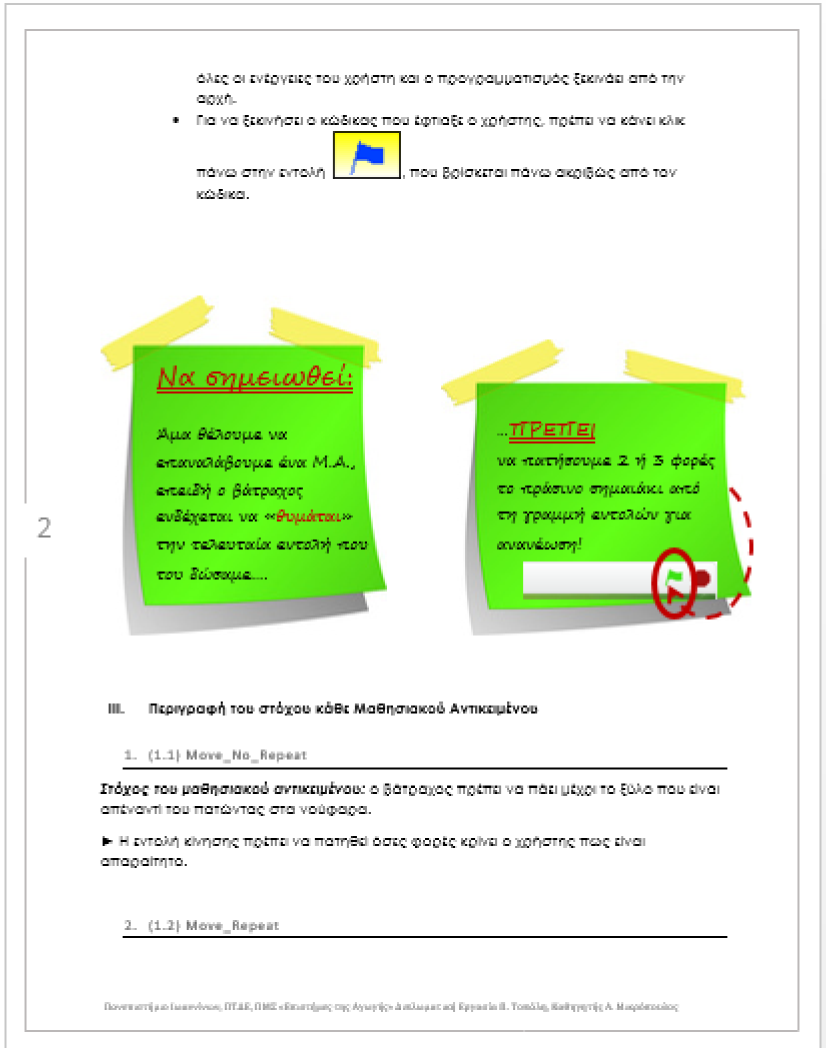
II. **Γενικές Οδηγίες**

Πριν την εκκίνηση του Μ.Α:

- Ο χρήστης πρέπει να μεγαλώσει τη οθόνη και να τρέξει το Μ.Α. χωρίς να βλέπει τον κώδικα.
- Για να ξεκινήσει ο χρήστης τα Μαθησιακά Αντικείμενα πρέπει να πατήσει την πράσινη σημαία που βρίσκεται πάνω δεξιά στην επιφάνεια

Κατά τη διάρκεια ενασχόλησης με το Μ.Α:

- Από όλη τη επιφάνεια, ο χρήστης χειρίζεται μόνο τις εντολές που του δίνονται στο κάτω πλαίσιο της οθόνης δεξιά, κάνοντας **κλικ** πάνω τους [χωρίς σύρσιμο ποντικιού].
- Σε περίπτωση λάθους κατά τον προγραμματισμό του κώδικα, ο χρήστης μπορεί να πατήσει τη γόμα . Με αυτή την ενέργεια μηδενίζονται



όλες οι ενέργειες του χρήστη και ο προγραμματισμός ξεκινάει από την αρχή.

- Για να ξεκινήσει ο κώδικας που έφτιαξε ο χρήστης, πρέπει να κάνει κλικ πάνω στην εντολή , που βρίσκεται πάνω ακριβώς από τον κώδικα.

2

III. **Περιγραφή του στόχου κάθε Μαθησιακού Αντικειμένου**

1. (1.1) Move_No_Repeat

Στόχος του μαθησιακού αντικειμένου: ο βάτραχος πρέπει να πάει μέχρι το ξύλο που είναι απέναντι του πατώντας στα νούφαρα.

► Η εντολή κίνησης πρέπει να πατηθεί όσες φορές κρίνει ο χρήστης πως είναι απαραίτητο.

2. (1.2) Move_Repeat

Στόχος του μαθησιακού αντικειμένου: ο βάτραχος πρέπει να πάει μέχρι το ξύλο που είναι απέναντί του πατώντας στα νούφαρα.

► Αυτή τη φορά ο χρήστης καλείται να αξιοποιήσει την εντολή **επανάληψης**. Μόλις ο χρήστης πατήσει την εντολή, αυτόματα θα ερωτηθεί "πόσες φορές να επαναληφθεί" η οποία εντολή περιλαμβάνεται στην εντολή επανάληψης.

3. (1.3) Move_Complex

Στόχος του μαθησιακού αντικειμένου: ο βάτραχος πρέπει να πάει μέχρι το ξύλο που είναι απέναντί του και έπειτα να γυρίσει πίσω στην αρχική του θέση πατώντας στα νούφαρα.

► Να αξιοποιηθεί η εντολή της **επανάληψης**, όσες φορές χρειάζεται.

4. If_else

Στόχος του μαθησιακού αντικειμένου: Εάν το νούφαρο λείπει, βάτραχος πρέπει να πει κπο νούφαρο εξαφανίστηκε, αλλιώς να πάει πάνω στο νούφαρο.

► Ο χρήστης καλείται να προγραμματίσει το βάτραχο και να παρατηρήσει τι αλλαγές συναντά όταν υπάρχει και όταν δεν υπάρχει νούφαρο.

3

5. Broadcast_ When I receive

Στόχος του μαθησιακού αντικειμένου: ο βάτραχος πρέπει να μεταδώσει ένα μήνυμα στην μύγα, η οποία μόλις το λάβει πρέπει να εξαφανιστεί.

► Αυτή τη φορά η επιφάνεια που έχει μπροστά του ο χρήστης αλλάζει.

Στο κάτω και δεξιά πλαίσιο της οθόνης υπάρχουν εντολές χωρισμένες σε δυο στήλες και αντιστοιχούν στην κάθε μια από της δυο μορφές. Ο χρήστης και πάλι επιδρά μέσω κλικ πάνω στις εντολές και καλείται να προγραμματίσει δυο σενάρια (ξεκινώντας από το σενάριο του βατράχου, έπειτα προχωρά στον προγραμματισμό των εντολών της μύγας).

4

Πανεπιστήμιο Ιωαννίνων, ΠΤΔΕ, ΠΜΣ «Επιστήμες της Αγωγής» Διπλωματική Εργασία Β. Τσιόλης, Καθηγητής Α. Μικρόπουλος

C.1 Survey Questionnaire

C.1.1 Questions about Demographic Data

Ονοματεπώνυμο

……………………......

Φύλο

( ) Άντρας

( ) Γυναίκα

Ηλικία

( ) < 30 ετών

( ) 30-39

( ) 40-49

( ) 50-59

( ) > 60 ετών

Σπουδές

( ) Πτυχίο ΑΕΙ

( ) Πτυχίο ΤΕΙ

( ) Μεταπτυχιακό

( ) Διδακτορικό

Βασικό Πτυχίο

( ) Πληροφορική/ Μηχανικός Υ/Π

( ) Φυσικομαθηματική

( ) άλλο………………

Σχολική Βαθμίδα Διδασκαλίας

( ) Δημοτικό

( ) Γυμνάσιο

( ) Λύκειο

( ) άλλο………………

Προϋπηρεσία στην Εκπαίδευση

( ) < 6 έτη

( ) 6-10 έτη

( ) > 10 έτη

C.1.2 Questions regarding the Evaluation of the LOs

1. ***Ποιότητα Περιεχομένου*** (*εγκυρότητα, ακρίβεια, ισορροπημένη παρουσίαση των ιδεών, και κατάλληλο επίπεδο λεπτομέρειας*)

Το περιεχόμενο είναι απαλλαγμένο από σφάλματα και παρουσιάζεται χωρίς προκαταλήψεις ή παραλείψεις που θα μπορούσαν να παραπλανήσουν τους μαθητές. Οι πιθανές διαφορές πολιτιστικών και εθνοτικών ομάδων παρουσιάζονται με έναν ισορροπημένο και ευαίσθητο τρόπο.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Το περιεχόμενο είναι επιστημονικά ορθό.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Τα γραφικά τονίζουν τα βασικά σημεία και τις σημαντικές ιδέες με το κατάλληλο επίπεδο λεπτομέρειας.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

2. ***Συνάφεια με τους μαθησιακούς στόχους*** *(συνάφεια με τους μαθησιακούς στόχους, τις δραστηριότητες, τις αξιολογήσεις και τα χαρακτηριστικά του μαθητή)*

Οι μαθησιακοί στόχοι είναι κατάλληλοι για τους μαθητές που πρόκειται να χρησιμοποιήσουν το μαθησιακό αντικείμενο. Το μαθησιακό αντικείμενο λειτουργεί αυτόνομα και συμβάλλει στην επίτευξη των μαθησιακών στόχων.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Οι δραστηριότητες, το περιεχόμενο και η τυχόν αξιολόγηση του μαθητή που περιλαμβάνεται είναι συναφείς με τους μαθησιακούς στόχους.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Το μαθησιακό αντικείμενο λειτουργεί αυτόνομα και συμβάλλει στην επίτευξη των μαθησιακών στόχων.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

3. ***Ανατροφοδότηση και Προσαρμογή*** *(προσαρμοστικό περιεχόμενο ή ανατροφοδότηση ως αποτέλεσμα της διαφορετικής απάντησης και του τύπου μάθησης κάθε μαθητή)*

Το μαθησιακό αντικείμενο έχει τη δυνατότητα να προσαρμόσει διδακτικά μηνύματα ή δραστηριότητες σύμφωνα με τις ιδιαίτερες ανάγκες ή τα χαρακτηριστικά του μαθητή.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Το μαθησιακό αντικείμενο έχει τη δυνατότητα να τροποποιεί την προσομοίωση ανάλογα με τις παραμέτρους που μεταβάλει ο μαθητής.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Η χρήση του μαθησιακού αντικείμενου μπορεί να προσαρμοστεί για διαφορετικούς μαθητές.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

4. ***Κίνητρο*** *(δυνατότητα να παρακινεί και να προκαλεί το ενδιαφέρον στους μαθητές)*

Το μαθησιακό αντικείμενο παρέχει ισχυρό κίνητρο για χρήση. Το περιεχόμενό του είναι σχετικό με τα ενδιαφέροντα των μαθητών.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Προσφέρει ποικιλία επιλογών, αυθεντικές δραστηριότητες, πολυμέσα, διαδραστικότητα, χιούμορ, ή παιγνιώδεις διαδικασίες. Παρέχει ρεαλιστικές προσδοκίες και κριτήρια για επιτυχία.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Οι μαθητές είναι πιθανό να παρουσιάσουν αυξημένο ενδιαφέρον για το μάθημα, αφού δουλέψουν με το μαθησιακό αντικείμενο.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

5. ***Σχεδιασμός Παρουσίασης*** *(ο σχεδιασμός των οπτικών και ακουστικών πληροφοριών σχετίζεται με την ενίσχυση της μάθησης και την αποτελεσματική νοητική επεξεργασία)*

Το περιεχόμενο συμβάλλει σε θετικά μαθησιακά αποτελέσματα. Η γλώσσα είναι σαφής, περιεκτική και χωρίς λάθη.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Τα γραφικά ελαχιστοποιούν την οπτική αναζήτηση.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Το κείμενο είναι ευανάγνωστο.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Τα πολυμεσικά στοιχεία (γραφικά, ήχος) είναι ευχάριστα και δεν παρεμποδίζουν τους μαθησιακούς στόχους.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

6. ***Λειτουργική αλληλεπίδραση*** *(ευκολία πλοήγησης, προβλεψιμότητα της διεπαφής χρήστη, ποιότητα διεπαφής)*

Η διεπαφή πληροφορεί έμμεσα τους μαθητές πώς να αλληλεπιδρούν με το αντικείμενο, ή υπάρχουν σαφείς οδηγίες χρήσης. Η πλοήγηση (λειτουργία) είναι εύκολη, διαισθητική και άμεση.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Η συμπεριφορά της διεπαφής είναι συνεπής σε όλα τα σημεία του μαθησιακού αντικειμένου και προβλέψιμη.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

7. ***Επαναχρησιμοποίηση*** *(το μαθησιακό αντικείμενο μπορεί να χρησιμοποιηθεί σε ποικίλα περιβάλλοντα μάθησης με μαθητές διαφορετικών επιπέδων)*

Το μαθησιακό αντικείμενο είναι αυτοδύναμο και μπορεί να χρησιμοποιηθεί με ευκολία σε διαφορετικά μαθήματα, διδακτικούς σχεδιασμούς και πλαίσια χωρίς τροποποίηση.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

Λειτουργεί αποτελεσματικά σε μαθητές διαφορετικών επιπέδων με την προσαρμογή του περιεχομένου ή με την παροχή συμπληρωματικού περιεχομένου.

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

8. ***Προσβασιμότητα (προαιρετική απάντηση)*** *(κατάλληλη σχεδίαση και παρουσίαση των πληροφοριών για χρήση από άτομα με ειδικές ανάγκες και χρήση σε φορητές συσκευές)*

Το μαθησιακό αντικείμενο είναι προσβάσιμο με τη χρήση βοηθητικών συσκευών για τους χρήστες με αισθητηριακές και κινητικές αναπηρίες. Επίσης είναι προσβάσιμο μέσω φορητών συσκευών. (Ακολουθεί τις κατευθυντήριες γραμμές της IMS για τις εκπαιδευτικές εφαρμογές και συμμορφώνεται με το W3C Web Content Accessibility Guidelines σε επίπεδο «AAA».)

( ) 1 καθόλου

( ) 2 λίγο

( ) 3 μέτρια

( ) 4 πολύ

( ) 5 πάρα πολύ

C.1.3 Open Question

Ώρα για ανατροφοδότηση! Παρακαλώ να σημειώσετε τα σχόλια σας ή/και τις προτάσεις σας!

…………………………………………………………………………………………………..

D.1 Sprites' Codes of the LO "*MoveNoRepeat*"

Table D.1.1 Main frog's code

Sprite	
Description	main character of the LO
LO	MoveNoRepeat

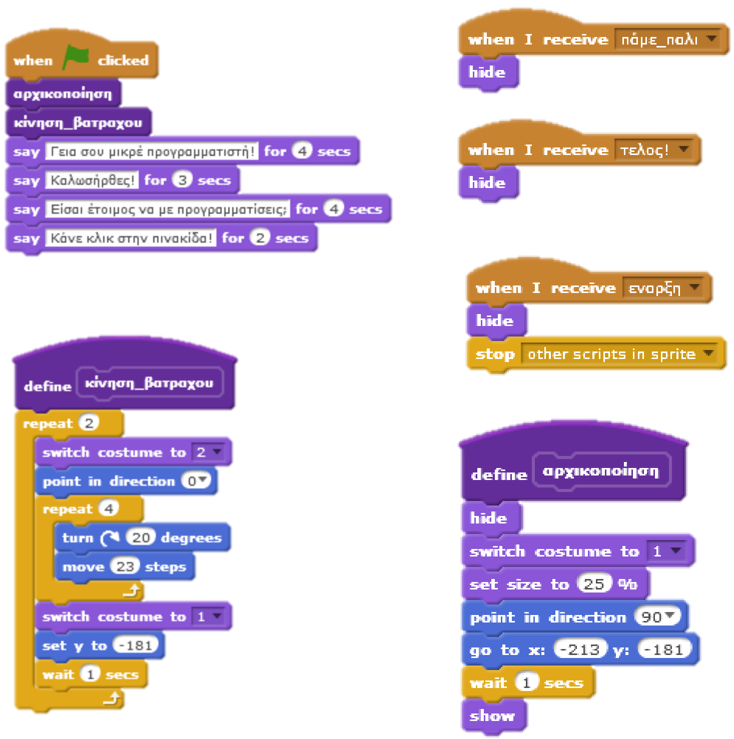


Table D.1.2 Initial frog's code

Sprite	
Description	Initial frog of the first scene of the LO
LO	MoveNoRepeat

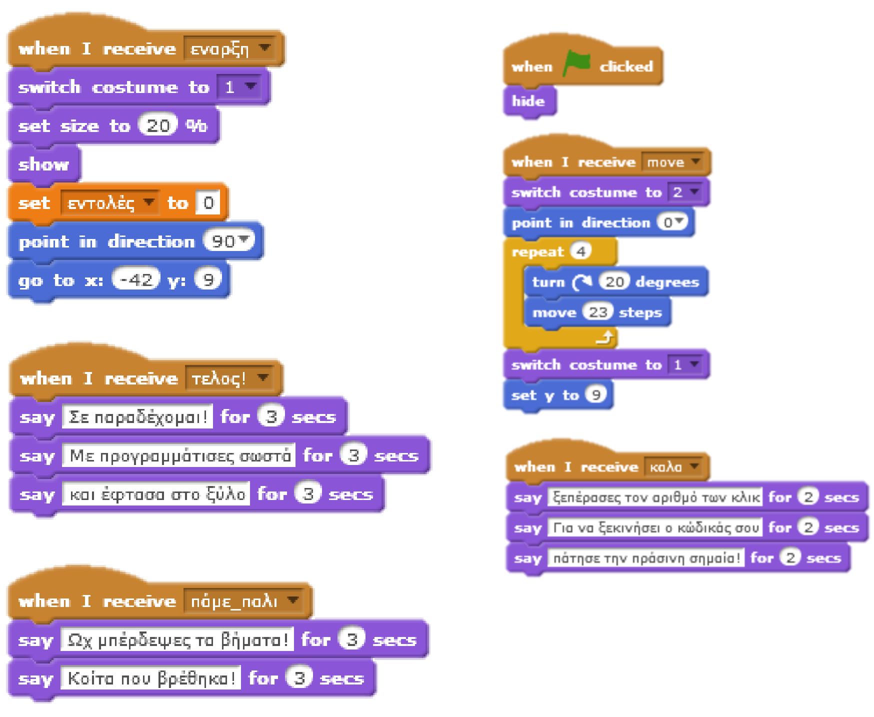


Table D.1.3. Move-block's code

Sprite	
Description	Move block
LO	MoveNoRepeat

Table D.1.4. Event-block's code

Sprite	
Description	Event block
LO	MoveNoRepeat

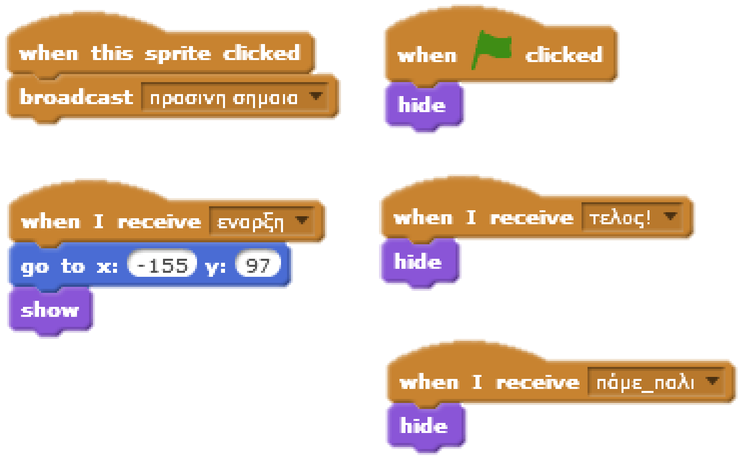


Table D.1.5. Arrow’s code

Sprite	
Description	Arrow sprite
LO	MoveNoRepeat

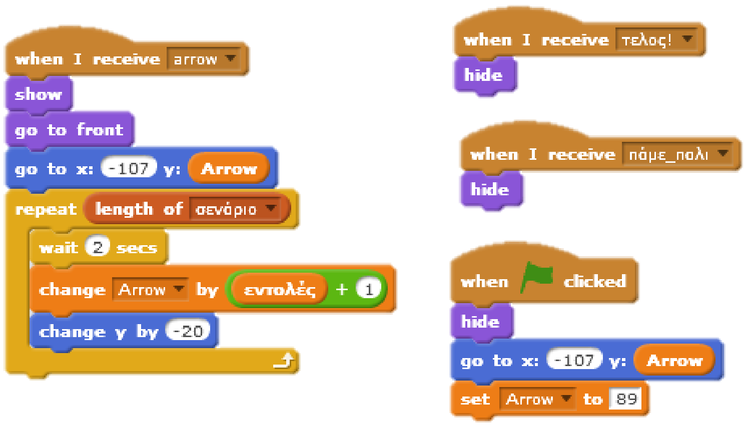


Table D.1.6. Eraser’s code

Sprite	
Description	Eraser sprite
LO	MoveNoRepeat

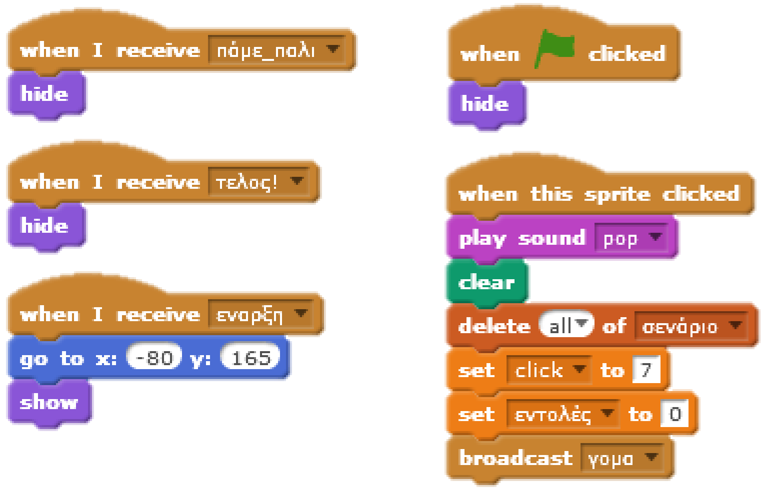
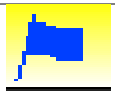


Table D.1.7. Blue-flag's code

Sprite	
Description	Blue flag sprite
LO	MoveNoRepeat

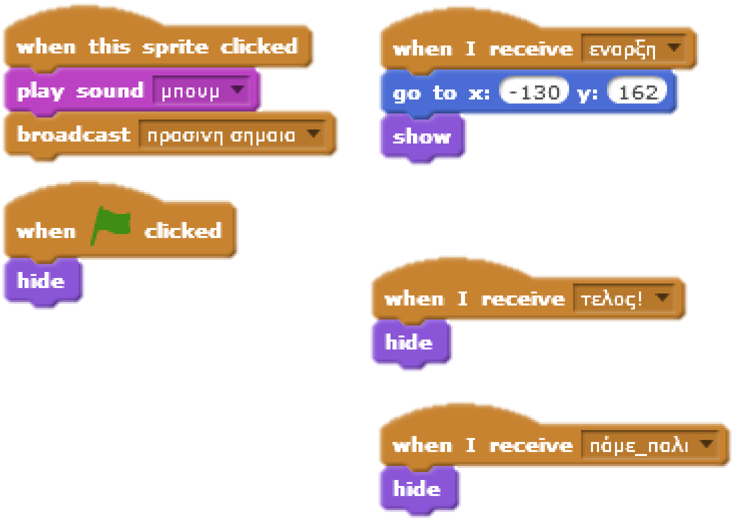


Table D.1.8 Go's code

Sprite	
Description	Go sprite
LO	MoveNoRepeat

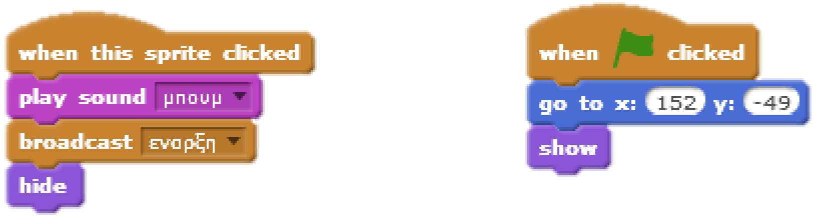
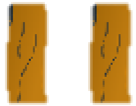


Table D.1.9. Woods' codes

Sprite	
Description	Woods sprites
LO	MoveNoRepeat

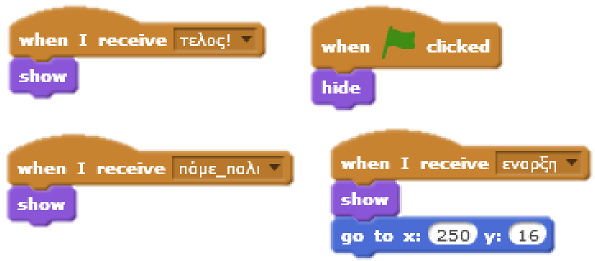
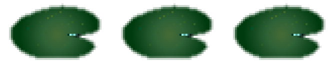
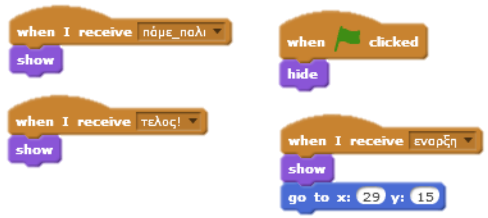
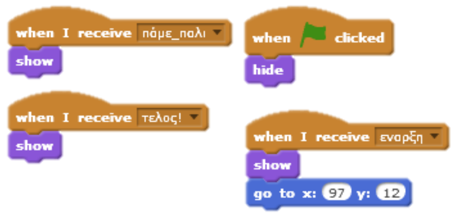


Table D.1.10. Waterlilies' codes

Sprite	
Description	Waterlilies sprites
LO	MoveNoRepeat

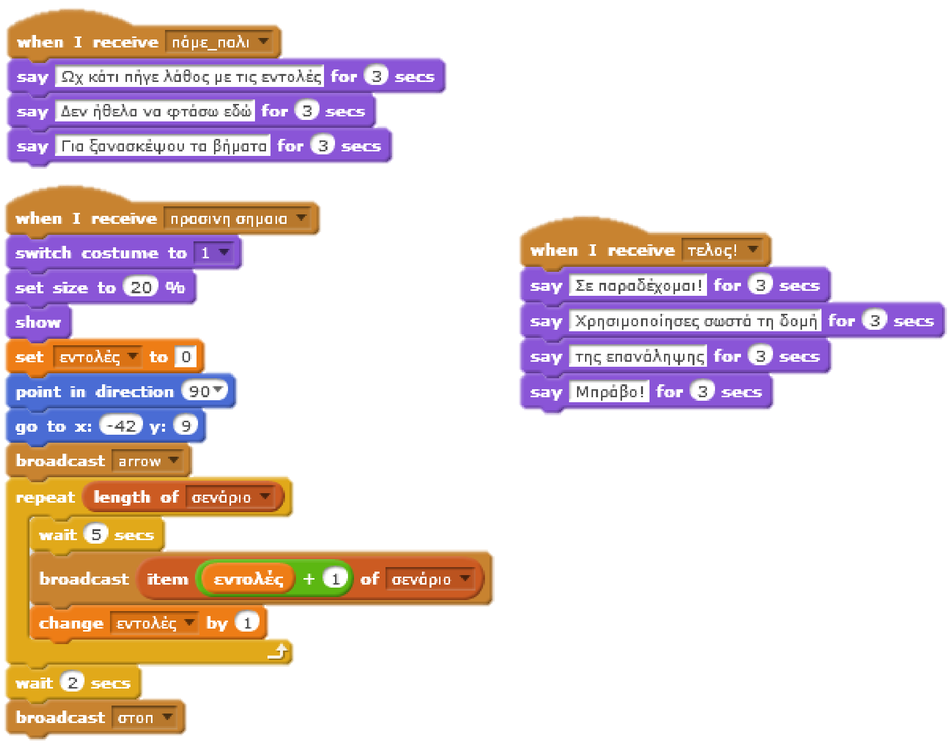




D.2 Sprites' Codes of the LO "*MoveRepeat*"

Table D.2.1 Main frog's code

Sprite	
Description	Main character of the LO
LO	MoveRepeat



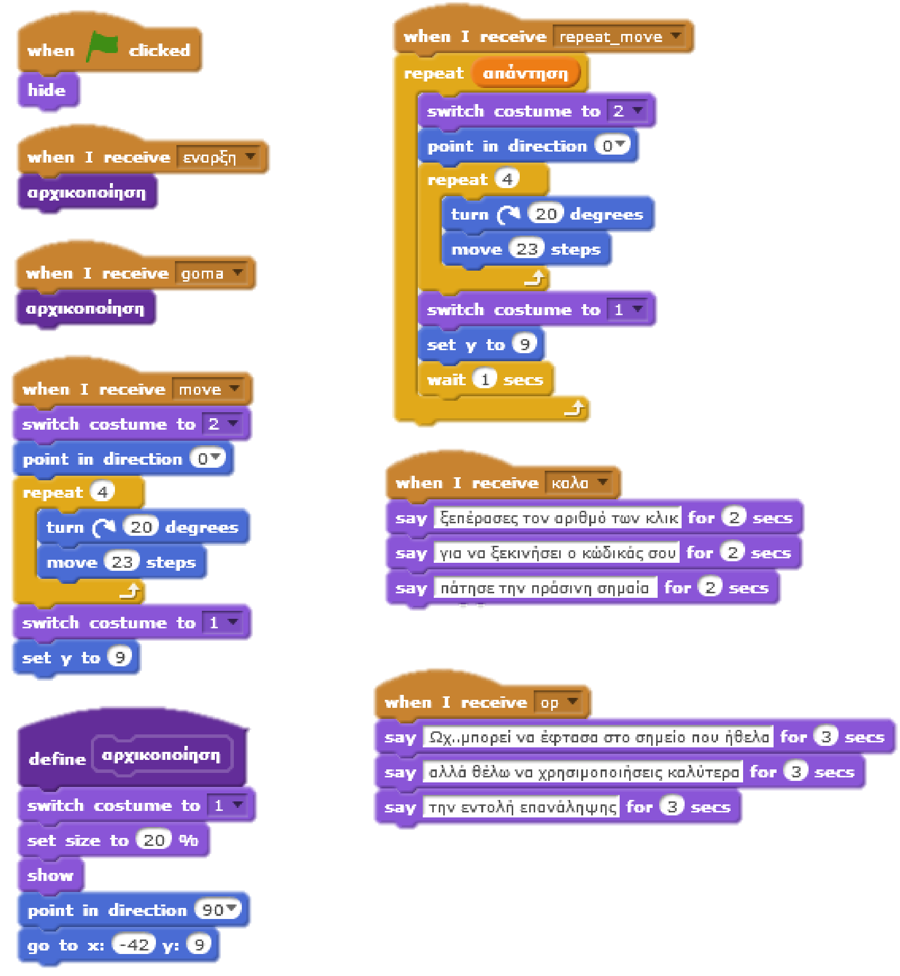
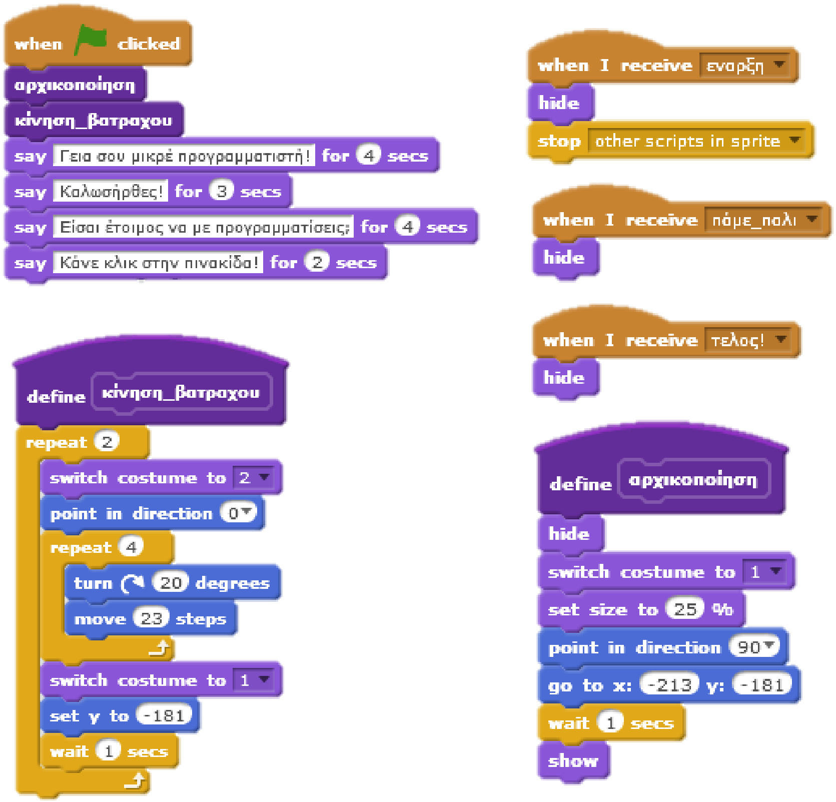


Table D. 2. 2. Initial frog's code

Sprite	
Description	Initial frog of the first scene of the LO
LO	MoveRepeat

Table D.2.3 Arrow's code

Sprite	
Description	Arrow sprite
LO	MoveRepeat



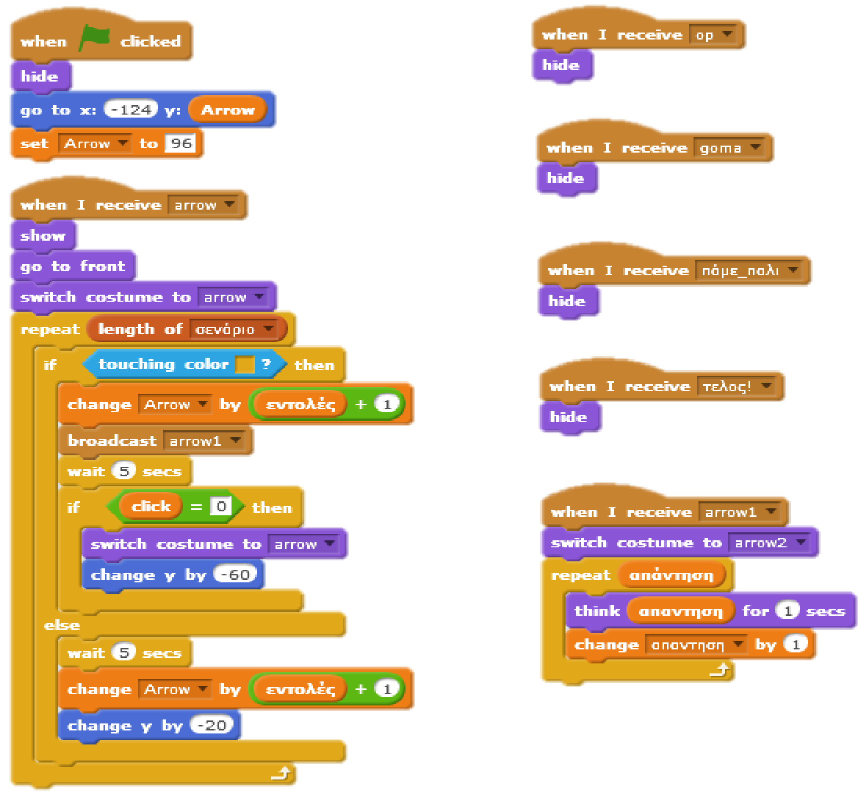
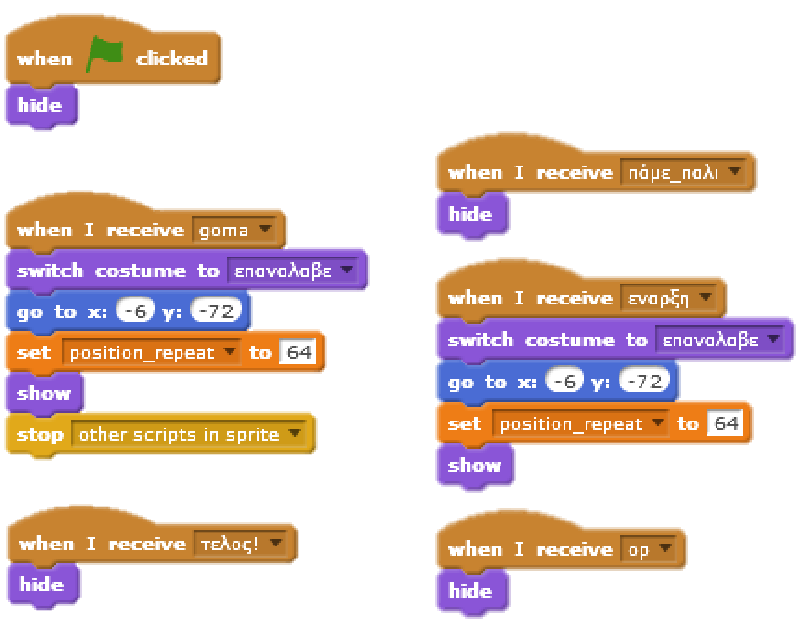


Table D. 2. 4 Repeat's Code

Sprite	
Description	Repeat block of the LO
LO	MoveRepeat



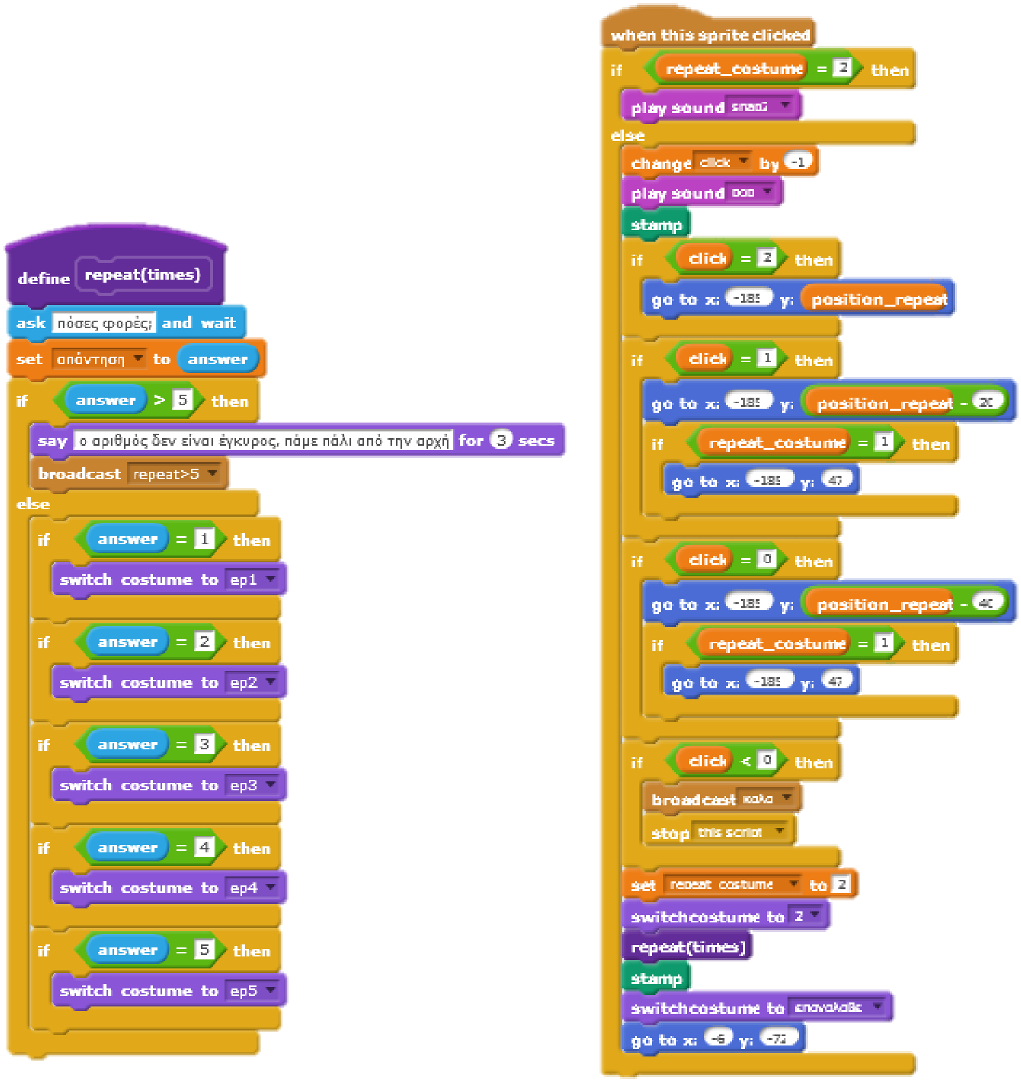


Table D.2.5 Move's Code

Sprite	
Description	Move block
LO	MoveRepeat

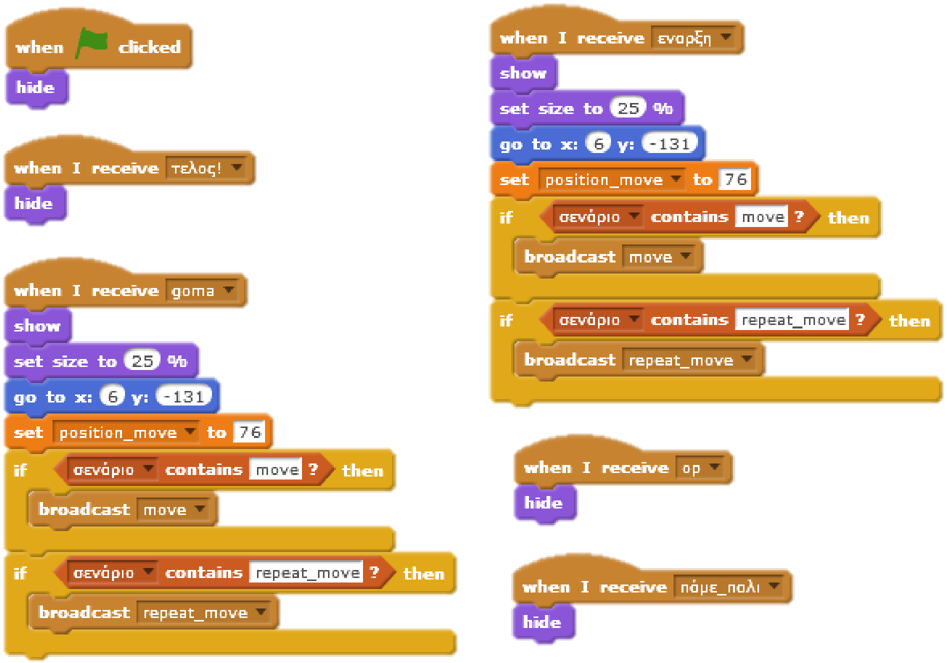


Table D.2.6 Eraser's Code

Sprite	
Description	Eraser sprite
LO	MoveRepeat

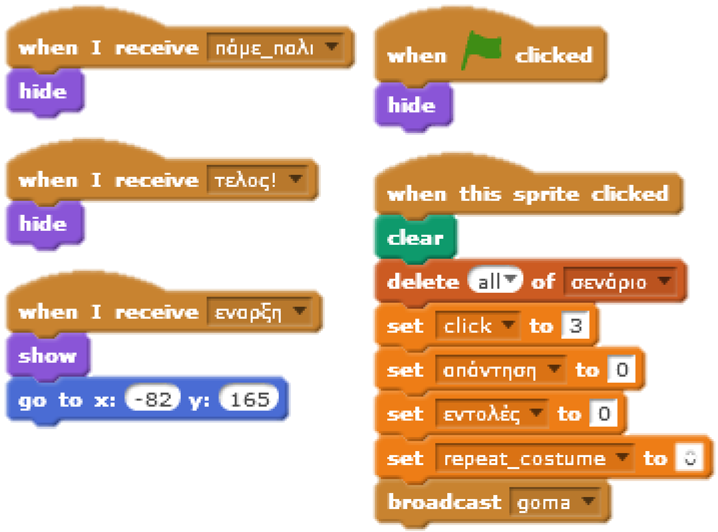
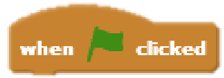


Table D.2.7. Event's code

Sprite	
Description	Event block
LO	MoveRepeat

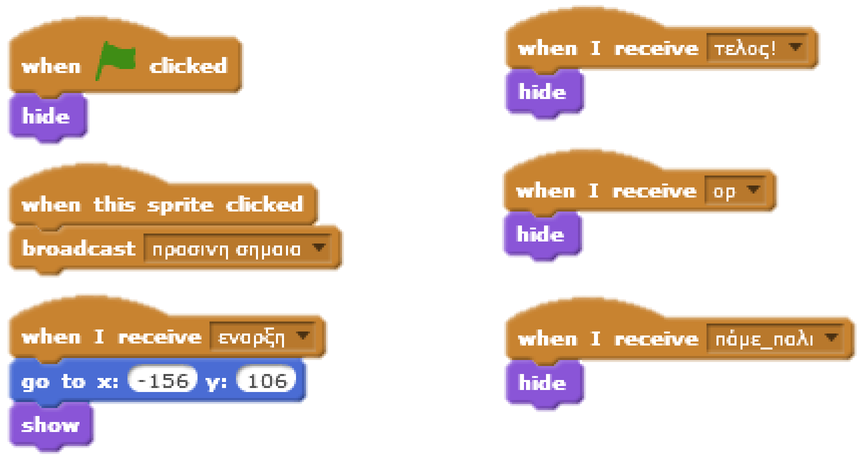
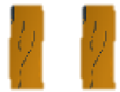


Table D.2.8 Go' code

Sprite	
Description	Go sprite
LO	MoveRepeat

Table D.2.9 Woods' code

Sprites	
Description	Woods sprites
LO	MoveRepeat

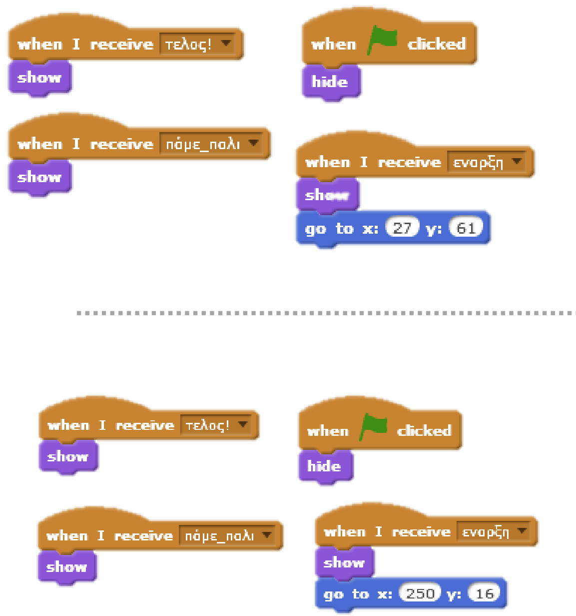
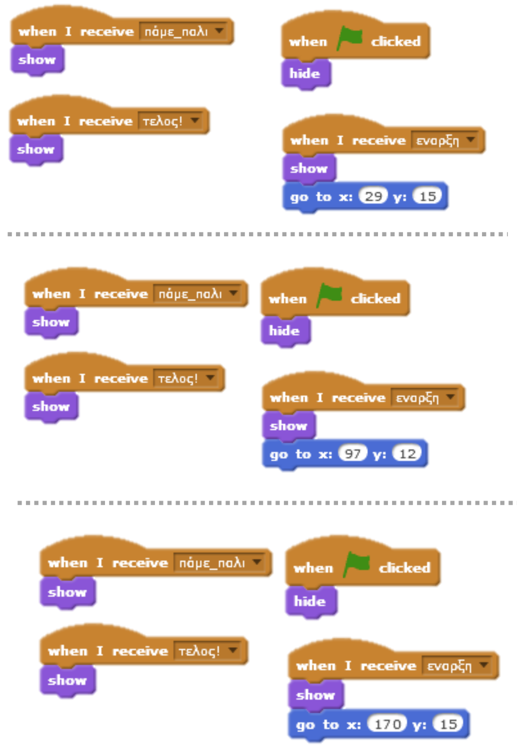


Table D. 2.10 Waterlilies' code

Sprites	
Description	Waterlilies sprites
LO	MoveRepeat



D.3 Sprites' Codes of the LO "*MoveComplex*"

Table D.3.1. Main frog's code

Sprite	
Description	Main character of the LO
LO	MoveComplex

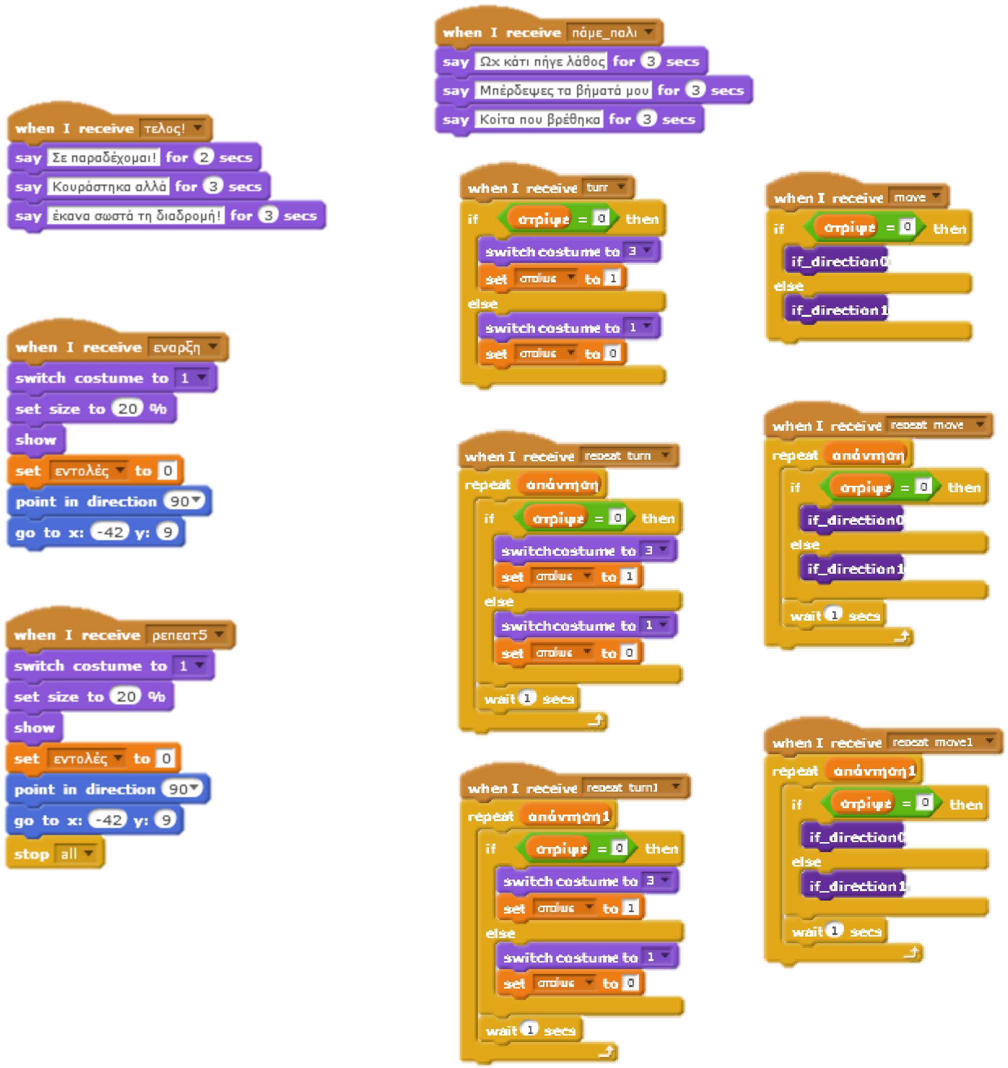
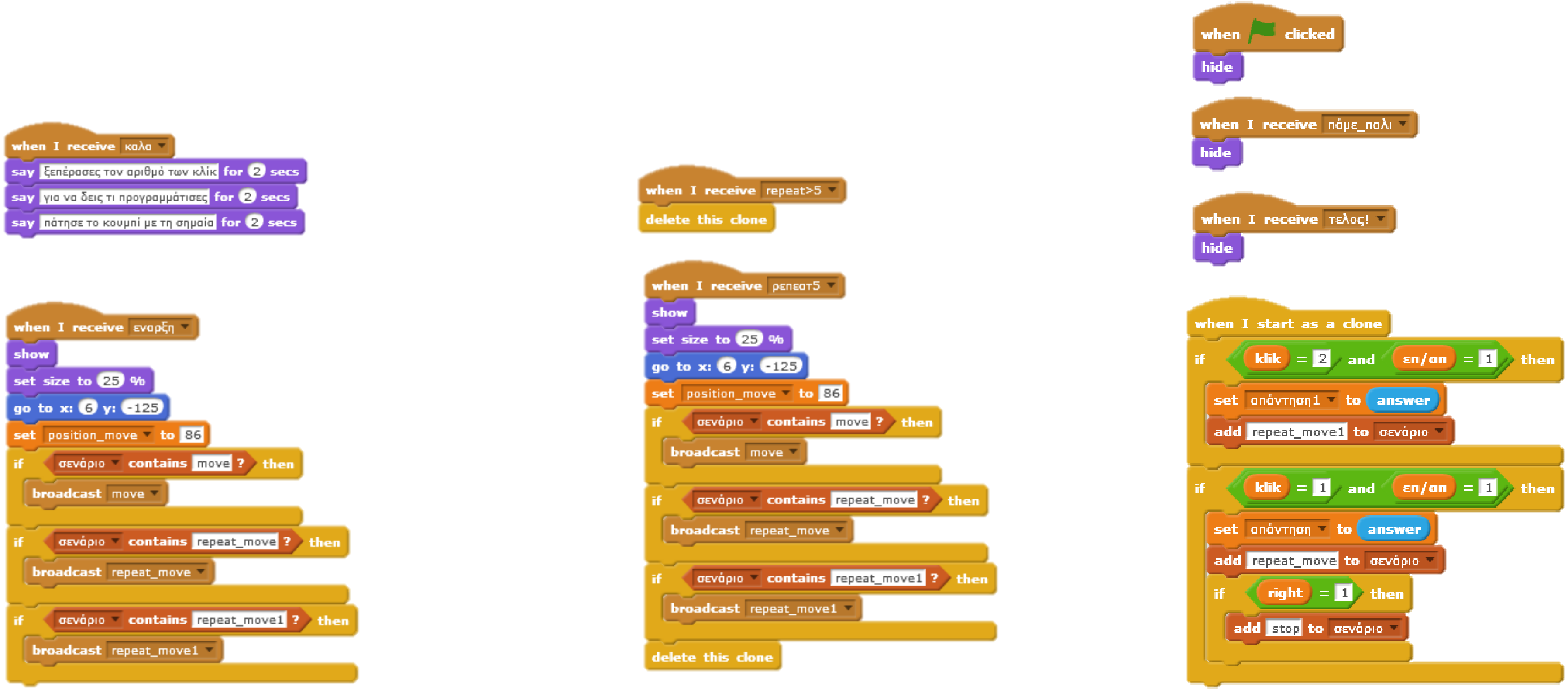


Table D.3.2 Move's code

Sprite	
Description	Move block
LO	MoveComplex



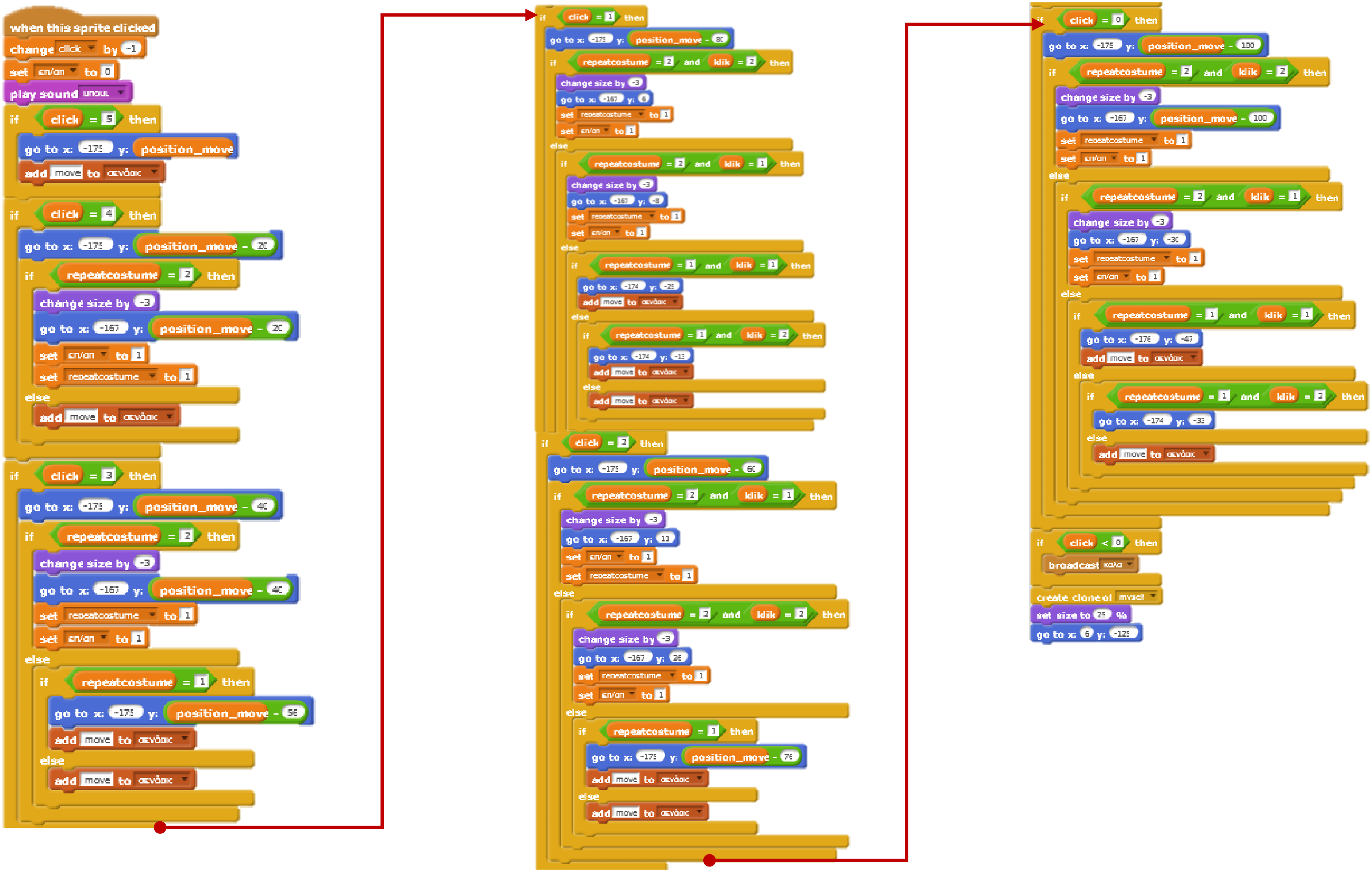
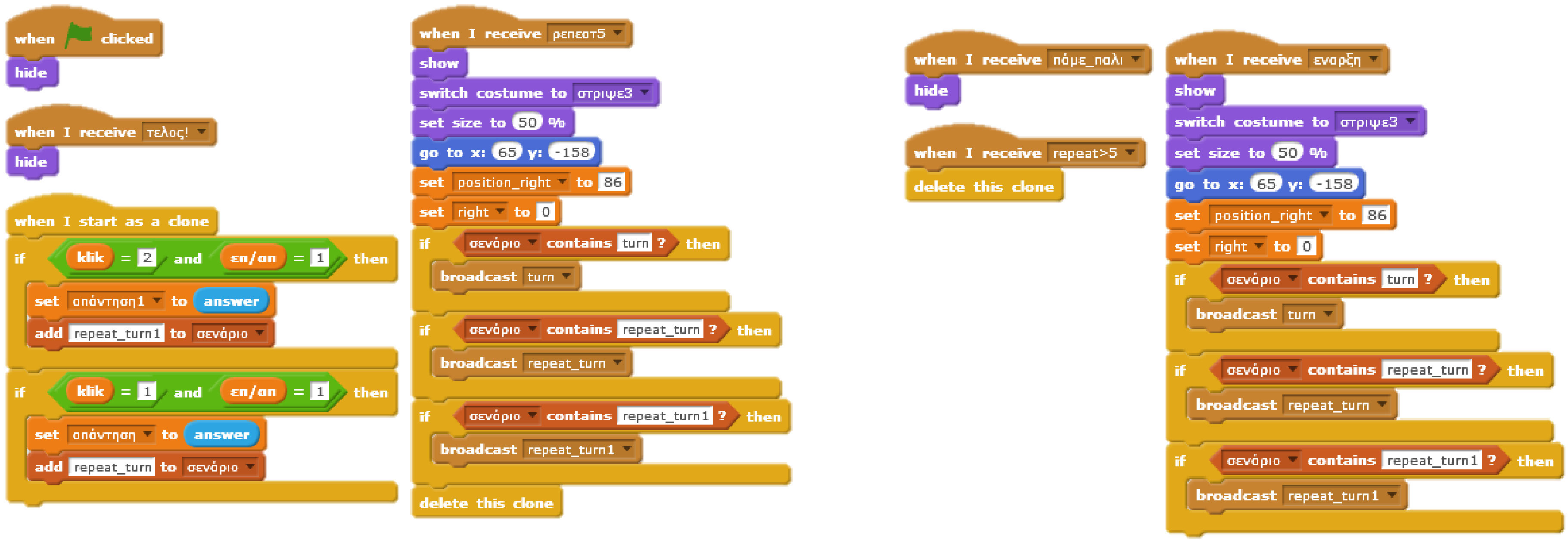

when this sprite clicked
change click by -1
set play sound
if click = 5 then
go to x: -175 y: position_move
add move to
if click = 4 then
go to x: -175 y: position_move - 20
if repeatcostume = 2 then
change size by -3
go to x: -167 y: position_move - 20
set to 1
set repeatcostume to 1
else
add move to
if click = 3 then
go to x: -175 y: position_move - 40
if repeatcostume = 2 then
change size by -3
go to x: -167 y: position_move - 40
set repeatcostume to 1
set to 1
else
if repeatcostume = 1 then
go to x: -175 y: position_move - 58
add move to
else
add move to
if click = 1 then
go to x: -175 y: position_move - 80
if repeatcostume = 2 and klik = 2 then
change size by -3
else
if click = 2 then
go to x: -175 y: position_move - 60
if click = 0 then
go to x: -175 y: position_move - 100
if click < 0 then
broadcast
create clone of myself
set size to 25 %
go to x: 6 y: -125

Table D.3.3. Turn's code

Sprite	στρίψε προς την αντίθετη κατεύθυνση
Description	Turn Block
LO	MoveComplex



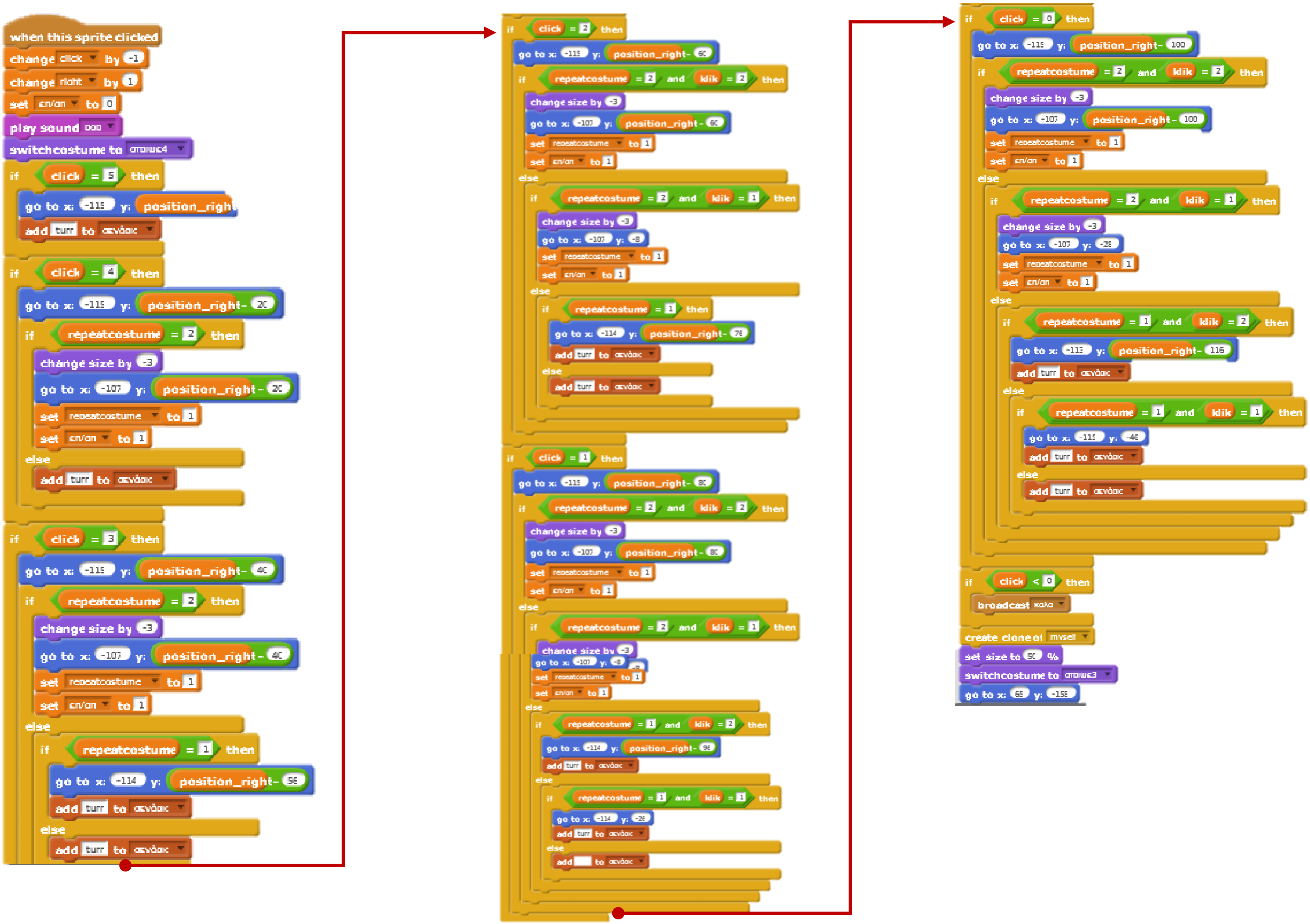
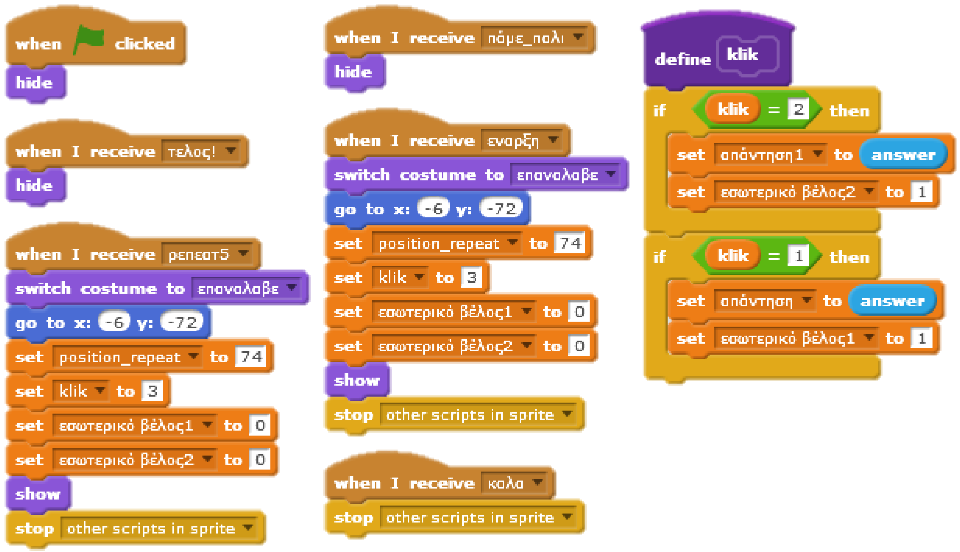
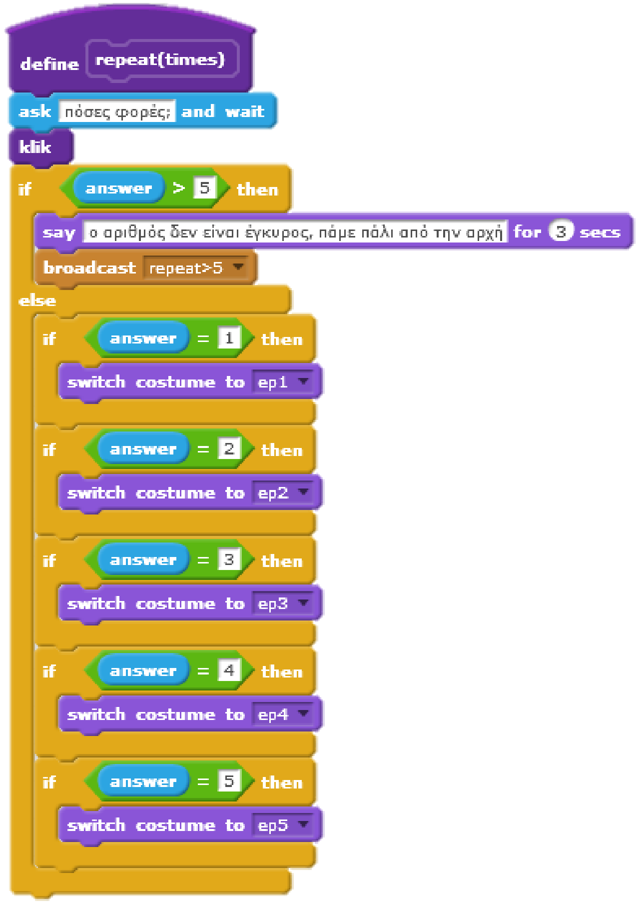
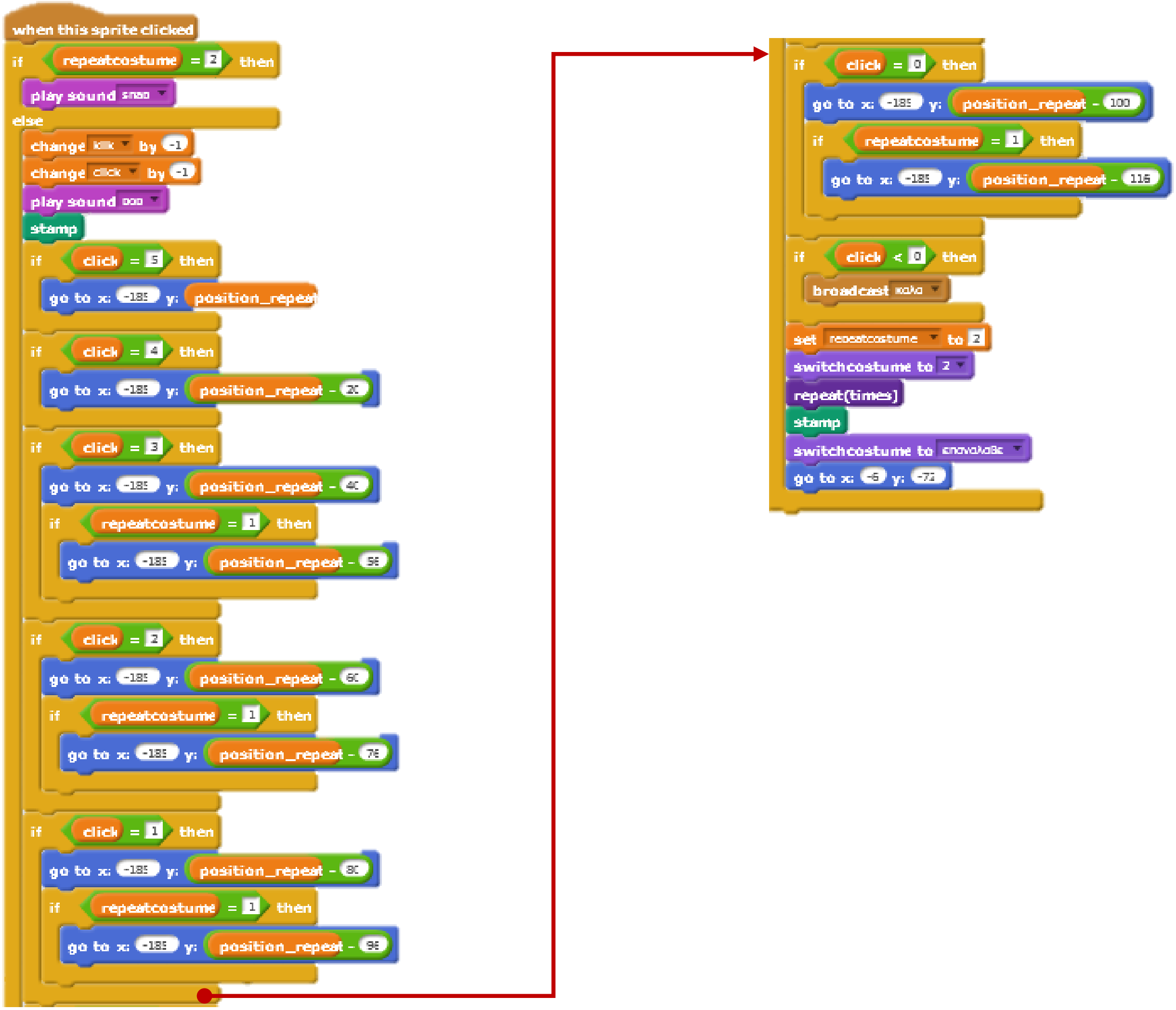


Table D.3.4 Repeat's Code

Sprite	
Description	Repeat block
LO	MoveComplex







when this sprite clicked
if repeatcostume = 2 then
play sound
else
change by -1
change click by -1
play sound
stamp
if click = 5 then
go to x: y: position_repeat
if click = 4 then
go to x: y: position_repeat -
if click = 3 then
go to x: y: position_repeat -
if repeatcostume = 1 then
go to x: y: position_repeat -
if click = 2 then
go to x: y: position_repeat -
if repeatcostume = 1 then
go to x: y: position_repeat -
if click = 1 then
go to x: y: position_repeat -
if repeatcostume = 1 then
go to x: y: position_repeat -
if click = 0 then
go to x: y: position_repeat - 100
if repeatcostume = 1 then
go to x: y: position_repeat -
if click < 0 then
broadcast
set repeatcostume to 2
switchcostume to 2
repeat(times)
stamp
switchcostume to
go to x: y:

Table D.3.5 Initial Frog's Code

Sprite	
Description	Frog of the initial scene of the LO
LO	MoveComplex

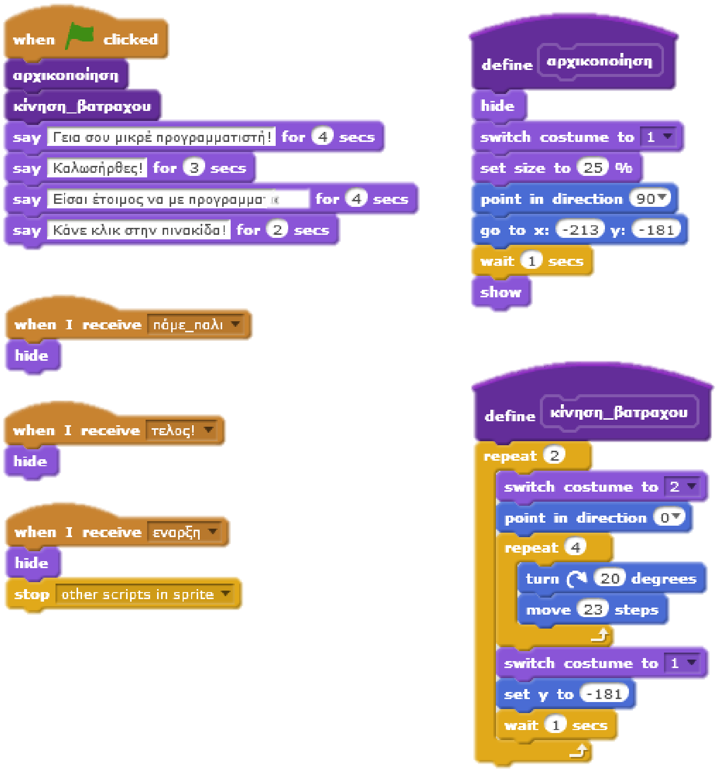


Table D.3.6. Eraser's Code

Sprite	
Description	Eraser sprite
LO	MoveComplex

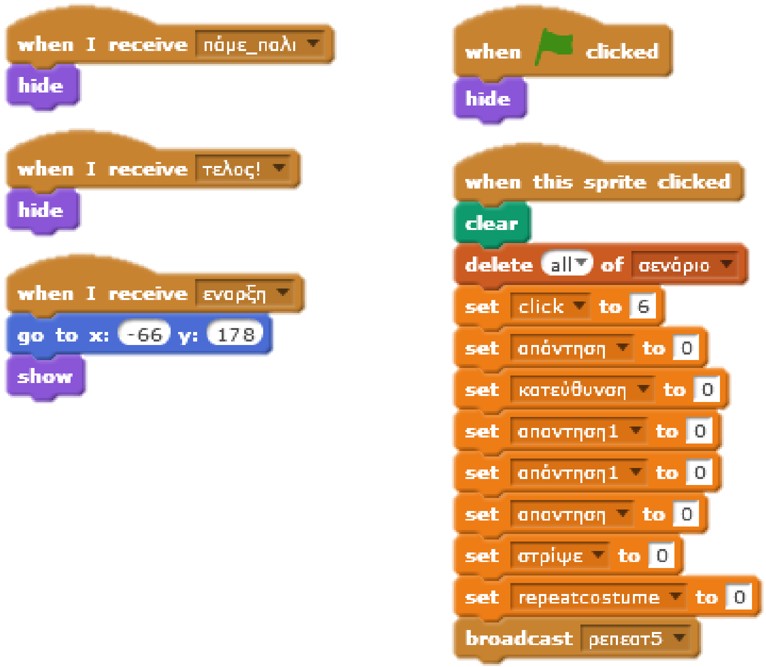


Table D.3.7 Arrow's code

Sprite	
Description	Arrow sprite
LO	MoveComplex

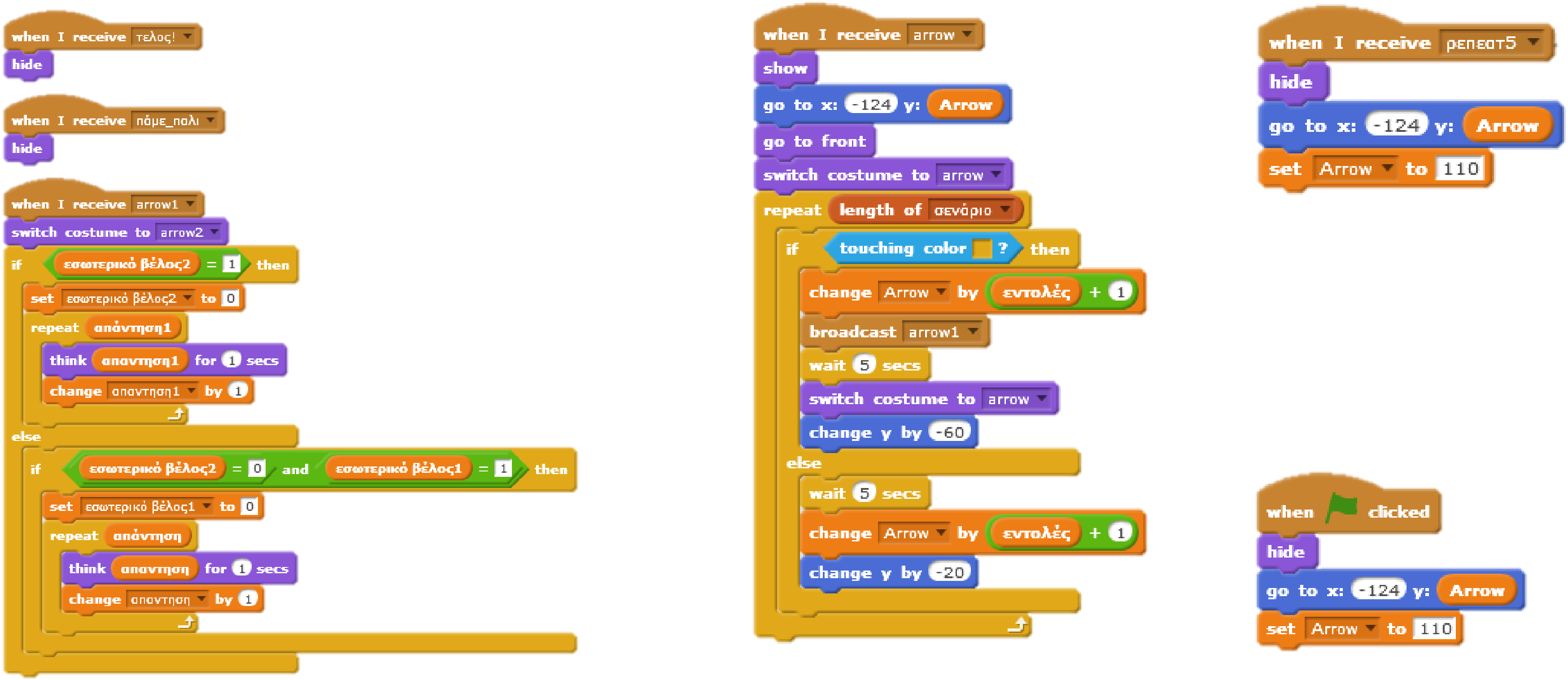
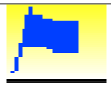


Table D. 3.8. Flag's code

Sprite	
Description	Flag sprite of the LO
LO	MoveComplex

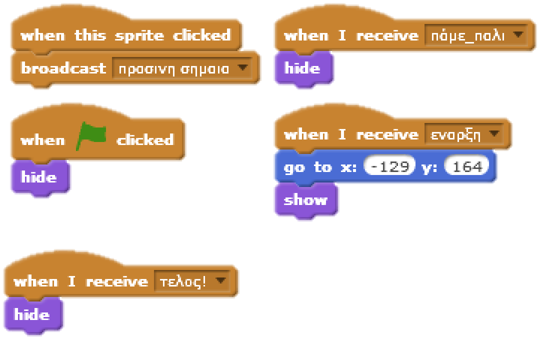


Table D.3.9 Go's code

Sprite	
Description	Go sprite of the LO
LO	MoveComplex

Table D.3.10. Event's code

Sprite	
Description	Event sprite of the LO
LO	MoveComplex

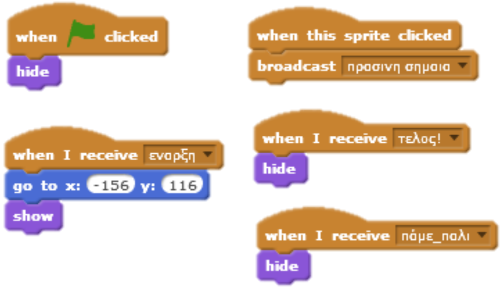
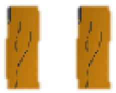


Table D.3.10 Woods' codes

Sprite	
Description	Woods spites
LO	MoveComplex

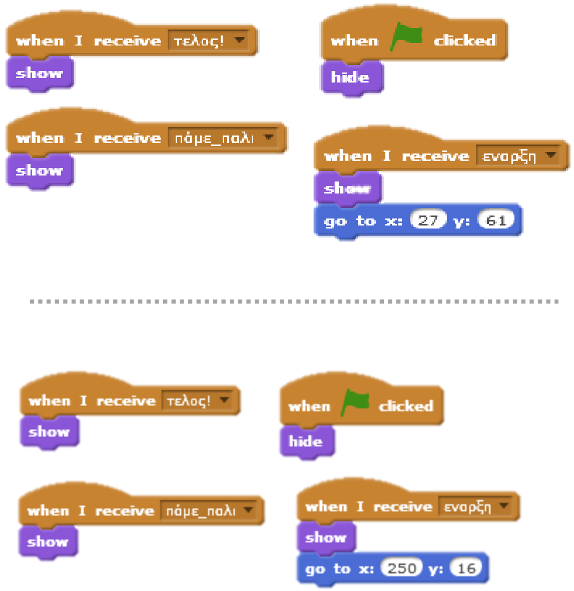
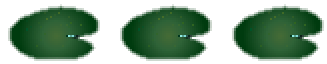
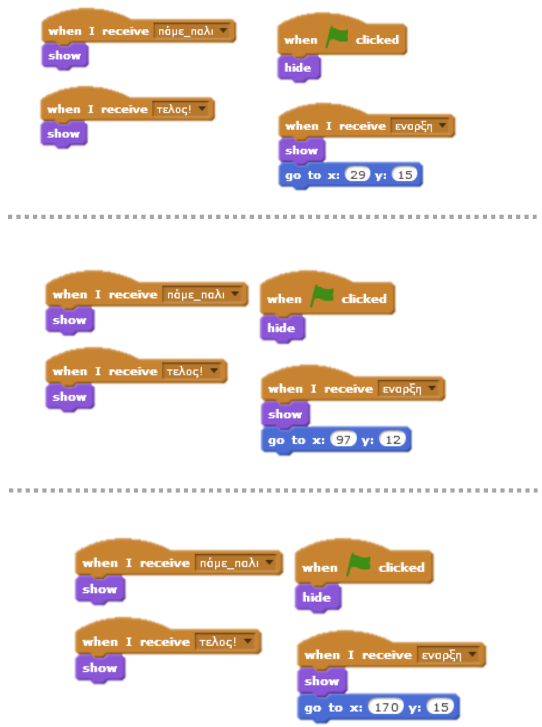


Table D.3.11 Waterlilies' codes

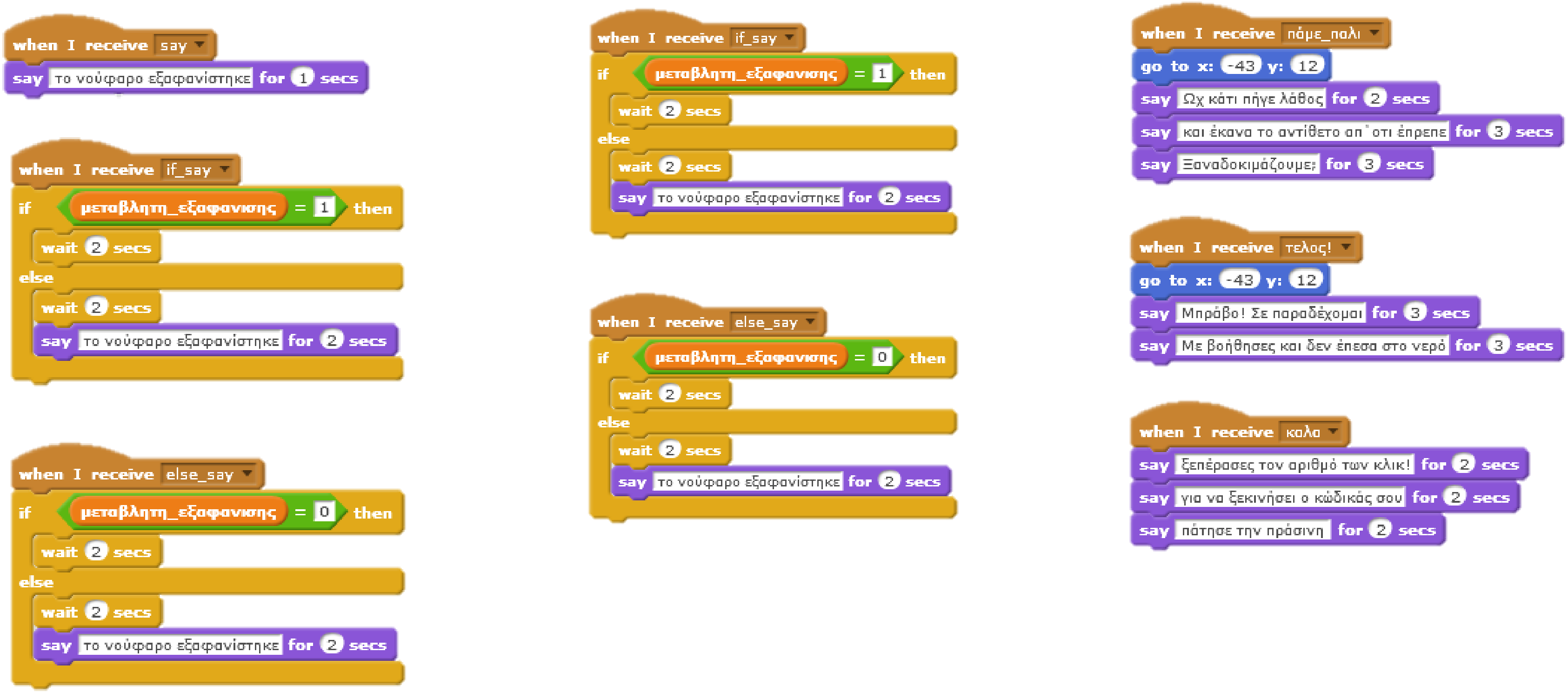
Sprite	
Description	Waterlilies sprites
LO	MoveComplex



D.4 Sprites' Codes of the LO "*If _Else*"

Table D.4.1. Main frog's code

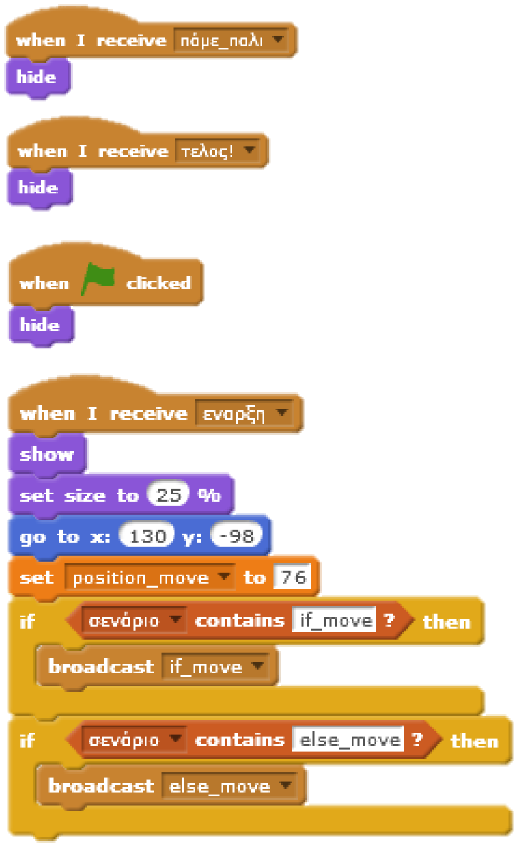
Sprite	
Description	Main character of the LO
LO	If_else




when I receive if_move
if μεταβλητη_εξαφανισης = 1 then
wait 2 secs
else
switch costume to 2
point in direction 0
repeat 4
turn 20 degrees
move 25 steps
switch costume to 1
set y to 9
when I receive else_move
if μεταβλητη_εξαφανισης = 0 then
wait 2 secs
else
switch costume to 2
point in direction 0
repeat 4
turn 20 degrees
move 25 steps
switch costume to 1
set y to 9
when I receive γομα
ΑΡΧΙΚΟΠΟΙΗΣΗ
when I receive εναρξη
ΑΡΧΙΚΟΠΟΙΗΣΗ
when clicked
hide
when I receive πρασινη σημαια
ΑΡΧΙΚΟΠΟΙΗΣΗ
broadcast arrow
repeat length of σενάριο
wait 2 secs
broadcast item εντολές + 1 of σενάριο
change εντολές by 1
wait 2 secs
broadcast σтоп
define ΑΡΧΙΚΟΠΟΙΗΣΗ
switch costume to 1
set size to 20 %
show
point in direction 90
go to x: -42 y: 9
when I receive move
switch costume to 2
point in direction 0
repeat 4
turn 20 degrees
move 25 steps
switch costume to 1
set y to 9
when I receive say
say το νούφαρο εξαφανίστηκε for 1 secs

Table D.4.2 Move's code

Sprite	
Description	Move block
LO	If_Else



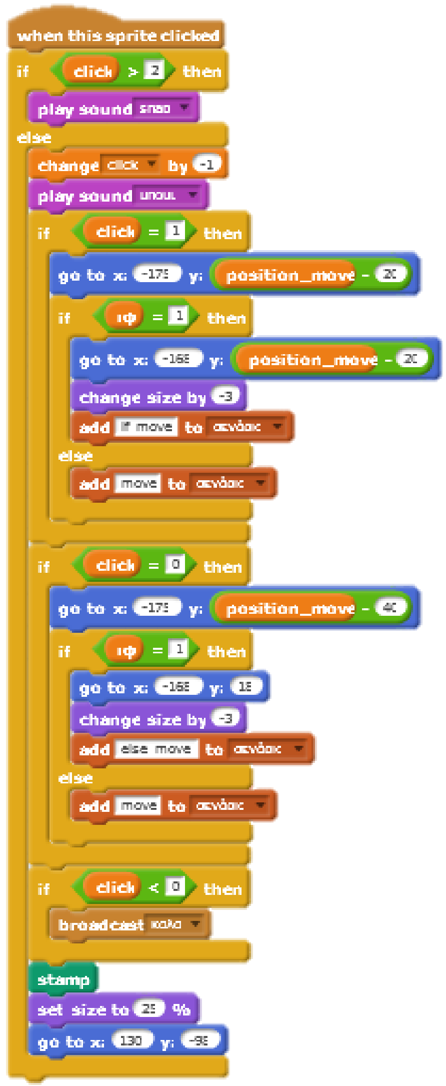


Table D.4.3 Say’s code

Sprite	
Description	Say sprite
LO	If_else

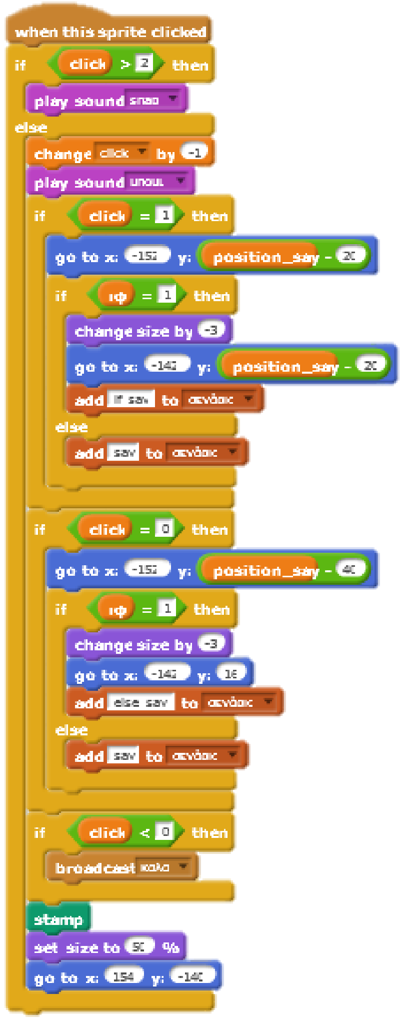


Table D. 4.4. If's code

Sprite	
Description	If block
LO	If_else

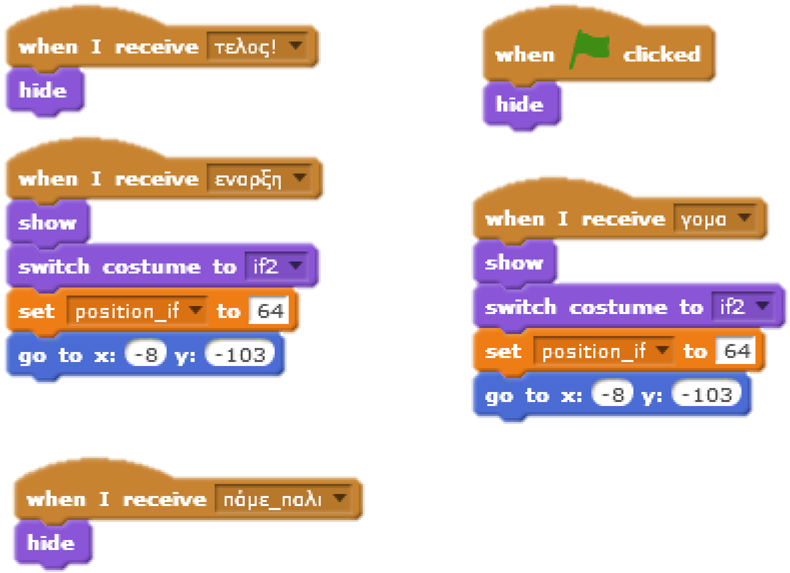


Table D. 4.5 Arrow' code

Sprite	
Description	Arrow
LO	If_else

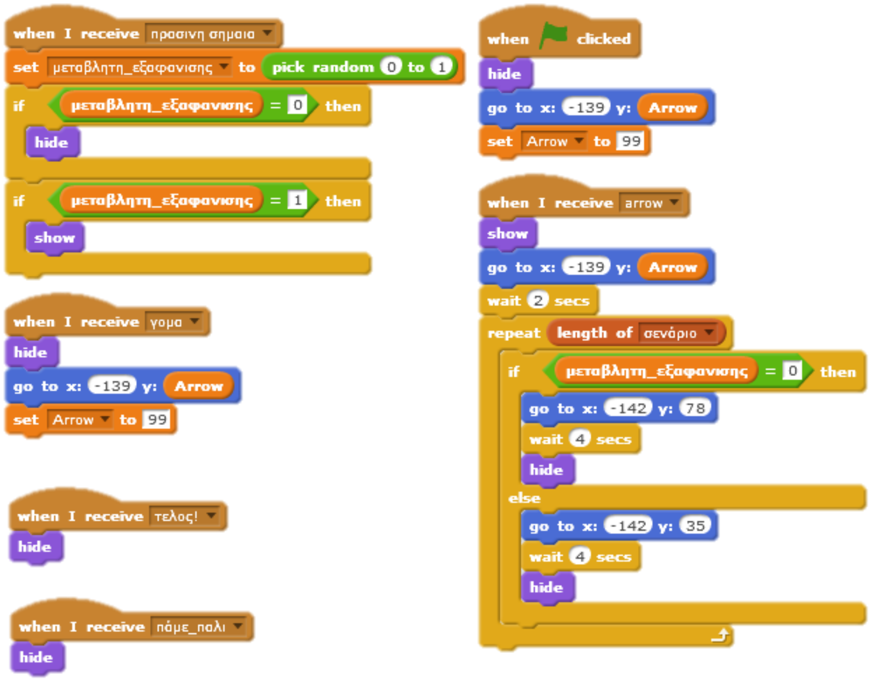


Table D. 4.6 Initial frog's code

Sprite	
Description	Frog of the initial scene of the LO
LO	If_else

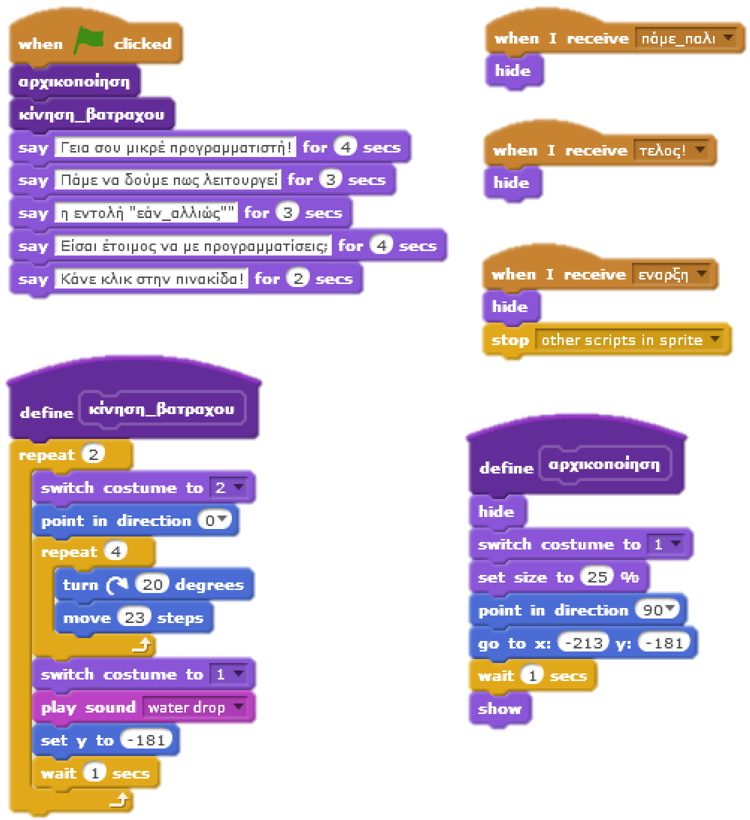


Table D. 4.7 Event' code

Sprite	
Description	Event block
LO	If_else

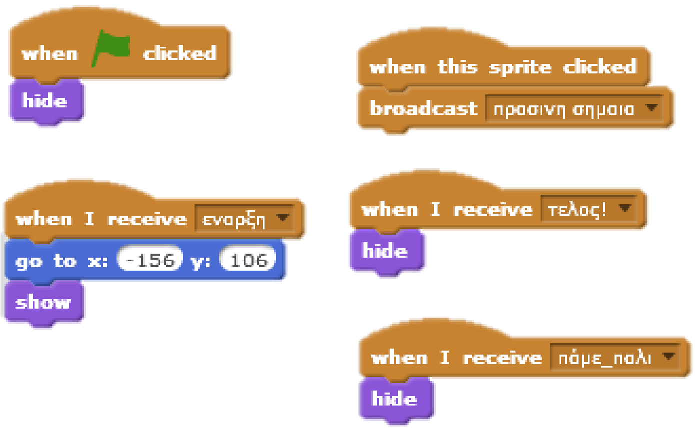


Table D. 4.8 Eraser's code

Sprite	
Description	Eraser sprite
LO	If_else

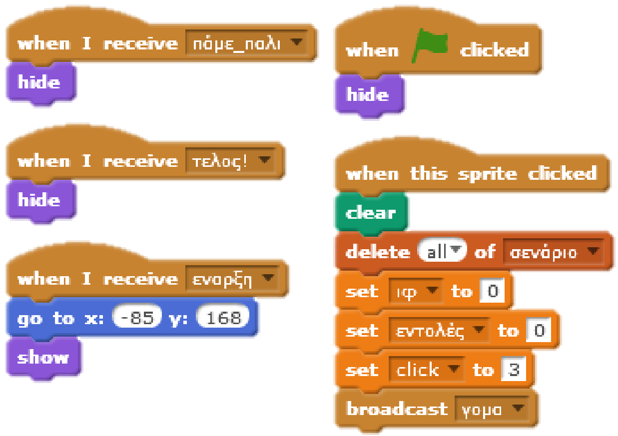
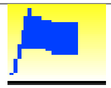


Table D.4.9 Flag's code

Sprite	
Description	Flag sprite
LO	If_Else

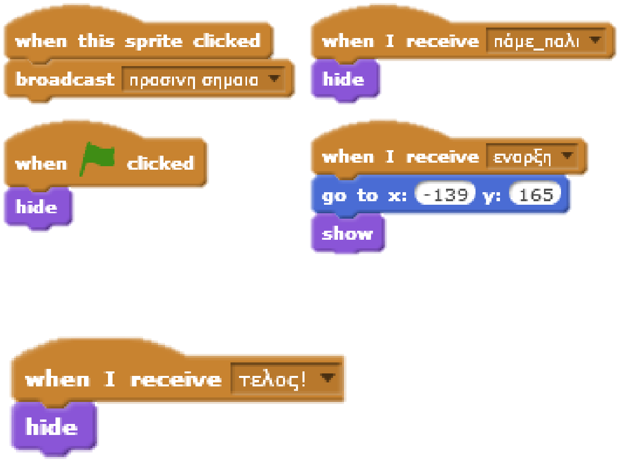


Table D.4.10 Go' code

Sprite	
Description	Go sprite
LO	If_Else

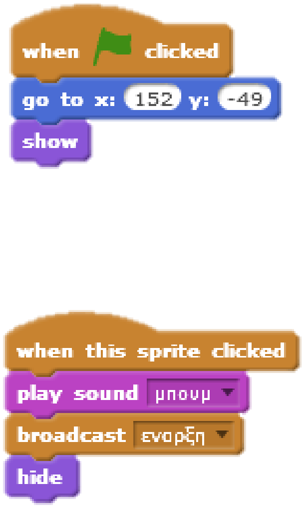
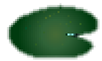
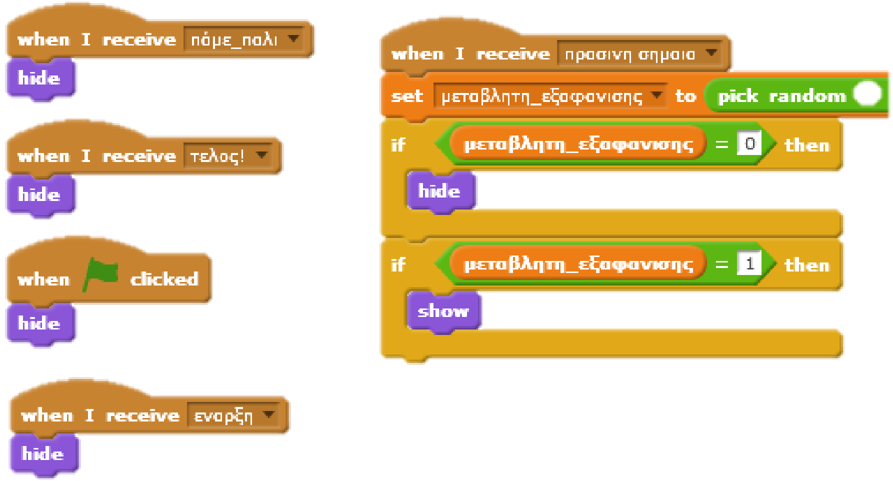


Table D.4.11 Wood’s code

Sprite	
Description	Wood sprite
LO	If_Else

Table D.4.12 Waterlily’s code

Sprite	
Description	Waterlily sprite
LO	If_Else



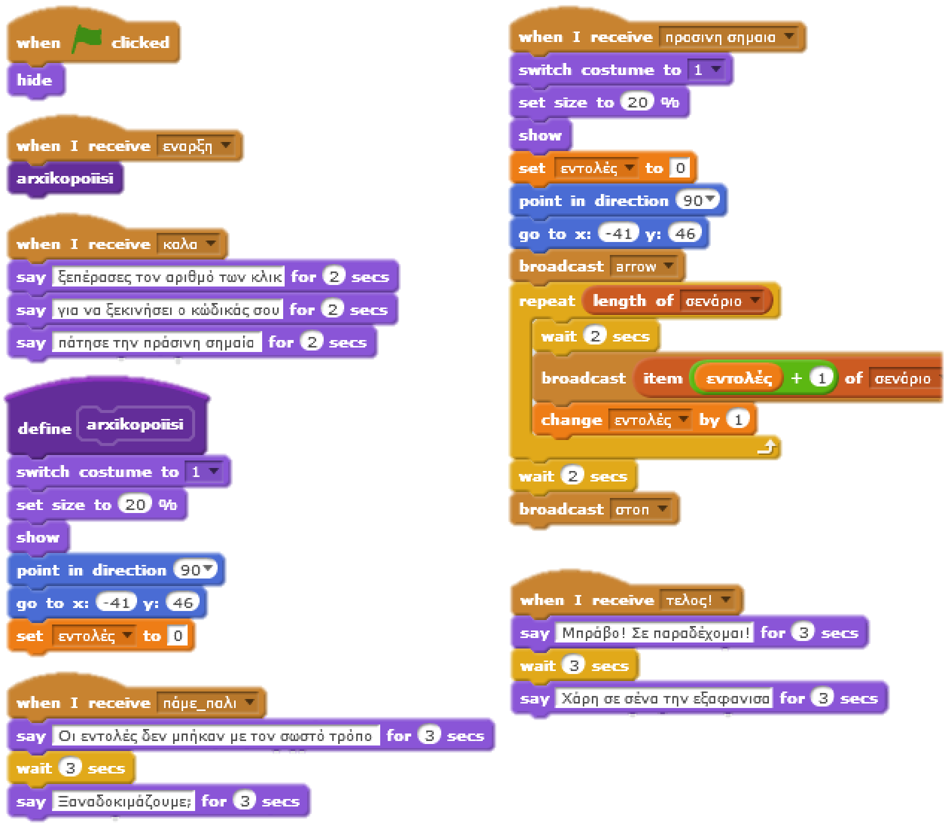
D.5 Code of the LO "Broadcast_When I receive"

Table D.5.1. Main frog's code

Sprite	
Description	Main character of the LO
LO	If_else

Table D.5.2 Initial frog's code

Sprite	
Description	Frog of the initial scene of the LO
LO	If_else



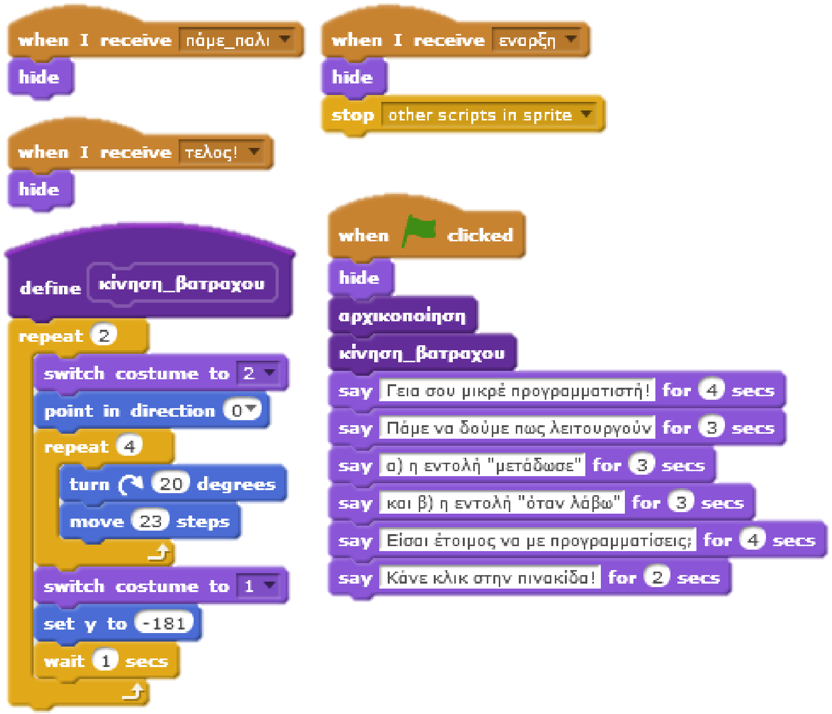


Table D.5.3. Yellow arrow's code

Sprite	
Description	Yellow arrow sprite of the LO
LO	If_else

Table D.5.4 Red Arrow's code

Sprite	
Description	Red arrow sprite of the LO
LO	If_else

Table D.5.5. Broadcast' code

Sprite	
Description	Broadcast block of the LO
LO	If_else

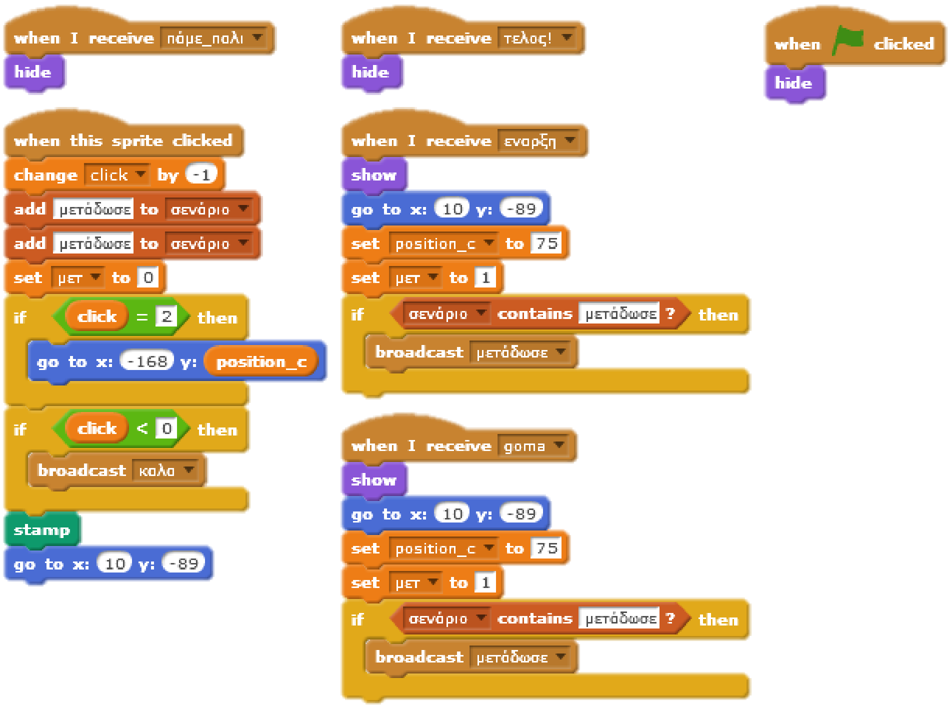


Table D.5.6 "When I receive" code

Sprite	
Description	"When I receive" block of the LO
LO	If_else

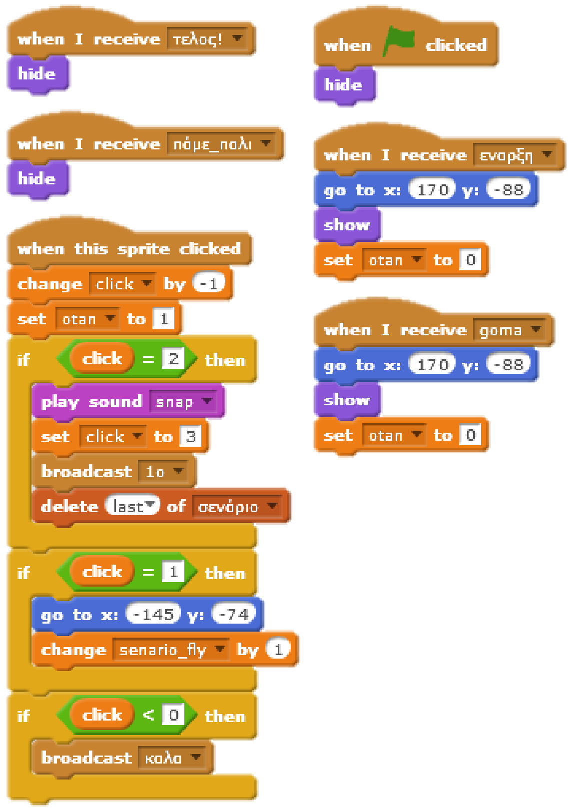
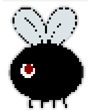


Table D.5.7. Disappear's code

Sprite	
Description	Disappear block of the LO
LO	If_else

Table D.5.8 Fly's code

Sprite	
Description	Fly sprite of the LO
LO	If_else

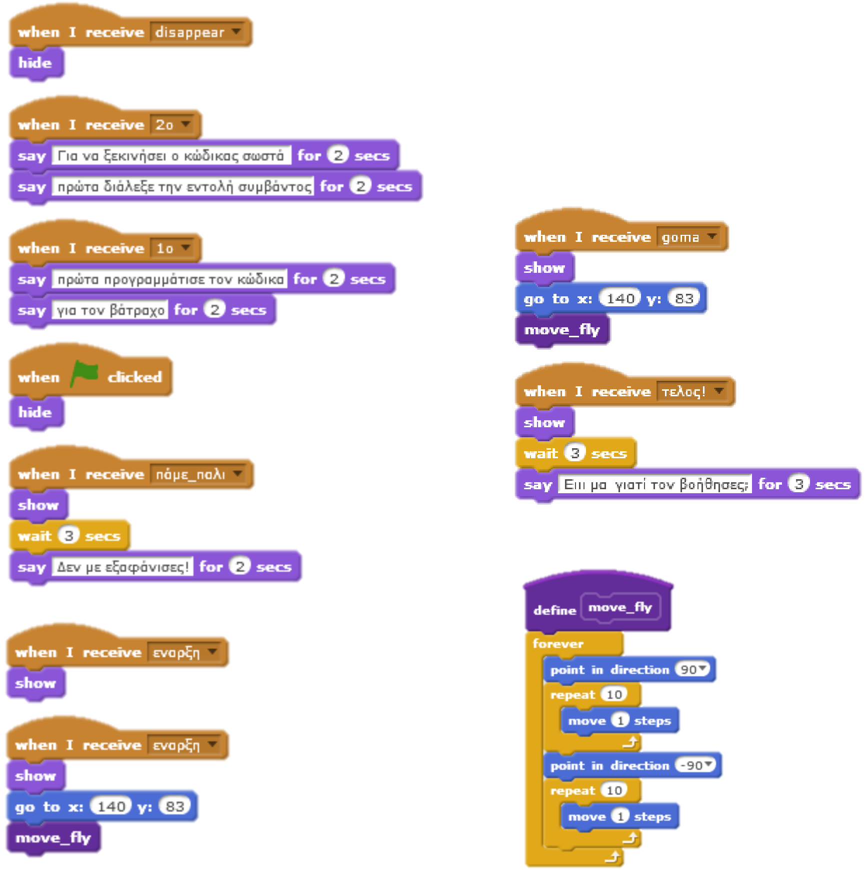


Table D.5.9. Event's code

Sprite	
Description	Event block of the LO
LO	If_else

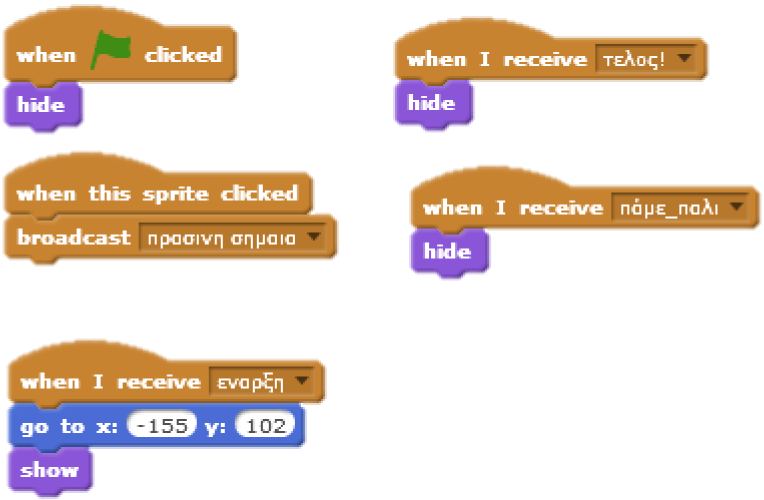
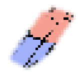


Table D.5.10. Eraser's code

Sprite	
Description	Eraser sprite of the LO
LO	If_else

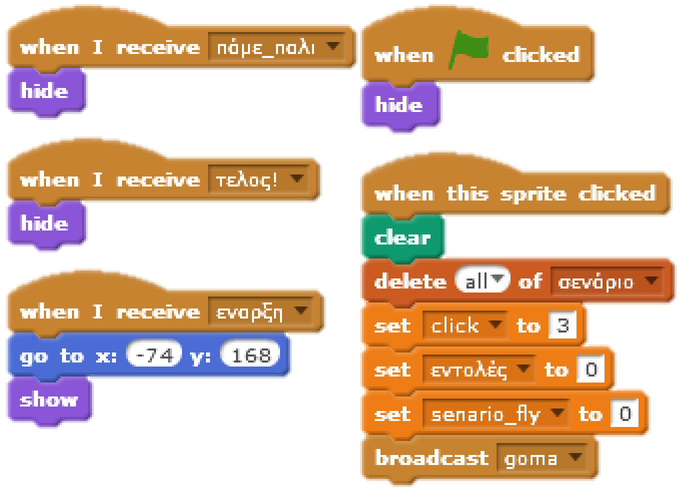
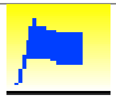


Table D.5.11. Flag's code

Sprite	
Description	Flag sprite of the LO
LO	If_else

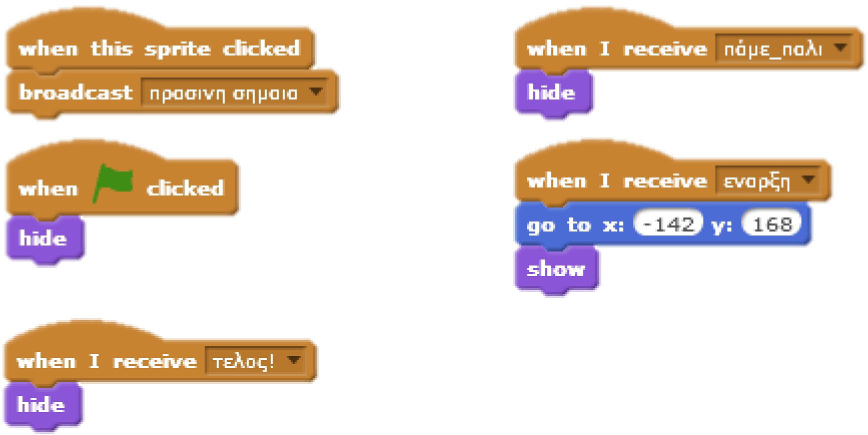


Table D.5.12. Go's code

Sprite	
Description	Go sprite of the LO
LO	If_else