

## A Serious Game for Learning C Programming Language Concepts Using Solo Taxonomy

<https://doi.org/10.3991/ijet.v12i03.6476>

Alaeeddine Yassine

Universite Sidi Mohamed Ben Abdellah, Fez, Morocco  
alaeyassine@gmail.com

Driss Chenouni

Universite Sidi Mohamed Ben Abdellah, Fez, Morocco  
d\_chenouni@yahoo.fr

Mohammed Berrada

Universite Sidi Mohamed Ben Abdellah, Fez, Morocco  
mohammed.berrada@gmail.com

Ahmed Tahiri

Universite Sidi Mohamed Ben Abdellah, Fez, Morocco  
tahiri\_ahmed@hotmail.com

**Abstract**—this paper conducts a study to identify pedagogical approaches and gameplay techniques involved in the development of serious games for teaching scientific courses in general especially programming languages. The concept of serious games is increasingly popular and is considered as an innovative teaching practice since it is based on information and communication technology and gamification to foster learning. To this end, a serious game "Perobo" will be introduced and discussed. It is based on a set of gameplay techniques and pedagogical approaches used for teaching pointers, considered as a difficult concept in C programming language. The game is also based on a taxonomy design to define the learning levels.

**Keywords**—Serious games, Programming, Pedagogical Approach, Gameplay, Drag & Drop, Point & Click, Solo Taxonomy.

### 1 Introduction

In recent years, serious games have become increasingly popular. Many researches are dedicated to the potential of serious gaming to foster learning and motivation. Games are known for their entertaining side, they make the players enjoy challenges: although they present difficult tasks hard to accomplish and even if it takes the players many hours or days -losing track of time- chasing objectives, they stay motivated and engaged. Furthermore, players could repeat the game many times without finding it boring. Games need motivation, willingness and hours of practice and so do learning.

In contrast, learning is not as challenging as videogames, especially when it comes to learning programming languages. Like playing games, programming is a discipline that requires many skills such as analysis, critical thinking, and problem solving. It also requires a lot of practice so that the programmer gets familiar with new instructions. These are the prerequisites to understand a programming language. However, writing codes are not always challenging for students even if they are trying to solve a problem through a plan of instructions. In fact, students taking a programming course are struggling with high-level abstractions and complex instructions, which make the code difficult to understand and master.

One solution to this problem could be to use a new pedagogy based on games for teaching programming. The purpose is to make learning programming fun and engaging. This new practice has the potential to encourage learners discover abstract concepts and facilitate their understanding through a less frustrating process. To this end, we developed "Perobo" which is a serious game to teach most difficult concept in C programming language, which are "pointers".

The remainder of this article is organized as follow: first, gameplay techniques and pedagogical approaches will be presented. Then it will be followed by an overview of some serious games used to learn various subjects. After that, we will introduce the Solo taxonomy design used to distribute the levels of a learning topic and how it affects the learning levels of our serious game Perobo that will be introduced right after. Through the levels of the game, we will evaluate the potential to teach student pointers.

## **2 Learning a programming language**

When learning a programming language, students develop their problem solving and critical thinking skills. It also teaches them how to understand a problem and its boundaries, how to carry out a plan, and organize actions to solve a problem through a set of instructions. It could help them look back and reflect on their solution. Another advantage of learning how to code is to show student how to persist at a task by checking errors and correcting them so as to get the program run. Still, there are many open questions: why most of students dislike the subject of programming language? Why programming is considered among the difficult tasks? Why does it take motivation and persistence to be expert on a programming language? How could one best learn a programming language?

The following paragraph discusses difficulties of classical learning approach as well as the use of serious games as a new approach to teach and learn a programming language.

### **2.1 Difficulties of classical learning approach**

Learning the basics of programming is challenging, especially for novice learners. Gomes and Mendes [1] posit that among the reasons behind the difficulty of programming is that it requires a high level of abstraction and many hours of study and

practice. Sid Stamm [2] pointed out that novice learners think that programming without practice is theoretical and not motivating.

Learning a programming language is about understanding syntax and the functioning of code to solve a problem. Sandra Milena Merchán Rubiano postulates that novice learners focus more on grammar of the language rather than the computational semantic behind the syntax [3]. However, what helps programmers solve problems is to design logical steps of instructions that the algorithm follows to find solution based on logic and semantic rather than correct syntax.

Another major difficulty is faced by teachers, when using an ordinary way to teach programming language students seem not to get the logic behind the code since it requires imagination and prerequisite in the field of mathematics and logic, students get discouraged when they get syntax errors that they couldn't solve or understand. Furthermore, most students are frightened out of learning a programming language because of the perception they already have. Most of them think programming is difficult and requires a complex cognitive effort.

Apart from the learning process, learners use computer especially and IDE - Integrated Development Environment- to create programs. The computer is also used to search for solutions or to share issues with the online community. Enough reasons to consider using this tool to teach programming language rather than using a black-board. Using a computer has the potential to foster student motivation since nowadays students are digital natives. It could also help the teacher create a simulation of program behavior so as to better understand its functioning. To overcome the problems related to teaching and learning programming, some researchers propose practical solutions. For instance, a group of [2] researchers emphasizes the importance of encouraging the learner to work directly on realistic and meaningful tasks related to a context.

To sum up, programming courses should be redesigned so to conform the ongoing challenges and fit student's characteristics. Teaching a programming language should be done through gamified challenges to help students learn basic principles and stay engaged throughout the learning process.

## **2.2 Using Serious Games to Learn a Programming Language**

Serious games are a new approach to rethink the teaching learning process using advances in new communication and information technology. The term 'serious game' refers to games used for learning, in addition to their entertainment purpose, serious games hold a pedagogical vocation and address learning outcomes.

Using games students could inhabit objects and could feel emotions. It activates their interest since to be a good player requires a student to have control over a situation. Students learn how to make strategies to solve a problem or to accomplish a task, and this is what is needed in order to make a program: good strategies translated into working instructions.

Another benefit of using serious games is that it enhances self-pacing because student might learn at their own pace, fail, go back and revisit the concept.

According to Clark Abt [4], 'Serious' games are not intended only for fun since they also call for thoughtful educational goals, and it's the entertaining side that attracts the attention of students and engage them in the learning process.

Two parameters are to be considered in serious games: (1) Entertainment side, represented through gameplay or scenarios, and (2) educational side that refers to content implicitly transmitted when playing. The use of these dimensions allows us to redesign teaching programming, by addressing four issues:

**Presenting learning concepts in a thoughtful and concrete way:** The study conducted by Essi Lahtinen & all[5]pointed out that learners find abstract concepts such as pointers and memory management hard to understand, which sometimes require a good understanding of the running of the program itself and the computer architecture. Another study conducted by Milne and Rowe [6]shows similar results: the most difficult concepts to understand in C programming language are pointers, recursion and data structures.

An example of using serious game for learning is 'world of variables' [7], which was developed to teach the concept of variables of C programming language. This game is based on the story of an alien who wants to leave its planet and seeks refuge elsewhere. The main goal of the game is to understand the features and running of the variable to novice learners.

Throughout the game, each level holds a mission, a game play and addresses a set of learning outcomes, the learner discovers by playing how to declare a variable, distinguishes the types of variables, initializes and stores a value in a variable or implicitly assigning a value to a variable.

This approach helps players best learn abstract concepts through concrete situations, increasing their motivation and interest for programming.

**Fostering student's motivation and interest:** Programming is often limited to repetitive and complex operations. To become an expert in a programming language one needs a lot of practice and a great motivation. This idea is supported by Judith Bishop & al [8]who claims that learner's motivation could be fostered trough integration of an entertaining aspect to programming by using audiovisual elements, social networks and mobile learning. They came up with a serious game Code Hunt [8].The player has to write a code in Java or C # to progress through the levels of the game. More specifically, the game asks players to write a code in order to move or to make an action. So to make the game more challenging, all activities are done in a competition setting where multiple players can compare their results.

**Using tools learners are familiar with:** The emergence of new technologies has enabled the education sector to renew and expand educational content and adapt it to students who daily use computers, smartphones and tablets.

According to Houda Mouaheb et al [9], a serious game is an information and communication tool. Resulting from the video game industry, the serious game aims to teach, inform, educate and raise awareness.

**Providing learners with a realistic setting with meaningful and motivating goals:** In a standard education setting, students often get confused without concrete examples. They don't get the purpose of learning. Everything can be understood by being mentally constructed/ students need context for learning.

So to make sense about learning, serious games offer one or more realistic situations in which the player has clear and significant goals. It's a tool for students to go back and revisit the information. For example, 'Darfur is dying'[10] is a serious game of awareness where the learner plays the role of a Darfuri villager who must ensure the survival of his village regularly attacked by militias. The player is therefore engaged in a humanitarian mission and feels motivated to succeed in the game.

Thus, the motivation of the learner is assured by the integrating of a realistic scenario with well-defined educational objectives, allowing him to perceive the interest of his learning.

To sum up, serious games have the technical and pedagogical prerequisites to overcome the problems of classical learning. It remains to define three parameters by which the serious game transmits knowledge:

- The gameplay techniques that defines the interaction of the learner with the serious game.
- The way of learning through appropriate pedagogy.
- Effectiveness principles that verify to what degree serious games created is a good way of learning.

Our choice will follow a methodology that aims to address the problems cited below:

The gameplay techniques will have to:

- Solve the problem of abstract concepts such as pointers, by using one or more playing techniques that exploit concrete elements (characters and objects)
- Put the player in a game environment so he can experience a new way of learning.
- Exploit new technologies familiar to learners and players.

Pedagogical approaches are supposed to:

- Make the practice of programming more entertaining by promoting the fun and spirit of challenge.
- Putting the learner into a game atmosphere with clear and significant educational goals.

Furthermore, the efficiency criteria will aim to ensure that the learner is at the heart of the design of the serious game.

### **3 Gameplay techniques to best learn programming**

This chapter aims to identify gameplay techniques and pedagogical approaches to best teach and learn C programming language. Gameplay stands for the way in which players interact with a game. Many factors should be taken into consideration when choosing a gameplay namely: the player's age, the device, the game strategies, etc. In this section, we highlight the most common gameplay techniques we found in the

literature, each technique is described and illustrated through an example of serious game.

### 3.1 Drag&Drop

Drag & Drop is a gameplay technique adapted to serious games. The interest of the drag & drop lies in the fact that it is suitable for beginner learners. Using a mouse or just the finger on a touch screen, the player can drag and drop game elements to generate a result. For example, the player must drag pieces of code and place them correctly in a computer program to create an interaction with the game.

For example, 'World of variables' [7] depicted in Fig 1, is a game which uses drag & drop technique in order to teach the declaration of a variable. The player must correctly combine objects (bottles, liquids, labels ...) by dragging and dropping them in shelves. The learner must put the liquid and the right label on the bottle.

The benefit of using this technique is to provide the learner with a realistic environment. This environment contains stimulus attracting the learner. Through interaction with objects and concretization of what might look like a variable. The learner constructs his/her own knowledge and abstract concept that become easier to understand. For instance, in "World of variables" The drag & drop implicitly reproduces the behavior of a variable when getting a value, this allows the learner to understand that a variable has a value (liquid inside the bottle) and a name (sticker on the bottle).



Fig. 1. Level 1 of the game 'World of variables'[7]

### 3.2 Point&Click

Point & click is a technique based on interactions with elements of the game's environment. Suitable for beginners, it encourages learners to use their senses (sight and hearing) to interact and learn. The exploration of interactive elements of the game which awaken player's curiosity and leads him to click on various objects embodied into the environment figuring out how to solve a problem or a mental challenge.

According to Marine BENEZECH [4], the choice of point and click allows to highlight reflection and exploration instead of confronting the player to games involving speed. In one hand, games which involve speed are known for adrenaline rush, where attention is absorbed by the activity of the game rather than the content itself. On the other hand, serious games based on point & click, are known for suspense and are arising from learner's curiosity. This moves the learner from the hope of reaching a goal to a hope of figuring things out and then learns.

In the game 'A la poursuite de Routix'[16] as illustrated in Fig2, the player must explore images representing urban Traffic and detect seven dangerous behaviors. He should then click on them to complete the mission.



Fig. 2. The header image of online-journals.org

## 4 Pedagogical approach to enhance learning

In this section, we present two educational approaches that we used to develop our serious game. For instance, game-based learning approach and constructivism.

#### 4.1 Game-based learning approach

Gamification is an educational approach which relies on activity played within the game [1]. It exploits the visual, sound and objects of the game to create the effect of immersion. Game-based learning theory could be defined as combination of games with educational content, it encompasses learning outcomes and a content to deliver[11].

Game-based learning is a result of effective integration of both pedagogy and educational approaches into game industry. Wong Seng Yue defines Game-based learning approach as an educational approach which uses technology to support the teaching learning process. It is about getting student engaged while playing and learning at the same time[12]. This approach intends to put the learner in a realistic and challenging situation. For example, as shown in Fig3, the serious game ‘Le roi et la salamandre’[13], the learner plays the character of King François first and performs tasks inspired from the life of the king. Thus, learning the events of that period of history is through the playful immersion in the life of King François first.

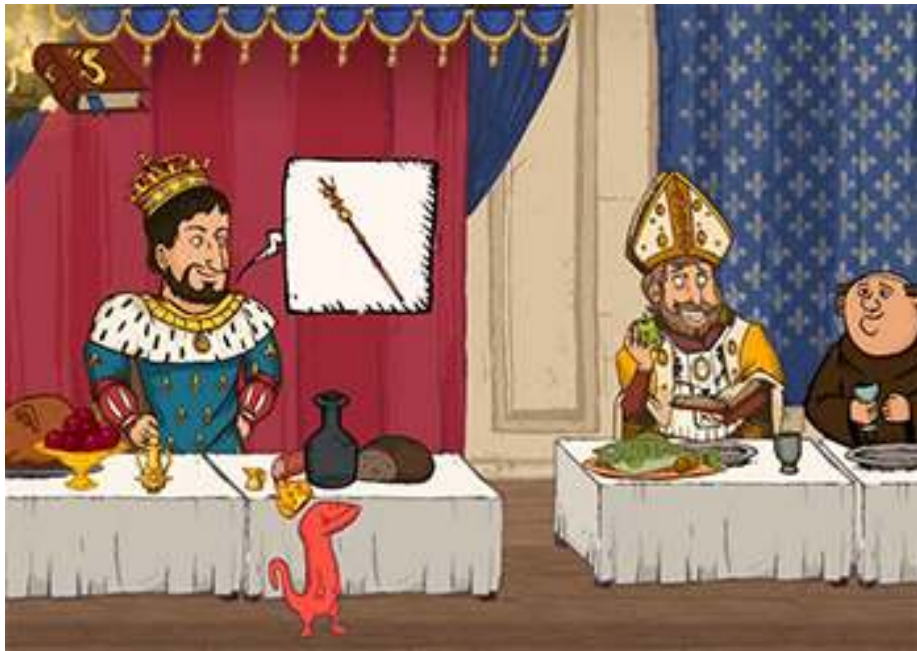


Fig. 3. In the game ‘le roi et la salamandre’, you embody the king François first.[13]

#### 4.2 Constructivism

Constructivism owes its legacy to Jean Piaget who postulates that knowledge is gained through experience. Piaget’s idea hinged on learning as being an active construction of knowledge. It is about a dynamic process of collecting information and constantly organizing it.



According to Jean Piaget learning theory[14], to leverage learning, the learner needs to create its own knowledge through discursive interactions with the environment. In a constructivist approach knowledge goes in line with experience to create deeper understanding.

## **5 Perobo serious game for introductory C programming language**

In order to foster student learning and motivate them while learning C programming language. We have created a prototype of the serious game 'PEROBO' using the software Stencyl 2.0[15].

The objectives of PEROBO Serious game are:

- Remind the learner / player of basic notions of variables in the C language.
- Remember definitions related to pointers as variables in the C language.
- Understand types of pointers and their functioning.
- Understand Pointers and Memory Allocation.
- Learn the syntax of a pointer in a computer program.





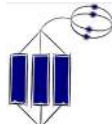
The game addresses a series of learning outcomes starting with simple definitions to complex programs. At this end, Solo Taxonomy is used to design levels of the game. The next section presents Solo Taxonomy and its benefits followed by a demonstration of its use for Perobo Serious Game.

### **5.1 Instructional Design using Solo taxonomy**

Solo taxonomy stands for Structure of Observed Learning Outcome. It was first introduced by Biggs and serves as model of learning which describes level of understanding[17]. It is based upon a theory about teaching and learning. Solo Taxonomy is used by teachers to design the teaching process based on students' level of understanding.

Using taxonomy, educators can describe, in human language, progression of learning activities, and developers can focus on game development to reach described learning outcomes. Furthermore, learning levels related to SOLO taxonomy can be assimilated to levels of the game. Solo taxonomy is organized as levels of knowledge acquisition each level has a goal. The following table (Table1) gives an overview of Solo Taxonomy.

**Table 1.** Solo Taxonomy For Knowledge Level Acquisition






Level of knowledge acquisition		Objective
	<i>Pre-structural</i>	I don't know anything about the subject
	<i>Uni-structural</i>	I know one aspect of the subject.
	<i>Multi-structural</i>	I know few aspects of the subject but I am not sure when or why to use them
	<i>Relational</i>	I can build relations between aspects and I know when and why I should use them
	<i>Extended abstract</i>	I can teach the subject to others and I can use what I learned in other contexts

Benefits of using Solo Taxonomy as design model:

- It facilitates serious games design: the teacher describes in a standard way content: in a step-by-step process.
- It starts with level 0, which implies that the learner knows nothing about pointers. The advantage is to encourage learners to start from scratch without frustration.
- It proposes a progressive organization of knowledge, moving students from novice to expert status.
- It enhances differentiation since it is possible to assess student's level at the beginning of the game and place it directly into the level that suits him/her the most.
- Provides a means for describing learning outcomes. It can be used as a scale for assessing learner at each level of the game by assigning scores.

In table 2, we give a case study of using solo taxonomy to design the serious game 'Perobo'. For each level of the game the learner should progress from knowing nothing about the subject (pre-structural level of the taxonomy) to, at least, competent.

**Table 2.** Solo Taxonomy For Knowledge Level Acquisition

Perobo Levels	 PRE- STURCTUR EL	 UNI- STRUCTUREL	 MULTI- STURUCTUREL	 RELATIONN EL	 ABSTRAIT ÉTENDU
<b>Score</b>	Novice	Advanced beginner	Competent	Proficient	Expert
<b>Goal</b>	Few or no background knowledge	Understand and deal with one aspect of a subject	Understand different aspects of a subject, without connecting them	Build a relation between different aspect of a subject	Innovate and create new things
<b>Level 1:</b> definition of the pointer in C programming language	Student do not know what pointer is and what is used for	Understand the definition of a pointer.	Make the difference between a variable, a memory address and a pointer.		
<b>Level 2:</b> Pointer declaration and use	Learn that there are several types of pointers.	Understand how a pointer interact with computer memory based on an address	Store and retrieve data using a memory address		
<b>Level 3:</b> Pointers and tables	Understand that a table can be formed of pointers.	Retrieve using pointers the first element of a table	Read table elements using pointer and loops	Use pointers and tables to store "String" data type	
<b>Level 4</b> : concept of allocating and freeing memory in c using a pointer	Understand the function of memory allocation in C language	- Memory Allocation. - Freeing memory.	Calculate the size to allocate.	Dynamic memory allocation for tables	

## 5.2 Presentation of serious game ‘Perobo’

To motivate learners, we thought of a story that could have an effect on learner’s immersion in the game. We put the focus on the attractiveness of the main character who is a student who should go through a series of challenges by interacting with game environment and constructing his/her own knowledge of the pointers functioning. The scenario is designed upon immersion as key feature as depicted in Fig4. The learner, being the main character has a mission to accomplish. This should help him/her feel as the one in control of the situation.



Fig. 4. First scene of PEROBO Serious Game

The game proposes an imaginative experience that is normally impossible in real life: The main storytelling of Perobo Serious Game is inspired from the functioning of computer memory while using pointers. The purpose is to enable learners use their abstract thinking to imagine how pointers work.

Perobo Serious game relies on constructivism as educational approach. Constructivism postulates that learning is a dynamic process that requires collecting information and constantly constructing and organizing it overtime[18]. The idea is to use objects to construct knowledge. For instance, we allow the player to interact with several objects to figure out how to advance in the gameplay. As depicted in Fig 5, learners are provided with a mission to accomplish and in the same time, they are assembling knowledge trough exploration of the game environment.

**Chapter 1 of 'Perobo': Recall of prior knowledge:** In early levels of the game, and before diving in learning pointers, a tutorial is presented; it is a step-by-step tutorial that provides players with instructions and learning goals.



Fig. 5. Level 1 pf chapter 1, PEROBO Serious Game

From an educational perspective, we use the constructivist approach by encouraging the learner to complete the definitions he has to learn. This activity involves the learner's critical thinking skills. S/he has to consider the options and choose the correct answer to constitute a definition s/he can easily remember. Furthermore, and in order to challenge the learner, the number of attempts is limited to three.

The following figure (Fig 6) illustrates level 2 of chapter 1, using “drag & drop”, the student must complete a definition by dragging pieces of paper containing answers towards the definition written on the blackboard.

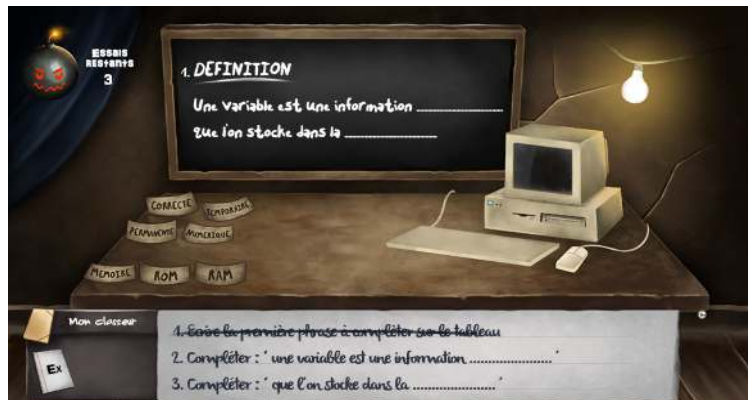


Fig. 6. Level 2 of chapter 1, PEROBO Serious Game

After finishing the first chapter, a feedback is shown to the learner in order to help him/her remember the definition s/he has completed. The next figure (Fig7) describes an example of feedback.



Fig. 7. PEROBO chapter 1 feedback Level 2 of chapter 1, PEROBO Serious Game

To sum up, the chapter 1 aims to prepare students to learn pointers by assessing some prerequisites related to variables in C programming language.

**Chapter 2 of Perobo: Pointer declaration and use:** Before diving in the second chapter of Perobo serious game, learning outcomes and instructions are displayed to learners. As illustrated in the following figure (Fig8), the mission is to understand what type of pointer one should declare in the program. To do so, learner is invited to play this chapter, and returns back to give an answer.

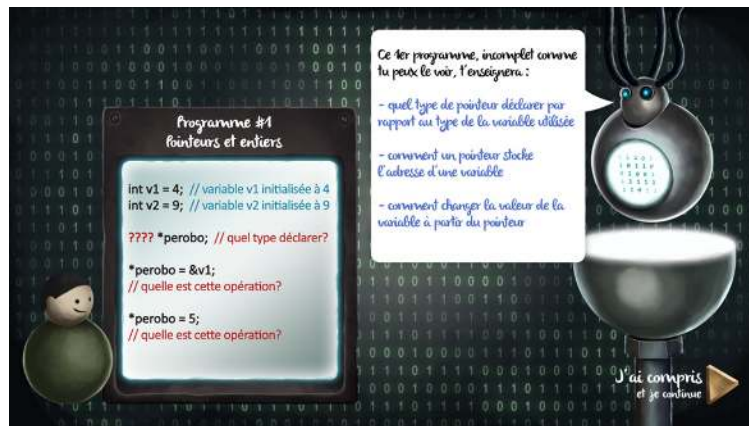


Fig. 8. PEROBO chapter 2 –instructions for learners

Level 1 of chapter 2: in this level, the player must guess what type of pointer to choose. To do so, he must drag and drop a battery on a robot that represents the pointer. Four robots are displayed on the screen and each one of them has a type (char, long, integer and real). So the learner is required to drag the battery towards the robot which type is integer to turn it on. The following figure (Fig9) gives an overview of level 1 of chapter 2.



Fig. 9. Level 1 of chapter 2, PEROBO serious game

The Integer type robot turns on when battery is dropped towards him. At this stage, learner should understand the first line of the program. For instance, that pointer is a

variable; it should be the same type as the variable the robot (pointer) will be pointing to.

The next step, as depicted in Fig10, aims to go deeper and understand that the pointer is a variable, but it is a kind of variable that holds a memory address. The selected robot will move to RAM (Random Access Memory) where two variables (represented by yellow robots V1 and V2) are stored. It is required to associate the pointer with the right address based on the value of the variable. Using a card, the learner should choose between the two addresses 10 or 11 to make the robot points to the variable v1 and then to pass the level. The following figure (Fig10) depicts the level 2 of chapter2. The card contains the operator “&”, in C programming language, it refers to the address of a variable. This means, to make a grey robot (pointer) points to a yellow robot (variable), learner should use the card containing the operator ‘&’.



Fig. 10. Level 2 of chapter 2, PEROBO serious game

## 6 Discussion

In order to master pointers in introductory C programming language, students should acquire new knowledge. However, the process of learning pointers requires abstract thinking. As pointed out by James Paul Gee [19], a student understands a word only when they can associate it with an image, action or experience. Perobo Serious Game seeks to challenge students and helps them become abstract thinkers by understanding, through an entertaining scenario, the running of pointers in their real context in low risk way. They learn through the game how do pointers use computer memory to store and retrieve values? What is the value added of using pointers and not variables?

Perobo serious game foster abstract thinking by using “drag and drop” and “point and click” as gameplay techniques. The game is about exploring the environment, making good strategies and translating them into working instructions. These are perquisites of good programming. To make exploration fun and attractive, we make the game environment fully responsive to learner’s actions. Learners are encouraged



to observe and reflect on things. Player's actions trigger sound effects or events to engage student's through emotional reactions. For example when entering an attic room, the player hears the sound of an old creaking door, then should figure out how to turn on the light in response to sudden darkness. The purpose is to make the game enjoyable through realistic motions such as turning an old computer on, cutting pieces of paper, moving items, writing on a blackboard, talking to characters, etc.

In addition to sound effects, lightning and motions and in order to make the gameplay more challenging, we added a specific restriction: limiting the number of items to use. The player could find as many objects as possible while exploring the environment but can only keep those useful to progress in the game. He/she has limited attempts to figure out the right tools (items) to use in order to make progress in the game. Some hints are shown from time to time to provide students with assistance and keep them on the right track. Perobo also promotes the feeling of accomplishment by rewarding learner's curiosity. For instance, the game give the player positive feedback and some ordinary items on screen can deliver learning content.

Teachers of C programming language, who participated in the instructional design step, state that is important for student to start with a wide-angle view of what might look like pointers. The game provides students with a general view of the subject to be taught so to be aware of the context and importance of pointer it also gives them a concretization of what might look a pointer. Furthermore, Perobo Serious Game makes the process of teaching and learning pointers less frustrating by enabling students to transfer what is learned from one context to another: from the game to classroom settings. For instance, Perobo is used out of the classroom, students are asked to play the game so to get familiar with the basic idea behind pointers and then come to classroom for deep learning of pointers. From teachers' points of view, it is worth experiencing a serious game as an introduction of teaching pointers. Perobo Serious Game is mainly used to persuade learners to change the way they think of pointers by being curious and exploring how pointers work. The game should help them overcome the frustration through an enjoyable journey. It would not replace the traditional course but should prepare students to learn pointers and get them interested in the subject.

Perobo presents information in a way that enhances retention and transfer of facts. Using solo taxonomy helped us in the instructional design step. The concept of pointers is broken down into small chunks, and presented to learners as a series of challenges sequenced in an increasing complexity. By including, at each level of the game, small concept to master. Learners are surprised, at the end of the game, of how much complexity they can deal with. After learners become familiar with each concept, we ask them to complete a quiz to assess their level on that particular concept. S/he could repeat the test as many times as needed but with different questions each time.

At this stage, Perobo Serious game is under testing by students supervised by a tutor who is a teacher of the subject matter. In order to improve the game and capture data about enjoyability, we propose to players a scale to rate the effectiveness of gameplay.



## 7 Conclusion

In this paper, we have discussed the potential of serious game to teach C programming language. We argued that serious games need an effective integration of both pedagogy and entertainment which offers great opportunities for strengthening motivation among learners. We developed “Perobo”: a prototype of serious game to teach pointers as a difficult concept of C programming language. The purpose is to make learning pointers more attractive than conventional teaching and learning method. As a perspective we need to conduct a usability evaluation of the prototype as proof of concept. Still, we need a mechanism to analyze data set generated by Perobo Serious game while learners are playing to provide them with immediate feedback, or recommend some actions performed by their peers to encourage them take a decision or advance in the game.

## 8 References

- [1] A. Gomes and A. J. Mendes, “An Environment to Improve Programming Education,” in Proceedings of the 2007 International Conference on Computer Systems and Technologies, New York, NY, USA, 2007, p. 88:1–88:6. <https://doi.org/10.1145/1330598.1330691>
- [2] M. Muratet, P. Torquet, F. Viallet, and J.-P. Jessel, “Évaluation d’un jeu sérieux pour l’apprentissage de la programmation.” *Rev. Intell. Artif.*, vol. 25, no. 2, pp. 175–202, 2011.
- [3] S. M. M. Rubiano, O. López-Cruz, and E. G. Soto, “Teaching computer programming: Practices, difficulties and opportunities,” in IEEE Frontiers in Education Conference (FIE), 2015. 32614 2015, 2015, pp. 1–9. <https://doi.org/10.1109/fie.2015.7344184>
- [4] C. C. Abt, *Serious Games*. University Press of America, 1987.
- [5] E. Lahtinen, K. Ala-Mutka, and H.-M. Järvinen, “A Study of the Difficulties of Novice Programmers,” in Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education, New York, NY, USA, 2005, pp. 14–18. <https://doi.org/10.1145/1067445.1067453>
- [6] I. Milne and G. Rowe, “Difficulties in Learning and Teaching Programming—Views of Students and Tutors,” *Educ. Inf. Technol.*, vol. 7, no. 1, pp. 55–66. <https://doi.org/10.1023/A:1015362608943>
- [7] M. Zapašek and J. Rugelj, “Learning programming with serious games,” *EAI Endorsed Trans. Game-Based Learn.*, vol. 1, no. 1, p. e6, Mar. 2013. <https://doi.org/10.4108/trans.gbl.01-06.2013.e6>
- [8] J. Bishop, R. N. Horspool, T. Xie, N. Tillmann, and J. de Halleux, “Code Hunt: Experience with coding contests at scale,” *Proc ICSE JSEET*, 2015.
- [9] “The Serious Game: What Educational Benefits?” [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S187704281202201X>. [Accessed: 06-Sep-2016].
- [10] “Darfur Is Dying - Play mtvU’s Darfur refugee game for change.” [Online]. Available: <http://www.darfurisdying.com/>. [Accessed: 06-Sep-2016].
- [11] M. Papastergiou, “Digital Game-Based Learning in high school Computer Science education: Impact on educational effectiveness and student motivation,” *Comput. Educ.*, vol. 52, no. 1, pp. 1–12, Jan. 2009. <https://doi.org/10.1016/j.compedu.2008.06.004>

- [12] W. S. Yue and M. R. Abdullah, “The Motivation Design Model and Acceptance Testing Analysis of Digital Game-Based Learning (DGBL) Software,” in 2015 7th International Conference on Information Technology in Medicine and Education (ITME), 2015, pp. 668–672. <https://doi.org/10.1109/itme.2015.127>
- [13] “Le Roi et la Salamandre,” Le Roi et la Salamandre. [Online]. Available: <http://www.jeusalamandre.com/>. [Accessed: 06-Sep-2016].
- [14] R. A.-J. Johnston, Constructivism: Theorists and their Theories. 2015.
- [15] “Stencyl: Make iPhone, iPad, Android & Flash Games without code.” [Online]. Available: <http://www.stencyl.com/>. [Accessed: 30-Sep-2016].
- [16] <http://education.francetv.fr/matiere/education-civique/ce1/jeu/a-la-poursuite-de-routix-un-jeu-d-initiation-a-la-securite-routiere>
- [17] J. Biggs, “Towards a Model of School-Based Curriculum Development and Assessment Using the SOLO Taxonomy,” *Aust. J. Educ.*, vol. 33, no. 2, pp. 151–63, 1989.
- [18] Hnida, M., Idrissi, M. K., & Bennani, S. (2014). Towards an adaptive e-learning system based on individualized paths in a competency-based approach. *Recent advances in educational technologies and education*, 73.
- [19] GEE, James Paul. What video games have to teach us about learning and literacy. *Computers in Entertainment (CIE)*, 2003, vol. 1, no 1, p. 20-20. <https://doi.org/10.1145/950566.950595>

## 9 Authors

**Alaeddine YASSINE** is with Laboratoire d’Informatique et de Physique Interdisciplinaire (LIPI), ENS de Fès, Université Sidi Mohamed Ben Abdellah, Fez, Morocco (alaeyassine@gmail.com)

**Driss CHENOUNI** is with Laboratoire d’Informatique et de Physique Interdisciplinaire (LIPI), ENS de Fès, Université Sidi Mohamed Ben Abdellah, Fez, Morocco (d\_chenouni@yahoo.fr).

**Mohammed BERRADA** is with Laboratoire d’Informatique et de Physique Interdisciplinaire (LIPI), ENS de Fès, Université Sidi Mohamed Ben Abdellah, Fez, Morocco (mohammed.berrada@gmail.com).

**Ahmed TAHIRI** is with Laboratoire d’Informatique et de Physique Interdisciplinaire (LIPI), ENS de Fès, Université Sidi Mohamed Ben Abdellah, Fez, Morocco (tahiri\_ahmed@hotmail.com).

Article submitted 30 November 2016. Published as resubmitted by the authors 27 February 2017.