Curriculum Integration Ideas for Improving the Computational Thinking Skills of Learners through Programming via Scratch

Filiz Kalelioğlu
Yasemin Gülbahar
Sümeyra Akçay
Dilek Doğan

Introduction

- Today's learners are expected to possess 21st century skills : problem solving, critical thinking …
- Computational thinking skills *can be called* «*the application of critical thinking processes via existing knowledge, and computers for solving complex technological problems*»
- «*…involves solving problems, designing systems, and understanding human behaviour, by drawing on the concepts fundamental to computer science*»

Introduction

- «*...the thought processes involved in formulating problems so their solutions can be represented as computational steps and algorithms*»
- «*… an approach to solve problems through computers and also as a problem solving methodology that can be automated and transferred and applied across subjects*»
- How can we teach computational thinking skill to the students?

Objective of the paper

- The human mind is the most powerful problem-solving tool and the ability to extend the power of human thought with computers and other digital tools has become an essential part of the 21st century skills set.
- Hence, this study discusses the implementation of a visual programming tool within the related curriculum in order to equip learners with computational thinking skills at grades 5-6.

“Information and Communication Technologies and Software” Course in Turkey

- A standards-based curriculum approach was preferred and a framework was established, based on the international standards of ICT proposed by ISTE and NAACE.
- It is composed of four dimensions:
  - Digital literacy,
  - Communication, Knowledge Sharing and Self-Expression via ICT,
  - Research Knowledge Construction and Collaboration, and
  - Problem Solving, Programming and Development of Authentic Materials.

“Information and Communication Technologies and Software” Course in Turkey

- In the “Problem Solving, Programming and Development of Authentic Materials” standard, learners are expected to possess skills about Problem-Solving Approaches, Algorithm and Strategy Development, Programming and finally Software Project Development, Implementation and Dissemination.
- With this framework, teachers are free to choose the tool and software to teach computing and programming skills to children.

“Information and Communication Technologies and Software” Course in Turkey

- As a first attempt, Scratch program has been translated into Turkish and inserted into the main portal of the National Ministry of Education, or “Education and ICT Network (http://www.eba.gov.tr/ara?q=scratch)”.
- Moreover, sample tutorials are also presented in the network for different examples of the use of Scratch.
- Since our implementation was at the introductory level, we planned a 5 week course and our objective was teaching “Problem-Solving Approaches” together with some programming skills.

“Information and Communication Technologies and Software” Course in Turkey

Table 1. Intended Learning Outcomes to Teach Program Solving Approaches at 5th Grade

Standard 4. Problem Solving, Programming and Development of Authentic Materials						
Levels → Standards ↓	**Basic I: Comprehension of ICT**	**Basic II: Access to information and evaluation**	**Inter-mediate I: Managing information**	**Inter-mediate II: Information conversion**	**Advanced I: Information generation**	**Advanced II: Share information**
Problem-Solving Approaches	1.1. Defines the concepts of algorithm, strategy and problem-solving. 1.2. Awareness of the problems encountered in the use of ICT. 1.3. Refers to the importance of problem solving.	2.1. Comments on solvability of a problem in the process of problem solving. 2.2. Determines the required variables and processes to solve problems. 2.3. Realises the relationship of the concepts of algorithms and strategies.	3.1. Lists different problem-solving approaches. 3.2. Debugging to make necessary corrections to run the program correctly.	4.1. Suggests a different solution for solving the problem 4.2. Creates flowchart for displaying the solution of a problem 4.3. Creates animated scenes according to prepared by flow.	5.1. Creates the specified steps to solving the problem. 5.2. Reaches the most effective solution to questioning the validity of developed steps for problem solving.	6.1. Offers creating solutions for the identified problems and approach 6.2. Shares program code and executable file in social media.

A Computational Thinking Tool for Children: Scratch

- Visual programming software, like Scratch, Alice etc., can facilitate the teaching of programming to children without the need for memorisation, since the curriculum aims to teach problem solving and computational thinking skills to learners.
- Scratch is a free, open-source software, offering support for 61 different languages. Although Scratch is a programming language for 8-16 year old children, younger children can also work with the help of their parents.

A Computational Thinking Tool for Children: Scratch

- Scratch could be seen as an effective computational thinking tool, since a computational thinking tool must meet some conditions within in a curriculum.
  - a student can easily program a working and playable game with this tool,
  - the curriculum has to support development of these skills with these tools,
  - curriculum has to support transferring between them,
  - the tool has to support equity across gender and ethnicity boundaries, and lastly
  - a CT tool and curriculum can be used by all students and teachers.

Ideas for Curriculum Integration

- A five week instructional programme to teach problem solving skills to 5th grade primary school students via Scratch was designed by the researchers.
  - The students were taught topics such as an introduction to Scratch programming,
  - installation of a Scratch platform,
  - introduction to User Interfaces and
  - the writing of programs such as Hello World, Parrot, Aquarium programs and Maze project. In total, students have five hours to write these programs at school.

For the first week

- Students are introduced to the basic terminology and concepts of problem solving and programming.
- They practice with the Scratch program and learn about the interface and logic of the software.
- Hence, it is aimed that the learning outcomes of Basic I Comprehension of ICT level are achieved (1.1, 1.2 and 1.3 in Table 1).

For the second week

- With the Hello World program, students added a background picture to their scenes. Then they assigned a meow sound command to the cat character, they moved the cat and finally they made the cat give messages such as “Welcome”, and “Today we will learn how to scratch”.

For the second week

- In the Parrot program, students learned how to start the program, iterate the block and change the cat character. They made the parrot character fly by changing the costumes of the character. Therefore, it is intended that the learning outcomes of Basic II: Access to information and evaluation level are achieved (2.1, 2.2, 2.3 in Table 1).

In the third week,

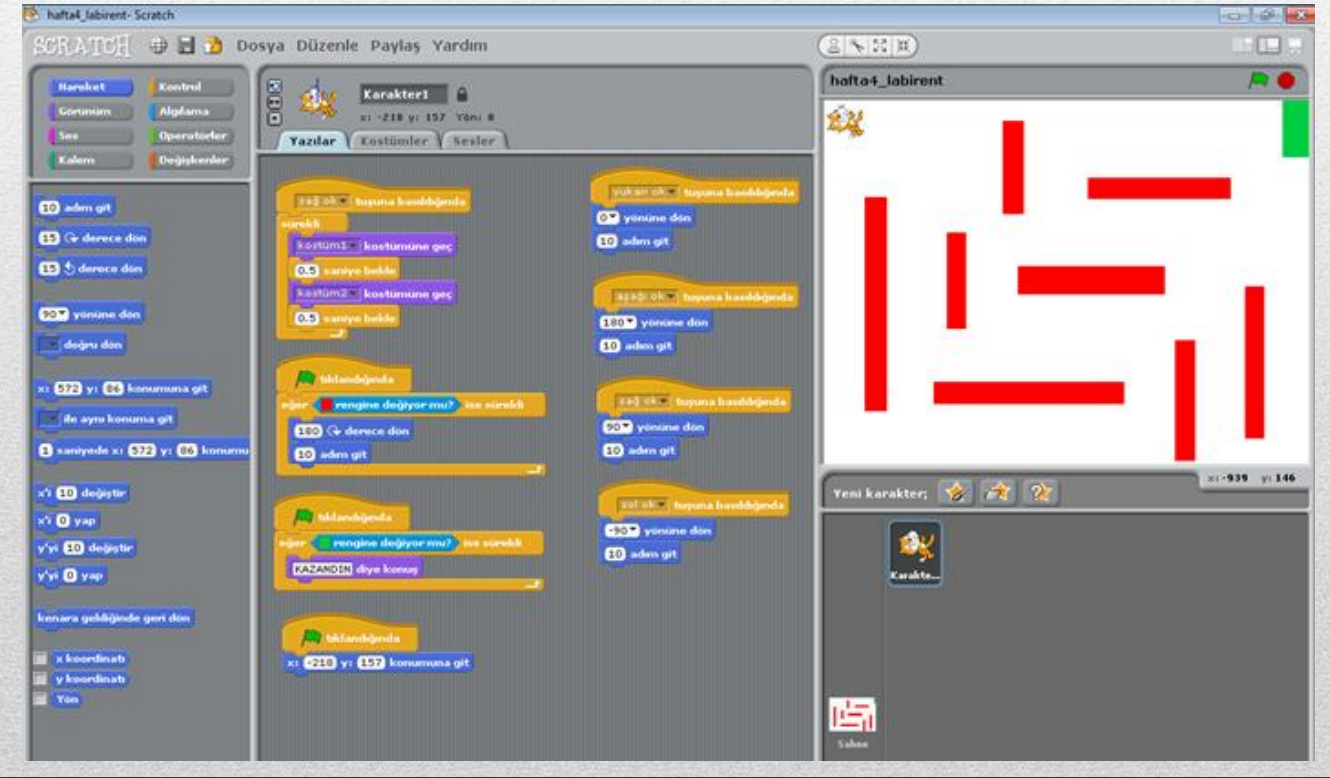
- With the Aquarium program, students learned how to manage and control more than one character, and practiced how to iterate the blocks.
- They learned to how to turn the character around if the character was on the edge, how to point towards the mouse cursor, how to change the colour effect, and how to play sounds.
- It is planned that the learning outcomes of Intermediate I: Managing information level are accomplished (3.1 and 3.2 in Table 1).

For the 4th and 5th weeks,

- Students spend two weeks on the Maze project by adding variables for calculating time spent in the game, life of the character and by adding sounds to the scene.
- Moreover, students learned how to use if conditions, how to use and activate arrow keys from the keyboard and how to change the X and Y positions of the characters.

For the 4th and 5th weeks,

- Consequently, in this lesson, different learning outcomes at different levels are accomplished (2.2, 3.1, 3.2, 4.1, 4.2 and 4.3 in Table 1).



Perceptions about the Implementation: Interview with ICT Teacher

- Scratch was pretty good as an interface for graphical programming language for students. She added that the basis of the algorithm could be given while teaching programming in Scratch.
- On the question about what kind of applications students should practice in Scratch, she suggested dialogues to begin with, then move on to design the scene, determine the character / characters, make characters talk and move and lastly, to create a game using conditions and variables. She explained that there were learning issues while teaching the conditions topic.

Perceptions about the Implementation: Interview with ICT Teacher

- In answer to the question, which competencies that students gain; she stated that students learn the rules and follow a logical sequence within the framework of blocks, learn to correct mistakes and reach quicker solutions while writing a program, and thereby learn to solve problems.
- On students' reflections about Scratch, she said that students take great pleasure from using the Scratch program. On the whole, most male students love the idea of writing a gaming program, whereas the girls usually face more difficulties.

Perceptions about the Implementation: Interview with ICT Teacher

- On suggestions about teaching programming more successfully, she said that students and teachers should take programming very seriously and should organise events. Children should be encouraged. A Turkish national resource should be created and IT teachers should share how they teach programming to their classes.

Conclusion

- When the intended learning outcomes of teaching problem solving approaches for the 5th grade were reflected into the lesson with sample program ideas for curriculum integration, it is revealed that the basic and intermediate levels were achieved, while the learning outcomes at advanced levels were not.
- Due to time limitations and distribution of the other subjects in the curricula, such a result is not surprising. In fact, for an entry level of such a course, those applications could be seen as sufficient; it could be decided that the learning outcomes at higher levels can be achieved in the upper classes as this curricula suggests.

Conclusion

- The more important point here is that whether or not students gain computational thinking skills with these intended learning outcomes. Writing programs is not enough to develop computational thinking skills, but it is important to support computational competencies.

Conclusion

- When the created applications of the students were considered, it can be said that students tried to achieve some cognitive tasks for developing computational thinking.
- Because, students processed information systematically, they used symbolic systems and representations, algorithmic notions of flow of control, practiced loops.
- And lastly, it can be said that they tried to use conditional logic, debugging with their programs, to identify, test, and implement possible solutions and to generalise solutions to other problems.

Thank you for your attention…