STEAM EmbRaCe

A STEAM project for Empathy, Resilience and Creativity

INTRODUCTION TO PROGRAMMING

Author(s)

Kechagias Andreas, Giannaros Ilias

Summary

This course is designed to provide students with a comprehensive introduction to programming using Python. It will cover the basics of programming concepts, including variables, data types, conditional statements, loops, and functions. Throughout this course, students will collaborate in groups, developing and implementing their acquired skills in projects with real-world applications. This course is suitable for students of all experience levels, offering a strong foundation for future careers in disciplines such as development, engineering, and technology.

Key elements

Key elements	*Programming fundamentals / Python syntax / Data types / Control structures / Functions / Modules*
Subject	*Computer Science*
Topic	*Introduction to Programming*
Age of students	12-17
Preparation time	8 hours
Teaching time	4-6 hours
Online teaching material	
Offline teaching material	
Resources used	

Licenses

Ai

Trends

Artificial Intelligence and Machine Learning, Data Science, Automation, Web Development, Internet of Things (IoT)

21st century skills

Creativity, Algorithmic thinking, Debugging and troubleshooting, Collaboration, Digital Literacy

Lesson Plan

Name of activity	Procedure	Time
Introduction to Algorithms and Programming	• Introduce yourself and your background in programming to initiate a conversation about the students' comprehension of the topic, its significance and get to know their experience in programming thus far to adapt accordingly. • Discuss the key concepts of algorithmic thinking and programming Ask fundamental questions on the topic such as what is a program and an algorithm, what is the difference between these concepts, why programming is important etc. Once most of the students have gotten involved and expressed their points of view, use the presentation slides to analyze and clarify the concepts.	45'
Software	• Provide a brief introduction to the platform that will be	15'

Demonstration	used so that the class is familiar with its fundamental features. Preferably, use a free online platform that is simple for tutors and students to utilize (indicatively https://replit.com).	
Introduction to Programming Fundamentals	• Use the presentation slides to explain the basic elements of programming using Python. Discuss data types, variables, conditional statements, loops, and functions. • In order to illustrate the programming concepts being taught, it's important to provide examples and encourage active participation from the class through the platform that was presented. Basic, exemplary problems should be presented that are adaptable based on the level of the class. (If the class is made up of students with differing levels of experience, it may be necessary to design different tasks to ensure that all students are appropriately challenged.) • Allow time for questions and clarification. it can be helpful to brainstorm real-world scenarios with the class to demonstrate how the concepts can be applied in practical settings.	90’
Overall Projects	• To reinforce the programming concepts taught in the lesson, it is recommended to prepare projects that require the implementation of as many of these concepts as possible. • Enable small group work among the students to improve collaboration and communication. Give them enough time to examine the problem, encourage them to design an algorithmic solution, and offer guidance while they develop their code. (It is advised that the difficulty of the problems be scalable.) • Groups should be given the chance to present their solutions and get feedback on both the outcome and the procedure they followed to get there after the assignment has been completed.	90’
Recap and Review	• Summarize the key concepts and skills learned in the course. • Allow time for further discussion. • Emphasize the importance of practice and continued learning in programming.	15’

Assessment

Here we include as an example the image of a rubric teachers can use to assess their students:

Students' and teachers' feedback after the implementation of the Learning Scenario during the Pilot phase of the project

Student feedback

Teacher's remarks

About STEAM EmbRaCe project

This Learning Scenario has been created in the framework of the STEAM EmbRaCe project.

Annex 1

Annex 2