ADVANTAGE OF SCRATCH IN IMPROVING STUDENTS' PROGRAMMING COMPETENCE

Sulaymonova D.B.

Karshi branch of the Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, teacher.

Bo’riyev F.F

Karshi branch of the Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Master's student

Abstract: this article presents recommendations for improving students' programming competence. Opinions about improving the programming competence of schoolchildren using the scratch tool are given.

Keywords: Scratch, Python, JavaScript, HTML/CSS, competence

Scratch is the most popular programming language for school students and is one of the best tools to improve their programming skills.

Scratch provides a simple and intuitive interface for school-age students and helps them learn programming principles, algorithms, and basic programming concepts. This increases their interest in programming and teaches them creativity and problem-solving thinking.

Schoolchildren can make new experiences and create creative programs through Scratch. They will be able to modify and update their programs. This helps them better understand programming and complex computer solutions.

To achieve this goal, school teachers should familiarize themselves with the guides and instructions created for learning Scratch lessons. They help students understand how to program and help them solve complex problems, develop higher-level creative thinking and complex programming skills.

Learning to program with Scratch has several advantages:

1. Simple and Intuitive Interface: Scratch's interface is easy and simple for people learning to program, making it easy to debug the code. Using blocks, you can create systematic and fast programs.

2. Graphics: Through Scratch, students create graphics programs, animations, and games. In addition to blocks and code, students work on demonstration and design.

3. Physicality: Scratch provides students with a physical understanding of programming. Working with blocks teaches students how to program through logic and advanced imagination.

4. Community Collaboration: Scratch allows students to share with each other and with the global online community. They can share and recommend their own programs with other students.

5. View and Analyze Code: With Scratch, students can quickly view and analyze their code. This provides them with an opportunity to understand and learn more about the programs you and others have created.

6. Be creative: Scratch allows students to express their ideas and solve problems through creative approaches. They can test their software in a safe environment, create their latest stuff and share their experiences with them.

These advantages make Scratch a great tool for making learning programming easier and earlier.

The following recommendations can help us to improve students' programming competence:

1. Giving students complex problems: By giving students complex problems, we have an opportunity to increase their creativity and higher-order thinking skills. By solving these problems, we will create an opportunity for them to learn programming concepts and algorithms.

2. Learning multiple programming languages: We can encourage students to learn multiple programming languages. For example, languages like Python, JavaScript, HTML/CSS. Through these languages, we can get them interested in programming and have knowledge about programming languages.

3. Exercise: Remember that exercise plays a big role in programming. Students should be engaged by preparing assignments and giving them problems to improve their programming skills.

4. Practical exercises: Teaching students through practical exercises helps them to improve their knowledge in the field of programming. For example, you should teach them how to create their own programs, and reinforce their knowledge by giving them a variety of exercises to solve complex problems.

5. Learner's Manual and Support: Providing the learner with a user manual and performance guide to work in the field of programming helps. This guide provides students with the knowledge they need to create complex programs and solve problems helps in

6. Job Exposure: Introducing students to jobs in the field of programming allows them to understand how to do programming in real life.

7. Joining programming communities: Students develop high-level programming competencies by interacting with programming communities. By joining these communities, they can collaborate with others and share ideas to create new programs.

With the help of the Scratch tool, school students will have the following advantages for developing programming competence:

1. Collaborative experience: The Scratch curriculum provides students with an interactive programming experience. It helps students learn the programming concepts they need to be able to create the programs they want.

2. Skills: Programming with Scratch helps students develop their verbal, visual, logical and mathematical skills. It helps students learn important programming concepts.

3. Job Preparation: Scratch prepares students well for practical programming. With this tool, students can participate in programming, get extra help to deal with errors and fix them.

All this helps students develop interest in programming and learn its basic concepts and develop skills in it. Scratch also helps students to develop creativity and creativity, and leads them to learn how to analyze and create the world. All these demonstrate the importance of developing good programming competence.