«Ғылыми дүниетаным құрамында барлығына міндетті болып табылатын ақиқат бар (эмпирикалық ақиқатпен бірге келетін – субъективті көзқарас пен уақытқа тәуелді болмайтын жерде)». (В.И.Вернадский.)

Дидактикада білімнің беріктілігінің келесі шарттары қалыптасқан:

- ✓ білімді саналы түрде меңгеру мақсатымен белсенді қабылдау;
- ✓ білім алудың ғылыми түрі;
- ✓ білім алуда оқу материалын есте сақтау үшін шарттар құру;

Оқыту барысындағы есте сақтау механизмінің негізін қарастырайық.

Есте сақтау дегеніміз нәтижесінде ертерек алынған біліммен байланыстыру жолы арқылы жаңа білімді бекітетін жадының үрдісі.

Есте сақтау әрқашан таңдамалы болатын: индивидтің сезу органдарына өз ықпалын тигізетіннің барлығы есте сақталмайды. Ол неге байланысты екен?

Есте сақтау субъект пен объект қызметінің заңды жемісі болып табылады. Яғни, адам немен қызмет атқарса сол ғана есте сақталады. Осылайша оқу материалын есте сақтау сәттілігі тұлға қызметінің әдістерімен, мақсаттарымен, себептерімен анықталады.

Оқудағы материалды есте сақтаудың негізгі шарттарын атап өтейін:

- ➢ оқу материалы қызмет мазмұнының негізгі мақсатына енетін жағдайда ғана жақсы меңгеріледі. Мысалы, егер оқушы қызметінің мақсаты болып тригонометриялық фигураның құрылымын анықтау болса, онда оны есте сақтау сол фигуралардың құрылымы мұғалімге байланысты болғаннан гөрі жақсы болады.
- ➢ Оқу материалы сол материалмен жұмыс жасау барысында ой қабілетінің белсенді жұмысында жақсы меңгеріледі. Сондықтан жеңіл материалдан гөрі қиын материал жақсы есте қалады. Себебі, қиын мәтіннің элементтері анағұрлым мазмұнды болып келеді.

Осылайша математика пәні бойынша үлгермеуді алдын алу нақты материалды меңгеру ерекшеліктеріне байланысты. Математика бойынша оқу бағдарламасының нақты сұрақтарының әдістемелік жолмен дұрыс ұйымдастырылып меңгерілуі, әр оқушыны жұмысқа белсенді қатысуға тарту, оқушыларға көбірек көңіл бөлу олардың математика бойынша оқу бағдарламасын тереңірек меңгеруге өз септігін тигізеді. Бірақ қандай жағдайда болсын оқушыларға қатысты математика сабағында жағымда атмосфера құру қажет. Сонымен қатар, ұйымдастырылған жұмыстар математиканы меңгеруге деген жақсы қатынасты, өзіндіктілікті дамытуға ықпалын тигізу керек.

Қолданылған әдебиеттер:

1. Кобдикова Ж.У. «Деңгейлеп саралап оқыту » 2000ж 15 б
2. Ахметов С. «Білім берудің тиімді жолдары »1989ж 254 б
3. Л. Занков, Д. Эльконин, В. Давыдов, В. Репкин, В. Левин «Дамытаотырыпоқытуәдістемесі»1989ж 65-98 б
4. Якиманская Н.С. «Дамытаоқытутұжырымдамасы »1997ж 36-45 б
5. Кененбаева М.А. «Бастауыш және негізгі мектепте математиканы дамыта оқытудағы сабақтастық мәселелері». Автореферат. - Астана, 2005 ж.32 б
6. Оспанов Т.Қ. «Бастауыш кластарда математиканы оқыту» Алматы, 2005 ж.65б
7. Оспанов Т.Қ Кочеткова О.В.,. Астамбаева Ж.Қ. Жаңа буын оқулықтары бойынша бастауыш сыныптарда математика оқыту әдістемесі. - Алматы, 2005.49б
8. Жақыпбекова Г.Т. Математика сабағында математикалық сауаттылықты қалыптастыру. Автореферат: Алматы. 2001.

УДК: 004.424

USING SCRATCH IN LEARNING PROCESS

Dana Zhumagulova
MSc of CS, Suleyman Demirel University, Almaty, Kazakhstan, dana_galymzhan@bk.ru
Kuralay Zhanabay

MSc of CS, Suleyman Demirel University, Almaty, Kazakhstan, j.kuralay@yahoo.com

Abstract

This paper describes usage of computer program Scratch in learning process. It consists of 3 parts. In first part there is general information about Scratch, then in second part there are lessons of how to use. The last part shows conclusion of using it.

Keywords: Scratch, MIT, learning process.

Introduction

Scratch is a object-oriented programming language learning environment what enables it's users to get results without having to learn syntactically correct writing first. It is a graphical programming environment that makes it easier for kids (ages 8 and up) to create their own interactive stories, games, animations, and simulations, and share their creations with one another online.

Scratch was created by the Massachusetts Institute of Technology Media Laboratory. Individuals who use Scratch at a young age develop a solid foundation of knowledge that helps prepare them for higher level programming languages later in the future[1]. Scratch allows creating a strong interest of children to programming; it meets all the requirements of modern object-oriented programming.

One of the main advantages of this system is the availability of versions for different operating systems: Windows, Linux and Mac OS.

1. General information about Scratch

Scratch has tactile and visual graphical user interface which allows children to explore by dragging-and-dropping blocks of conditions and consequences onto selected agents and backgrounds how interactive animations, presentations, stories, and simple games can be playfully and programmatically created. To include children who are not able to write syntactically correct structures – nor read them, yet – visually grouped blocks can be tested by clicking on them and they can be easily replaced with different ones in order to remix, modify, and create new versions of projects. You can see the interface of Scratch in figure 1 given below.

The user interface of the Scratch environment includes several panes: on the left is the blocks palette, in the middle the current sprite info and scripts area, and on the right the stage (backgrounds) and sprites list.

The blocks palette has code fragments (called "blocks") that can be dragged onto the scripts area to make programs. To keep the palette from being too big, it is organized into eight groups of blocks: motion, looks, sound, pen, control, sensing, operators, and variables. Different kinds of blocks have different colors and shapes.

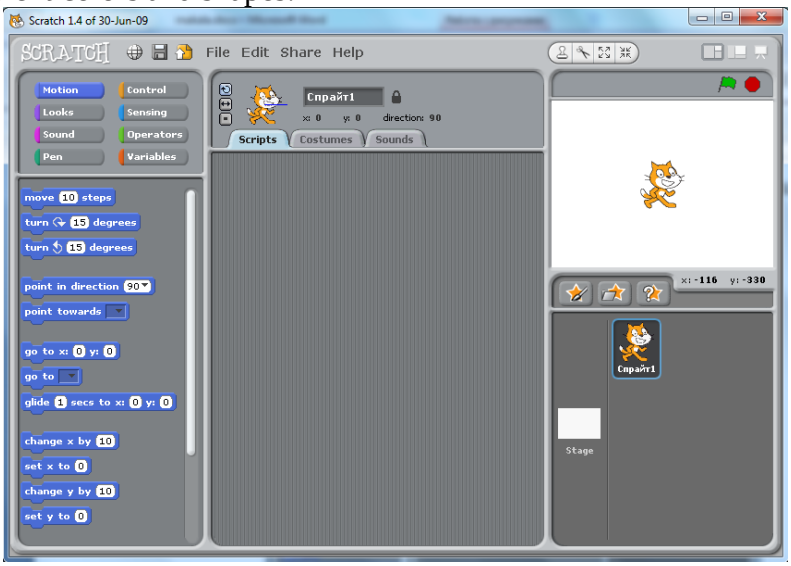


Figure 3 The Scratch Interface

Sprites are the main characters of visual part of program. There can be many sprites as user wants. The stage is a background what can be easily changed by choosing other pictures from file or Scratch library of backgrounds. And the program code in Scratch is called scripts.

The middle of the window is where the program is assembling by simply dragging and dropping each building block user wishes to use. Scratch will even help assemble them in the middle so they fit together to create the action desired. Most of the hundreds of tools or building blocks are intuitive, so kids can learn them by trying them out. [2]

What do students learn as they create interactive stories, animations, games, music, and art with Scratch?

By programming on Scratch kids can develop their following skills:

1. Information and Media Literacy Skills. By working on Scratch projects, students learn to select, create, and manage multiple forms of media, including text, images, animation, and audio recordings. As students gain experience creating with media, they become more perceptive and critical in analyzing the media they see in the world around them.

2. Communication Skills. Effective communication in today's world requires more than the ability to read and write text. Scratch engages young people in choosing, manipulating, and integrating a variety of media in order to express them creatively and persuasively.

3. Critical Thinking and Systems Thinking. As they learn to program in Scratch, young people become engaged in critical reasoning and systems thinking. In order to build projects, students need to coordinate the timing and interactions between multiple "sprites" (programmable moving objects). The ability to program interactive input provides students direct experience with sensing, feedback, and other fundamental systems concepts.

4. Problem Identification, Formulation & Solution. Scratch supports problem finding and solving in a meaningful design context. Creating a Scratch project requires thinking of an idea, then figuring out how to break the problem into steps and implement those using Scratch programming blocks. Scratch is designed to be "tenderable": students can dynamically change pieces of code and immediately see the results (e.g., doubling a number to see how it changes a graphic effect). Throughout the design process, students engage in experimenting and iterative problem-solving.

5. Creativity and Intellectual Curiosity

Scratch encourages creative thinking, an increasingly important skill in today's rapidly changing world. Scratch involves young people in seeking innovative solutions to unexpected problems—not just learning how to solve a predefined problem, but being prepared to come up with new solutions as new challenges arise.

6. Interpersonal and Collaborative Skills. Because Scratch programs are built of graphical blocks, the programming code is more readable and shareable than other programming languages. The visual objects and modular code supports collaboration, enabling students to work together on projects and exchange objects and code.

7. Self-Direction. Taking an idea and figuring out how to program it in Scratch requires persistence and practice. When young people work on project ideas they find personally meaningful, their ideas provide internal motivation for overcoming challenges and frustrations encountered in the design and problem-solving process.

8. Accountability and Adaptability. When students create Scratch projects, they have an audience in mind, and need to think about how other people will react and respond to their projects. Since Scratch projects are easy to change and revise, students can modify their projects based on feedback from others.

9. Social Responsibility. Because Scratch programs are shareable, students can use Scratch to provoke discussion of important issues with other members of their immediate learning environment, as well as with the wider international Scratch community [3]

The next one is the tutorial what explains how to start working on Scratch.

2. Using Scratch in course

2.1 Getting Started with Scratch

This is the Scratch window:

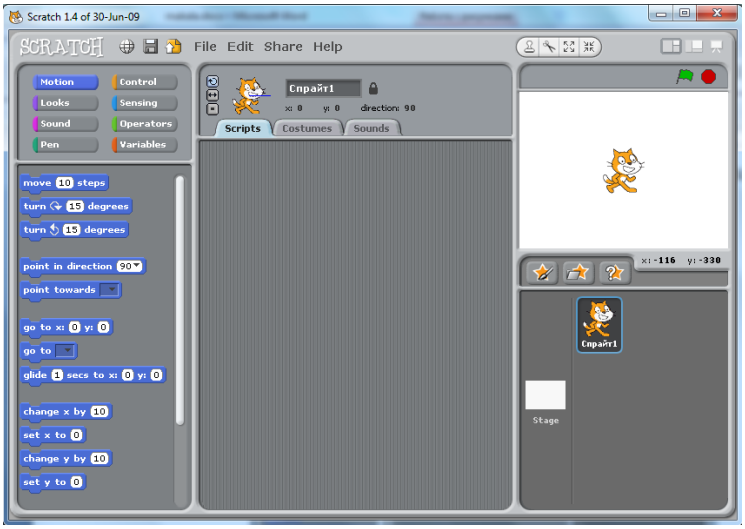


Figure 4 Main page

Here we can create our own stories, games, animations.

As you see the Scratch interface consists of three parts: Part 1: blocks palette, Part 2: sprite info and scripts area, Part 3: stage and sprites list.

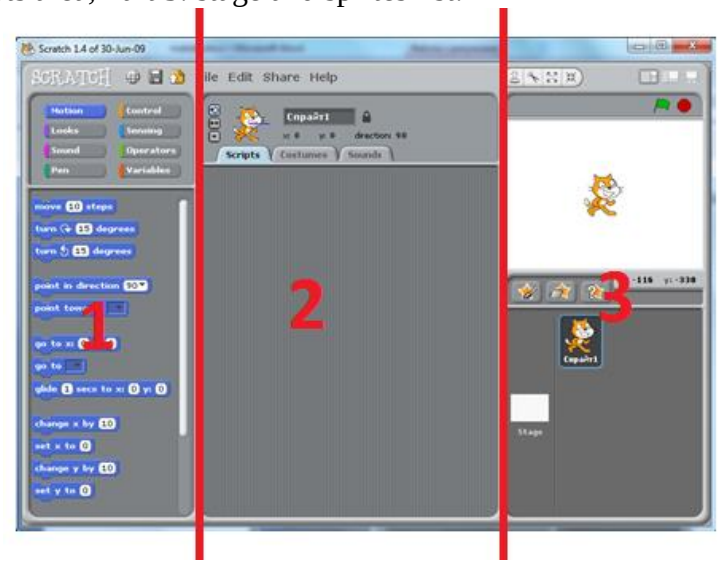


Figure 5 Scratch 3 parts

Now we can see only one character on the stage:

Figure 6 Character

But we can also change this character to an another one.

So let's start doing our first program.

When we first open the Scratch environment we can see only Motion blocks on the blocks palette. But in Scratch there are 8 blocks: Motion, looks, sound, pen, control, sensing, operators, and variables:

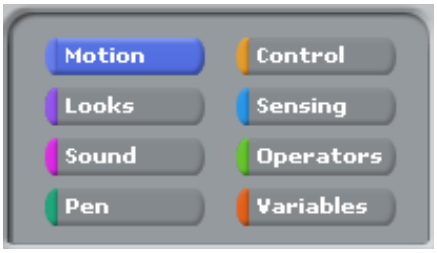


Figure 7 Motion block

So let's choose the script move 10 steps 

from the Motion blocks. Choosing scripts is doing by dragging and dropping block to the scripts area:

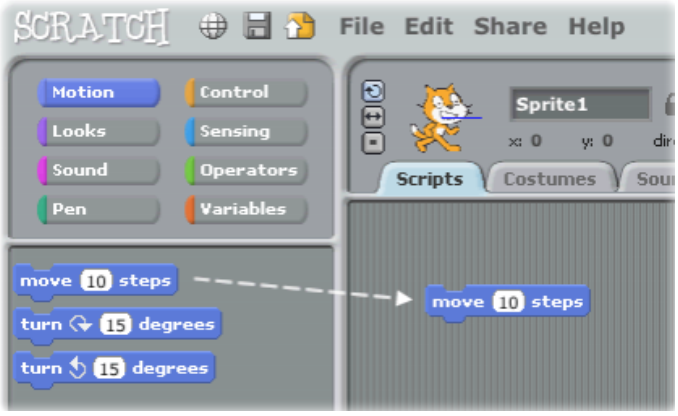


Figure 8 Script area

Now each clicking on this script will let our character move 10 steps forward. But, of course, we can change the number of steps. So even this one script we can call program.

In Scratch programs must run by clicking the green flag  . But now it's impossible, because our program is not complete. To complete our program it needs one more script. It's a 

from Control blocks. This script allows the program character to understand what commands to do when the green flag clicked.

So let's join 

to 

, also by dragging and dropping:

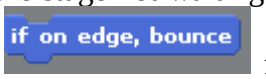
Figure 9 Example of drop

Now our program will start by clicking green flag.

Let's develop our program. Now our program let's the character to move 10 steps and stop. To move the character permanently choose the script forever 

and join it to the program code as shown below:

Figure 10 Example of loop

As a result we see that the character starts to move by clicking green flag and it must stop only by clicking red button  . But we see that the character stops at the end of the stage not waiting clicking the red button. It means we must complete our program. Join the script 

from the Motion blocks:

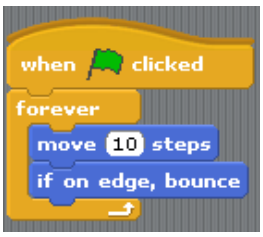


Figure 11 Example of program

What is the result? The cat moves on the stage and when it gets the end of the stage turns back and moves again. But here is a problem: when cat goes back it moves head over heels:

Figure 12 The output of program

Of course we can solve this problem. Go to the sprite info, it's located in the middle at the top of the window:

Figure 13 Sprite view

Here we can see 3 buttons:

 - can rotate

 - only face left-right

 - don't rotate.

Now we see that the button can rotate  is choosen. So why cat moves head over heels. We must choose only face left-right  to let the cat to move correctly:

Figure 14 Output

Now cat moves on the stage permanently. So this is our little, but perfect first program on Scratch.

What did we learn?

So we can write programs in Scratch environment. To write programs we use special blocks called scripts. The result of our program we can see by how characters responds on the stage.

This is our first program on Scratch:

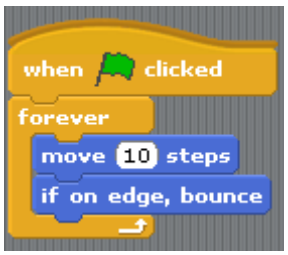


Figure 15 General program

The meaning of this program is when we click the green flag cat moves on the stage, when it gets to the end of the stage it goes back and so on. This proccess will continue untill we click the red button.

There is a one cycle on this program. We use cycle when we need repeating of some actions. In our program the cycle is the script forever, because it lets the 2 scripts including it to repeat permanently.

Task 1. Write this program and explain how it works:

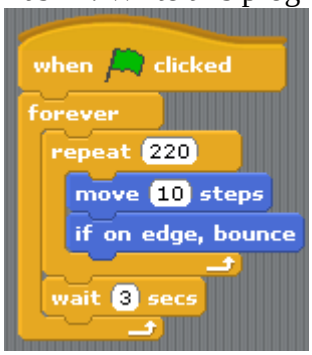


Figure 16 Task1 view

Make the following changes to the program given in Task 1: may the cat after each 20 steps wait 1 second.

3. Conclusion

Of course, most students will not grow up to become professional programmers, just as most will not become professional writers. But learning to program offers benefits for everyone: it enables students to express themselves more fully and creatively, helps them develop as logical thinkers, and helps them understand the workings of the new technologies that they encounter everywhere in their everyday lives.

[1] (http://en.wikipedia.org/wiki/Scratch_(programming_language))
[2] (http://cmc-math.org).
[3] (http://learnscratch.org/resources/why-learn-scratch).

УДК: 004.424

Шахин З.
Иогуртчу Я.
Сүлеймен Демирел университетінің Магистранттары
Алматы, Қазақстан,

Андатпа

Стандартты емес есептердің түрінің көптігін ескере отырып, белгілі саласының, соның ішінде стандартты емес теңдеулер мен теңсіздік ұғымдарының анықтамасын жүйелеп, оқыту әдістемесіне тоқталып, есептер шығару үрдісінде ең тиімді әдіс-тәсілдерді анықтап, мектеп курсында оқу бағдарламасына факультатив негізінде енгізе отырып, эксперимент жасап, жақсы нәтижелерге қол жеткізу.Осы мақсатқа жету үшін стандартты емес есептерді түрлеріне қарай классификациялап, олардың әрқайсысының шығару жолын және әдістерін табу міндеті қойылды.

Кілт сөздер: математика, стандарт есептер, графика.