

УДК 372.8
DOI 10.17513/snt.39915

ОБ ОРГАНИЗАЦИИ ПРОБЛЕМНОГО ОБУЧЕНИЯ ИНФОРМАТИКЕ НА ПРИМЕРЕ ТЕМЫ «ИСПОЛНИТЕЛИ И АЛГОРИТМЫ» С ИСПОЛЬЗОВАНИЕМ СРЕДЫ SCRATCH

^{1,2}Акимова И.В., ¹Губанова О.М., ³Титова Н.В.

¹ФГБОУ ВО «Пензенский государственный университет», Пенза,
e-mail: ulrih@list.ru, olga.penza@mail.ru;

²ФГБОУ ВО «Московский государственный университет технологий и управления
имени К.Г. Разумовского» (Первый казачий университет) (филиал), Пенза;

³ФКВБОУ ВО «Военная академия материально-технического обеспечения
имени генерала А.В. Хрулёва», Пенза, e-mail: nvtitova77@mail.ru

В своей работе авторы рассматривают возможность организации проблемного обучения информатике на примере темы «Исполнители и алгоритмы». Проблемное обучение представляется одной из инновационных технологий обучения, которая позволяет активно использовать наблюдение, выдвигать собственные гипотезы, активно включаться в работу всем обучающимся. В первой части статьи авторы исследуют теоретические основы реализации проблемного обучения в школе, рассматривая работы таких авторов, как М.И. Махмутов, Г.К. Селевко, И.Я. Лернер. При выборе темы школьного курса информатики, на которой будут реализовываться элементы проблемного обучения, предлагается тема «Исполнители и алгоритмы». Выбор обусловлен сложностью изучения данного материала и его широким использованием в последующих темах. Авторы проводят анализ нескольких популярных сред, таких как Кумир, Чертежник. В итоге в качестве среды исполнителя предлагается использовать среду Scratch. Методическая составляющая исследования посвящена реализации элементов проблемного обучения при изучении циклических алгоритмов. Выбор данного вида алгоритма определяется тем, что циклическая конструкция является одной из самых распространенных при решении задач. Авторы приводят примеры создания проблемной ситуации и деятельности обучающихся по ее разрешению. Авторы сообщают об эффективности предложенных методических решений.

Ключевые слова: проблемное обучение, проблемная ситуация, алгоритм, Scratch

ABOUT THE ORGANIZATION OF PROBLEM-BASED TRAINING IN INFORMATICS USING THE EXAMPLE OF THE TOPIC “PERFORMERS AND ALGORITHMS” USING THE SCRATCH ENVIRONMENT

^{1,2}Akimova I.V., ¹Gubanova O.M., ³Titova N.V.

¹Penza State University, Penza, e-mail: ulrih@list.ru, olga.penza@mail.ru;

²K.G. Razumovsky Moscow State University of Technologies and Management
(the First Cossack University), Penza;

³Military Academy of Logistics named after General A.V. Khrulev, Penza,
e-mail: nvtitova77@mail.ru

In their work, the authors consider the possibility of organizing problem-based computer science education on the example of the topic “Performers and algorithms”. Problem-based learning seems to be one of the innovative learning technologies that allows you to actively use observation, put forward your own hypotheses, and actively involve all students in the work. In the first part of the article, the author explores the theoretical foundations of the implementation of problem-based learning at school, considering the works of such authors as M.I. Makhmutov, G. K. Selevko, I.Ya. Lerner. When choosing the topic of the school computer science course, on which the elements of problem-based learning will be implemented, the topic “Performers and algorithms” is proposed. The choice is due to the complexity of studying this material and its wide use in subsequent topics. The authors analyze several popular environments, such as Idol, Draftsman, As a result, it is proposed to use the Scratch environment as the performer’s environment. The methodological component of the study is devoted to the implementation of the elements of problem-based learning in the study of cyclic algorithms. The choice of this type of algorithm is determined by the fact that the cyclic construction is one of the most common in solving problems. The authors gives examples of creating a problem situation and the activities of students to resolve it. The authors report on the effectiveness of the proposed methodological solutions.

Keywords: problem-based learning, problem situation, algorithm, Scratch

В настоящее время идет внедрение инновационных образовательных технологий в школьное образование. Главной целью является получение качественных изменений

личности обучающегося, которые не может обеспечить традиционная система. Такие изменения могут быть достигнуты с помощью привития навыков мотивационной

деятельности, самостоятельности в получении новой информации, формирования нестандартности в мышлении.

Решение обозначенной задачи видится в использовании технологии проблемного обучения. Проведение уроков информатики с использованием элементов проблемного обучения позволяет активно использовать наблюдение, выдвигать собственные гипотезы, активно включаться в работу всем обучающимся.

Одной из целей изучения в соответствии с Федеральной рабочей программой основного общего образования по информатике (базовый уровень) является «обеспечение условий, способствующих развитию алгоритмического мышления как необходимого условия профессиональной деятельности в современном информационном обществе, предполагающего способность обучающегося разбивать сложные задачи на более простые подзадачи, сравнивать новые задачи с задачами, решенными ранее, определять шаги для достижения результата и так далее» [1].

Исходя из всего сказанного, можно определить актуальность выбранной темы исследования.

Цель исследования может быть сформулирована как рассмотрение возможности реализации элементов проблемного обучения на уроках информатики на примере темы «Исполнители и алгоритмы».

Материалы и методы исследования

В отечественной методической науке концепция проблемного обучения получает свое развитие в 1970-е гг. В работах таких авторов, как М.Н. Скаткин, И.Я. Лернер, В. Оконь, Н.А. Менчинская, М.А. Данилов, Ю.К. Бабанский, М.И. Махмутов, А.М. Матюшкин, А.В. Хуторской и др. [2–5], встречаются разработки тех или иных аспектов проблемного обучения, происходит становление проблемного обучения как концепции в целом.

Свое начало данное направление берет в теоретических работах педагога и философа Дж. Дьюи, который в своей работе «Как мы мыслим» заявил о высокой эффективности обучения, которое предполагает организацию активной деятельности, в ходе которой ребенок самостоятельно может решить учебную проблему. Впоследствии Дж. Дьюи ввел и психологическое обоснование данного механизма самостоятельного решения ребенком проблем.

Если говорить об отечественной науке, то идеи проблемного обучения стали развиваться со второй половины 1950-х гг.

Согласно М.И. Махмутову, можно определить проблемную ситуацию как интеллектуальное затруднение человека, возникающее в случае, когда он не знает, как объяснить возникшее явление, факт, процесс действительности, не может достичь цели известным ему способом, что побуждает человека искать новый способ объяснения или способ действия [6].

По степени проблемы существуют три основных уровня проблемных ситуаций. Первый уровень – творческое обучение, которое подразумевает активное участие обучающихся в постановке проблемы и поиске путей ее решения. Второй уровень – проблемная ситуация создается учителем и решается совместно с обучающимися. Третий уровень – учитель создает проблему и сам находит ее решение [7].

Под организацией проблемного обучения на уроках информатики будем понимать проектирование проблемных ситуаций, которые требуют от обучающихся проявления инициативы, ведения творческого поиска решения задачи, а также при парной и групповой работе взаимодействия и совместной деятельности.

В настоящее время существует достаточно много сред исполнителей алгоритмов, которые могут быть использованы как в школьном курсе информатики, так и в рамках внеурочной деятельности и дополнительного образования [8, 9].

Представим описание и сравнение нескольких популярных сред.

1. Формальный исполнитель Чертежник: предназначен для построения рисунков, чертежей и т.д. на координатной плоскости.

2. Графический исполнитель Стрелочка: программа-тренажер для развития алгоритмического мышления и формирования умений составления управляющих алгоритмов.

Достоинством первых двух исполнителей является доступность среды. Из недостатков можно выделить малую наглядность, отсутствие мотивационной и креативной составляющих.

3. Среда программирования Scratch: визуальная программно-ориентированная среда для отработки навыков алгоритмизации. Обучающиеся могут выбрать или создать сами действующего персонажа, которым они будут управлять (спрайт). Среда обладает возможностями создания анимации, живых открыток, рисования и других видов творческой деятельности. Достоинствами использования данной среды является доступность работы в онлайн-режиме, интуитивность интерфейса, красочность, возможность реализации креативного под-

хода, возможность использования переменных, массивов.

Предлагается выбрать среду Scratch. Scratch был создан на языке Squeak, который представляет собой одну из разновидностей Smalltalk. Главным идеологом Scratch является ученик Пейперта Мич Резник из MIT MediaLab (Массачусетский технологический институт), именно в нем в 1968 г. С. Пейперт разработал Logo.

Scratch имеет собственный редактор текста программы, построенный на интересной идее конструкторов Lego: все операторы языка и другие его элементы представлены блоками, которые могут соединяться один с другим, образуя скрипт. Программа собирается из разноцветных блоков, таких как движение, внешний вид, звук, операторы и др. [10].

Результаты исследования и их обсуждение

Рассмотрим методику реализации элементов проблемного обучения на примере темы «Конструкция “повторения”»: циклы с заданным числом повторений, с условием выполнения, с переменной цикла».

В школьном курсе рассматриваются традиционно несколько классических видов алгоритмов, среди которых циклы, или циклические алгоритмы, занимают центральные позиции. Составление циклических алгоритмов различной степени сложности – важная задача пропедевтики курса программирования. Объяснение видится в том, что повторения, те или иные итерации окружают нас постоянно, поэтому существует широкий круг циклических задач.

Рассмотрим методические аспекты реализации проблемного обучения при рассмотрении данной темы.

Постановка проблемы

Учитель ставит перед обучаемыми следующую задачу.

Задача 1. Изобразить квадрат с произвольной стороной.

Обучающиеся могут предложить следующее решение, сторона квадрата – 100 шагов (рис. 1).

Осознание противоречий

Учитель обозначает противоречие в предлагаемом решении. Все ли шаги устривают в алгоритме? Нет ли лишних команд?

В результате анализа учащиеся делают вывод, что некоторые одинаковые команды алгоритма выполняются несколько раз.

Далее учитель ставит проблему, возможно ли предложить другое решение, которое

не будет содержать многократного повтора одинаковых действий.

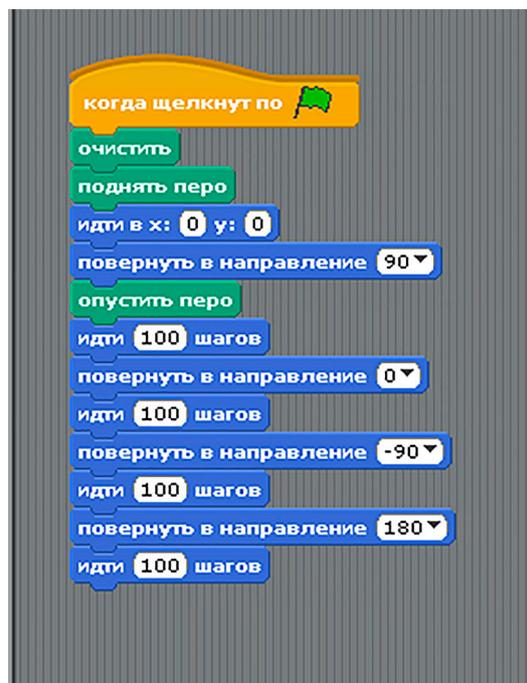


Рис. 1. Алгоритм решения задачи 1 без цикла

Обучающиеся выделяют следующую группу действий, которая была выполнена несколько раз: пройти 100 шагов, повернуть в определенном направлении.

Далее учитель на основе полученных выводов дает следующее определение: такая алгоритмическая конструкция, когда многократно выполняется одно и то же действие, называется цикл.

Поиск решения проблемы

После вывода определения учитель выдвигает идею записи решения данной задачи с использованием циклической конструкции. Предлагается выделить группу действий, которая многократно повторилась: пройти 100 шагов, повернуть направо на 90°. Сколько раз было выполнено повторение? Учащиеся отмечают, что повторы были выполнены 4 раза. Предлагается составить новый итоговый алгоритм.

Доказательство гипотезы

Используя соответствующую конструкцию среды Scratch, учащиеся получают новый алгоритм (рис. 2) и результат его работы.

Далее для закрепления учитель предлагает решить следующую задачу.

Задача 2. Изобразить цифру 5.

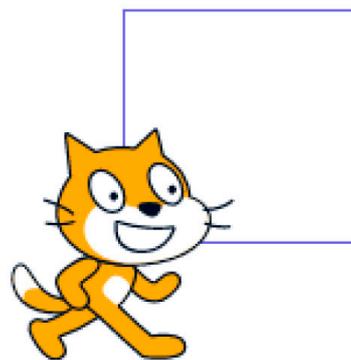
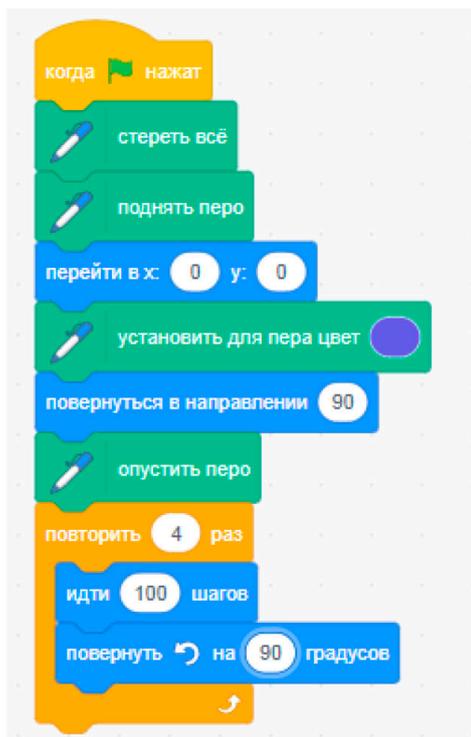


Рис. 2. Алгоритм решения задачи 1 с циклом

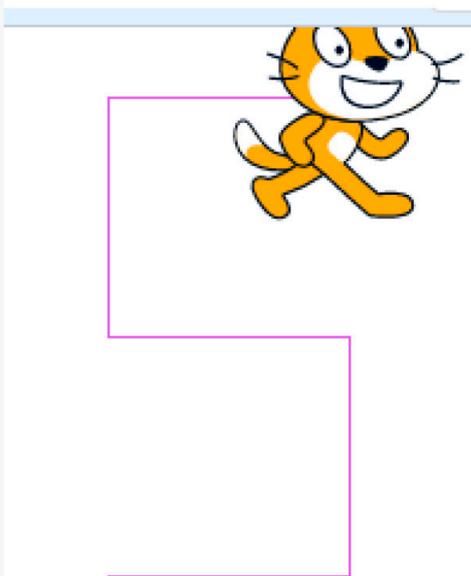


Рис. 3. Алгоритм решения задачи 2 и результат работы

Обучающиеся самостоятельно пробуют решить эту задачу, предлагают свои решения, среди которых могут быть и варианты с циклом: не полный цикл из задачи 1, а только его часть – повторений должно быть только три. Остальную же часть изображения можно построить и без использования цикла.

Далее учитель также может предложить учащимся задачи, в которых могут быть использованы аналогичные циклы.

Задача 3. Изобразить цифру 6.

Обучающиеся также делают предположение, что для изображения цифры 6 также может быть использован цикл, построенный ранее для изображения квадрата. Поэтому выдвигается предположение, что можно использовать часть алгоритма из первой задачи, а затем выполнить еще одно дополнительное построение.

Но возникает проблема, так как исполнитель завершает движение не в той точке, где должно быть выполнено дополнительное построение.

Могут быть выдвинуты несколько предположений:

- 1) пройти к нужной точке с поднятым пером;
- 2) изменить порядок изображения квадрата.

Для реализации первого предположения учащиеся строят алгоритм, в результате ко-

торого исполнитель оказался в нужной точке и осталось выполнить дополнительное построение (рис. 4).

В результате обучающиеся делают вывод, что предположение подтвердилось, и алгоритм строит заданное изображение.

Для реализации второго предположения обучающиеся изменяют алгоритм из первой задачи, и делается вывод, что исполнитель оказывается в нужной точке и также можно продолжить построение. В результате обучающиеся также делают вывод, что второе предположение подтвердилось, и алгоритм строит заданное изображение (рис. 5).

Накопленный опыт реализации элементов проблемного обучения при изучении циклических алгоритмов можно применить и в рамках дополнительного образования. Учебно-тематический план представлен в таблице.

Применение элементов проблемного обучения в рамках дополнительной общеобразовательной программы обеспечивает умение самостоятельно формулировать проблему и ставить задачи для ее решения; находить и отбирать актуальную информацию; представлять результаты работы перед аудиторией и др.

Таким образом, представлены примеры заданий для реализации элементов проблемного обучения при изучении циклических алгоритмов.

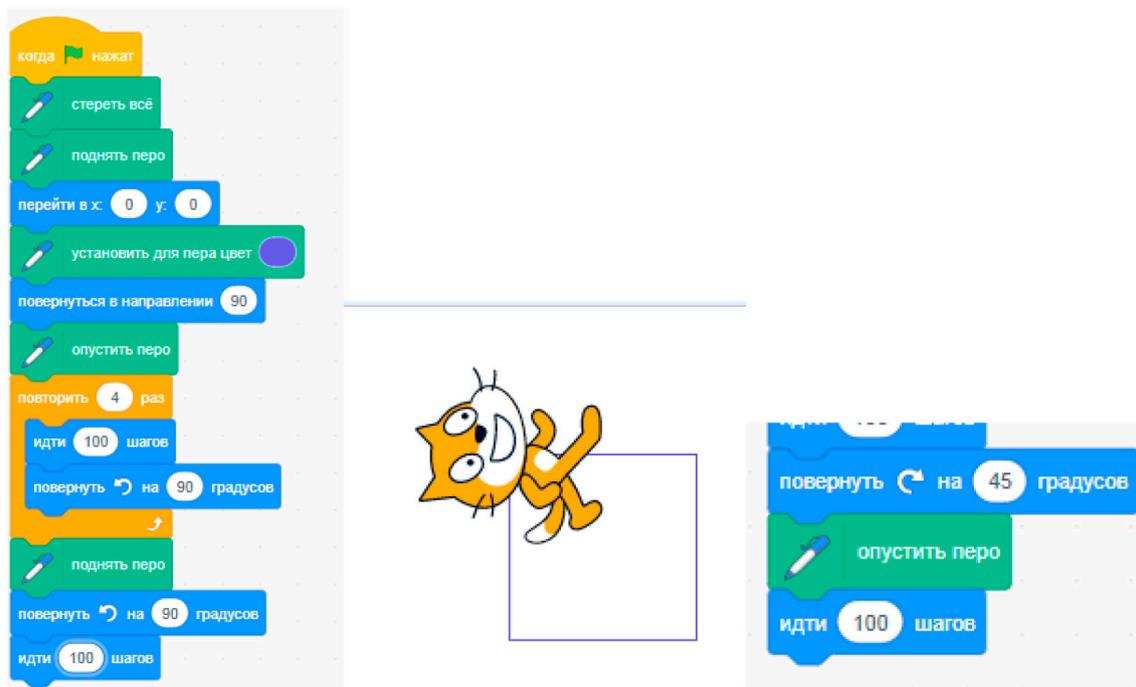


Рис. 4. Фрагмент алгоритма и результат работы, задача 3

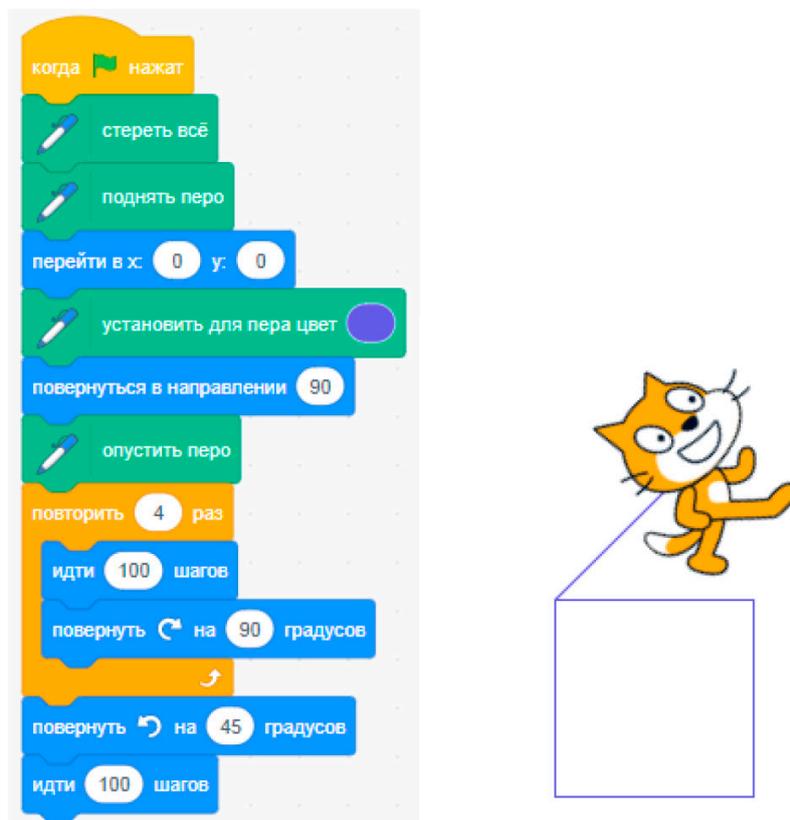


Рис. 5. Алгоритм решения и результат работы задачи 3

Учебно-тематический план

№ п/п	Название темы занятия	Количество часов			Формы контроля
		всего	теория	практика	
	Среда программирования Scratch	42	8	34	
1.	Особенности среды программирования	4	1	3	Фронтальный опрос. Оценка выполнения индивидуальных практических заданий
2.	Создание программ. Линейные алгоритмы	4	1	3	
3.	Создание программ. Разветвляющиеся алгоритмы	4	1	3	
4.	Создание программ. Циклические алгоритмы	6	1	5	
5.	Создание подпрограмм	6	2	4	
6.	Создание игр, мультфильмов и динамичных комиксов	8	2	6	
7.	Подготовка индивидуального проекта	10		10	Оценка проекта

Заключение

Данное методическое решение было опробовано в МБОУ СОШ с углубленным изучением информатики № 68 г. Пензы.

Тема «Алгоритмы» в соответствии с Федеральной рабочей программой основного общего образования по информатике (базовый уровень) изучается в 8 классе. Результаты диагностических проверочных работ,

проведенных в восьмых классах, показали эффективность предложенных методических решений.

Перспективой данного исследования является распространение методики реализации элементов проблемного обучения на последующие темы школьного курса информатики, в частности программирование.

Список литературы

1. Приказ Министерства просвещения РФ от 31 мая 2021 г. № 287 «Об утверждении федерального государственного образовательного стандарта основного общего образования» [Электронный ресурс]. URL: <https://www.garant.ru/products/ipo/prime/doc/401333920/> (дата обращения: 01.10.2023).
2. Лернер И.Я. Проблемное обучение. М., 1974. 64 с.
3. Оконь В. Введение в общую дидактику. М., 2014. 22 с.
4. Оконь В. Основы проблемного обучения. М.: Просвещение, 2013. 65 с.
5. Быстрова Н.В., Зиновьева С.А., Филагова Е.В. Проблемное обучение в современном образовании // Проблемы современного педагогического образования. 2020. № 67–1. С. 43–46.
6. Тузинец А. Две концепции проблемного обучения: В. Оконь и М.И. Махмутов // Вестник Московского государственного лингвистического университета. Секция: Образование и педагогические науки. 2020. № 3 (836). С. 128–139.
7. Кудрявцев Т.В. Проблемное обучение: истоки, сущность, перспективы. М.: Знание, 2016. 80 с.
8. Уфимцева П.Е., Рожина И.В. Обучение программированию младших школьников в системе дополнительного образования с использованием среды разработки Scratch // Наука и перспективы. 2018. № 1. С. 29–35.
9. Акимова И.В. Методика изучения линейных алгоритмов с помощью среды Scratch // Информатика в школе. 2013. № 7 (90). С. 28–33.
10. Сайт среды Scratch/ [Электронный ресурс]. URL: <https://scratch.mit.edu/> (дата обращения: 01.10.2023).