



Л. Л. Босова

Московский педагогический государственный университет, г. Москва, Россия

А. Ю. Босова

Учебно-методический центр, г. Истра, Московская область, Россия

В. И. Филиппов

Академия социального управления, г. Москва, Россия

«ПРОГРАММИРУЕМ, УЧИМСЯ И ИГРАЕМ!». ПРОГРАММА КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ ДЛЯ УЧАЩИХСЯ III—VI КЛАССОВ

Аннотация

В статье обосновывается актуальность раннего обучения программированию и описывается возможный вариант такой работы с обучающимися III—VI классов в рамках курса внеурочной деятельности общеинтеллектуальной направленности. Раскрыт потенциал курса в достижении обучающимися личностных, метапредметных и предметных результатов. Приведено тематическое планирование и содержание обучения младших школьников в среде программирования Scratch. Изложена основная идея методики проведения внеурочных занятий с младшими школьниками: работа с готовыми примерами по схеме «использование — изменение — дополнение — создание», позволяющая обучающимся создавать цифровые продукты (мультимедийные открытки, интерактивные плакаты, игры, тренажеры и викторины).

Ключевые слова: младшие школьники, программирование, внеурочная деятельность, Scratch, Скретч.

DOI: 10.32517/2221-1993-2021-20-6-3-15

1. Введение

Жизнь современного человека тесно связана с использованием широкого спектра стремительно изменяющихся компьютерных устройств и информационных технологий. Задача современной школы — обеспечить вхождение обучающихся в информационное общество, научить каждого пользоваться информационно-комму-

никационными технологиями, целенаправленное изучение которых в отечественной школе в обязательном порядке происходит довольно поздно — в VII—IX классах. К этому времени все обучающиеся уже имеют стихийно сформировавшийся опыт использования ресурсов и сервисов сети Интернет; многие из них обладают теми или иными навыками создания цифровых продуктов: текстовых документов, изображений, звукозаписей,

Контактная информация

Босова Людмила Леонидовна, доктор пед. наук, доцент, зав. кафедрой теории и методики обучения математике и информатике, Институт математики и информатики, Московский педагогический государственный университет, г. Москва, Россия; *адрес:* 119991, Россия, г. Москва, ул. Малая Пироговская, д. 1, стр. 1; *e-mail:* akull@mail.ru

Босова Анна Юрьевна, директор Учебно-методического центра, г. Истра, Московская область, Россия; *адрес:* 143500, Россия, Московская область, г. Истра, пл. Революции, д. 7, оф. 2; *e-mail:* abosova@gmail.com

Филиппов Владимир Ильич, канд. пед. наук, ст. преподаватель кафедры методики преподавания технологии, информатики и информационно-коммуникационных технологий, Академия социального управления, г. Москва, Россия; *адрес:* 129344, Россия, г. Москва, ул. Енисейская, д. 3, корп. 5; *e-mail:* vf95@rambler.ru

L. L. Bosova

Moscow Pedagogical State University, Moscow, Russia

A. Yu. Bosova

Training and Methodical Centre, Istra, Moscow Region, Russia

V. I. Filippov

Academy of Social Management, Moscow, Russia

"PROGRAM, LEARN AND PLAY!". THE COURSE OF EXTRACURRICULAR ACTIVITIES FOR STUDENTS OF GRADES 3–6

Abstract

The article substantiates the relevance of early teaching in programming and describes a possible variant of such work with students in grades 3–6 as part of the course of extracurricular activities of general intellectual orientation. The potential of the course in achieving personal, metasubject and subject results by students is revealed. Thematic planning and content of teaching younger schoolchildren in the Scratch programming environment are presented. The main idea of the methodology for conducting extracurricular activities with junior schoolchildren is stated: working with ready-made examples according to the "use — change — addition — creation" scheme, which allows students to create digital products (multimedia postcards, interactive posters, games, simulators and quizzes).

Keywords: junior schoolchildren, programming, extracurricular activities, Scratch.

мультимедийных презентаций, сайтов, 3D-моделей; отдельные школьники имеют представление о современных языках программирования, приобрели опыт программирования мобильных приложений и создания объектов дополненной реальности. Такие школьники, не обладая системными и глубокими знаниями по перечисленным выше вопросам, тем не менее не настроены на освоение фундаментального школьного курса информатики, в результате чего не хотят и не могут понять основные заложенные в нем концепции, необходимые каждому человеку для комфортной жизни в условиях цифрового окружения и эффективного использования возможностей информационных технологий.

Выходом из сложившейся ситуации является организация широкомасштабной работы по обеспечению раннего старта в освоении школьной информатики, что может быть сделано в рамках курса внеурочной деятельности общеинтеллектуальной направленности «Программируем, учимся и играем!», основная идея которого заключается в том, чтобы своими руками создавать готовые к использованию продукты в среде программирования Скретч (*англ.* Scratch).

В среде Скретч обучающиеся в полной мере могут раскрыть свои творческие таланты, создавая мультфильмы, игры, анимированные открытки, презентации, обучающие программы, тренажеры, интерактивные тесты. Они могут придумывать различные объекты, определять, как эти объекты будут выглядеть в разных условиях, перемещать их по экрану, устанавливать способы взаимодействия между объектами; сочинять истории, рисовать и оживлять на экране своих придуманных персонажей, осваивая при этом технологии обработки графической и звуковой информации, анимационные технологии, мультимедийные технологии.

Педагогический потенциал среды программирования Скретч позволяет рассматривать ее как перспективный инструмент организации междисциплинарной проектной учебно-познавательной деятельности обучающегося, направленной на личностное и творческое развитие ребенка и позволяющей ему воссоздать единую картину мира, наводя мостики между различными изучаемыми в школе предметами.

Работая над проектами в Скретч, обучающиеся имеют возможность познакомиться с важными вычислительными концепциями, такими как повторения, условия, переменные, типы данных, события, процессы, и выразить себя в компьютерном творчестве. Выполняя коллективные проекты, ученики объединяются в группы, распределяя между собой роли программиста, сценариста, звукорежиссера, художника. Выбирая себе дело по душе, учащийся может более полно самореализоваться и, что не менее важно, актуализировать знания, полученные по «формальным» каналам.

Программирование в среде Скретч широко используется за рубежом [5, 6, 13]; с ним знакомы многие наши педагоги, включающие элементы работы в этой среде в курс информатики или ведущие внеурочные занятия со школьниками [2, 3, 7, 9, 10]; сеть Интернет изобилует многочисленными приглашениями в платные онлайн-школы и на курсы по работе в этой среде; с каждым годом все большую популярность у детей и подростков приобретает Международная Scratch-Олимпиада по кре-

ативному программированию (<https://robbo.ru/olymp/>). Все это доказывает эффективность идеологии Скретч-программирования и позволяет поставить вопрос о переходе к его массовому изучению в общеобразовательных организациях нашей страны. Одним из вариантов такой работы может быть организация внеурочной деятельности по представляемой в данной статье программе, разработанной на основе собственного многолетнего опыта работы со школьниками авторского коллектива и анализа отечественных и зарубежных разработок в этой области.

2. Цели и задачи изучения курса внеурочной деятельности «Программируем, учимся и играем!»

Курс «Программируем, учимся и играем!» нацелен на:

- развитие алгоритмического, логического и системного мышления обучающегося, формирование у него творческого подхода к решению задач;
- формирование культуры пользования информационными и коммуникационными технологиями, умений и навыков проектной и исследовательской деятельности;
- воспитание интереса к программированию как к ключевой технологии XXI века, воспитание стремления использовать полученные знания, умения и навыки в учебной деятельности и в повседневной жизни.

Основные задачи курса внеурочной деятельности «Программируем, учимся и играем!» — сформировать у обучающихся:

- понимание принципов устройства и функционирования объектов цифрового окружения периода цифровой трансформации современного общества;
- знания, умения и навыки грамотной постановки задач, возникающих в практической деятельности, для их решения с помощью информационных технологий;
- знание основных алгоритмических структур и умение применять эти знания при создании цифровых продуктов;
- эффективные приемы работы с мультимедийной информацией;
- умения и навыки совместной деятельности и сетевой коммуникации;
- умения и навыки проектирования, разработки и презентации цифровых продуктов;
- знание базовых норм информационной этики и права, основ информационной безопасности.

3. Место курса «Программируем, учимся и играем!» в плане внеурочной деятельности

Освоение курса внеурочной деятельности «Программируем, учимся и играем!» предусмотрено последовательно в следующих классах:

- III класс. Модуль 1. «Первые шаги в мире Скретч» — 34 часа;

- IV класс. Модуль 2. «Азы программирования в Скретч» — 34 часа;
- V класс. Модуль 3. «Навыки программирования в Скретч» — 34 часа;
- VI класс. Модуль 4. «Разработка цифровых продуктов» — 34 часа.

Программа курса может быть уплотнена за счет перераспределения содержания при начале его освоения с IV или с V класса.

По завершении освоения каждого из четырех модулей обучающийся получает сертификат, определяющий достигнутый им уровень мастерства:

- «Стажер» (модуль 1);
- «Любитель» (модуль 2);
- «Специалист» (модуль 3);
- «Гуру» (модуль 4).

4. Планируемые результаты освоения курса внеурочной деятельности

Анализ требований ФГОС НОО [11], ФГОС ООО [12], примерной рабочей программы по учебному предмету «Информатика» для уровня основного общего образования [8] позволяет определить потенциал курса внеурочной деятельности «Программируем, учимся и играем!» в достижении обучающимися следующих личностных, метапредметных и предметных результатов.

4.1. Личностные результаты

Личностные результаты — результаты, имеющие направленность на решение задач воспитания, развития и социализации обучающихся средствами курса внеурочной деятельности.

Гражданско-патриотическое воспитание:

- ценностное отношение к своей Родине — России.

Духовно-нравственное воспитание:

- осознание социальных норм и правил межличностных отношений в коллективе, в том числе в социальных сообществах;
- осознание необходимости совместной деятельности при выполнении учебных, познавательных задач, создании учебных проектов; стремление к взаимопониманию и взаимопомощи в процессе этой учебной деятельности;
- стремление оценивать свое поведение и поступки, а также поведение и поступки других людей с позиции нравственных и правовых норм с учетом осознания последствий поступков; активное неприятие асоциальных поступков, в том числе в сети Интернет.

Эстетическое воспитание:

- восприимчивость к разным видам искусства;
- стремление к самовыражению в разных видах художественной деятельности.

Формирование культуры здоровья:

- соблюдение правил здорового и безопасного (для себя и других людей) образа жизни в окружающей среде (в том числе информационной).

Трудовое воспитание:

- интерес к практическому изучению профессий и труда в сферах профессиональной деятельности,

связанных с программированием и информационными технологиями.

Экологическое воспитание:

- бережное отношение к природе.

Ценности научного познания:

- первоначальные мировоззренческие представления об информации, информационных процессах и информационных технологиях;
- интерес к обучению и познанию; любознательность; стремление к самообразованию, проектной деятельности;
- сформированность основ информационной культуры.

4.2. Метапредметные результаты

Метапредметные результаты — результаты, отражающие овладение универсальными учебными действиями — познавательными, коммуникативными, регулятивными.

Универсальные учебные познавательные действия.

Базовые логические действия:

- умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, устанавливать причинно-следственные связи, строить логические рассуждения, выявлять недостатки информации для решения поставленной задачи;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач.

Базовые исследовательские действия:

- формулировать вопросы, фиксирующие разрыв между реальным и желательным состоянием ситуации, объекта, и самостоятельно устанавливать искомое и данное;
- оценивать на применимость и достоверность информацию, полученную в ходе исследования;
- прогнозировать возможное дальнейшее развитие процессов, событий и их последствия в аналогичных или сходных ситуациях.

Работа с информацией:

- выбирать источник получения информации;
- применять различные методы, инструменты и запросы при поиске и отборе информации или данных из источников с учетом предложенной учебной задачи и заданных критериев;
- соблюдать правила информационной безопасности при поиске информации в сети Интернет;
- оценивать надежность информации по критериям, предложенным учителем;
- анализировать и создавать текстовую, видео-, графическую, звуковую информацию в соответствии с поставленной задачей.

Универсальные учебные коммуникативные действия.

Общение:

- сопоставлять свои суждения с суждениями других участников диалога, обнаруживать различие и сходство позиций;
- публично представлять результаты выполненного проекта.

Совместная деятельность (сотрудничество):

- понимать и использовать преимущества командной и индивидуальной работы при решении конкретной проблемы, в том числе при создании цифрового продукта;
- принимать цель совместной информационной деятельности, коллективно строить действия по ее достижению: распределять роли, договариваться, обсуждать процесс и результат совместной работы;
- выполнять свою часть работы, достигая качественного результата по своему направлению и координируя свои действия с другими членами команды;
- оценивать качество своего вклада в общий результат.

Универсальные учебные регулятивные действия.**Самоорганизация:**

- выявлять в жизненных и учебных ситуациях проблемы, требующие решения;
- самостоятельно составлять алгоритм решения задачи (или его часть), выбирать способ решения учебной задачи с учетом имеющихся ресурсов и собственных возможностей, аргументировать предлагаемые варианты решений;
- составлять план действий (план реализации намеченного алгоритма решения), корректировать предложенный алгоритм с учетом получения новых знаний об изучаемом объекте;
- делать выбор в условиях противоречивой информации и брать на себя ответственность за решение.

Самоконтроль (рефлексия):

- давать адекватную оценку ситуации и предлагать план ее изменения;
- учитывать контекст и предвидеть трудности, которые могут возникнуть при решении учебной задачи, адаптировать решение к меняющимся обстоятельствам;
- объяснять причины достижения (недостижения) результатов информационной деятельности, давать оценку приобретенному опыту, уметь находить позитивное в произошедшей ситуации;
- оценивать соответствие результата цели и условиям.

4.3. Предметные результаты

Предметные результаты — результаты, отражающие сформированность у обучающихся умений:

- пояснять назначение основных устройств компьютера;
- работать с файловой системой персонального компьютера с использованием графического интерфейса, а именно: создавать, копировать,

перемещать, переименовывать и удалять файлы и каталоги;

- соблюдать требования безопасной эксплуатации технических средств информационно-коммуникационных технологий;
- соблюдать сетевой этикет, базовые нормы информационной этики и права при работе с приложениями на любых устройствах и в сети Интернет;
- искать информацию в сети Интернет (в том числе по ключевым словам, по изображению), критически относиться к найденной информации, осознавая опасность для личности и общества распространения вредоносной информации, в том числе экстремистского и террористического характера;
- распознавать попытки и предупреждать вовлечение себя и окружающих в деструктивные и криминальные формы сетевой активности (в том числе кибербуллинг, фишинг);
- использовать современные интернет-сервисы (в том числе коммуникационные сервисы, облачные хранилища данных, онлайн-программы (среды разработки)) в учебной и повседневной деятельности;
- пояснять на примерах смысл понятий «информация», «алгоритм», «исполнитель», «объект», «событие», «управление», «обработка событий»;
- пояснять назначение базовых алгоритмических конструкций (следование, ветвление, цикл);
- осуществлять разработку, тестирование и отладку несложных программ;
- использовать переменные и списки в среде программирования Скретч;
- использовать при разработке программ логические значения, операции и выражения с ними;
- выполнять построение правильных многоугольников и композиций из правильных многоугольников в среде программирования Скретч;
- разбивать задачи на подзадачи; составлять и выполнять в среде программирования Скретч несложные алгоритмы с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителями;
- пояснять на примере понятия проекта и этапов его разработки;
- разрабатывать проекты: интерактивные истории, мультимедийные открытки, интерактивные игры, мультфильмы, интерактивные плакаты и викторины;
- размещать в сети проекты, созданные в среде программирования Скретч;
- сотрудничать при разработке проектов в среде программирования Скретч.

5. Тематическое планирование

№ п/п	Тема	Кол-во часов
1	Модуль 1. «Первые шаги в мире Скретч»	
1.1	Компьютер и интернет	3

№ п/п	Тема	Кол-во часов
1.2	Знакомство с графическим редактором Скретч	2
1.3	Создание мультимедийной открытки	4
1.4	Как думают и говорят спрайты	4
1.5	Планирование последовательности действий	3
1.6	Компьютерная игра — своими руками	4
1.7	Интерактивный плакат	3
1.8	Взаимодействие объектов	4
1.9	Движение и рисование. Инструмент <i>Перо</i>	3
1.10	Презентация проектов, выполненных в рамках модуля	1
1.11	Резерв учебного времени	3
	Итого:	34 часа
2	Модуль 2. «Азы программирования в Скретч»	
2.1	Повторение	2
2.2	Циклы	4
2.3	Переменные	3
2.4	Механика движения	3
2.5	Координаты	4
2.6	Спрайты обучаются	2
2.7	Ветвления	4
2.8	Диалоги и списки	4
2.9	Тренажеры и викторины	4
2.10	Презентация проектов, выполненных в рамках модуля	1
2.11	Резерв учебного времени	3
	Итого:	34 часа
3	Модуль 3. «Навыки программирования в Скретч»	
3.1	Повторение	3
3.2	Черепашья графика. Процедуры с параметрами	5
3.3	Создание и использование собственных фонов и спрайтов	6
3.4	Музыка и звукозапись	3
3.5	Дублирование и клонирование	2
3.6	Управление временем. Таймер	2
3.7	Включаем таймер. Игры на время	3
3.8	Лабиринты	2
3.9	Многоуровневые игры. Командная работа над проектом	4
3.10	Презентация проектов, выполненных в рамках модуля	1
3.11	Резерв учебного времени	3
	Итого:	34 часа

№ п/п	Тема	Кол-во часов
4	Модуль 4. «Разработка цифровых продуктов»	
4.1	Повторение	3
4.2	Ремиксы популярных игр	9
4.3	Рекурсия	2
4.4	Параллельные алгоритмы. Проект «Стройка»	4
4.5	Многоуровневые квесты. Командная работа над проектом	4
4.6	Дополненная реальность	2
4.7	Создание приложений в App Inventor	6
4.8	Презентация проектов, выполненных в рамках модуля	1
4.9	Резерв учебного времени	3
	Итого:	34 часа

6. Содержание программы

6.1. Модуль 1. Первые шаги в мире Скретч

1. Компьютер и интернет (3 часа).

Основные устройства компьютера. Правила работы и поведения в компьютерном классе.

Исполнители. Команды. Программы. Скретч — среда программирования, в которой программа собирается из блоков, как из кубиков LEGO.

Режимы работы в среде Скретч: 1) работа непосредственно в интернете, на сайте Скретч: <https://scratch.mit.edu/> (режим «онлайн»); 2) работа со средой Скретч, загруженной со страницы <https://scratch.mit.edu/scratch2download> и установленной на компьютере, без подключения к интернету (режим «офлайн»). Запуск среды программирования Скретч (офлайн).

Спрайт — графический объект, выполняющий команды; его действиями управляет программа, которая может состоять из одной или нескольких самостоятельных частей. Скрипт — самостоятельная часть программы. Сцена — среда, в которой действуют спрайты.

Знакомство с интерфейсом программы. Группы команд. Блок — графическое изображение команды в Скретч. Кнопки *Старт* и *Стоп*. Фон и костюм. Библиотека фонов и костюмов.

Создание первой анимации по образцу. Выбор фона. Выбор спрайта. Простая анимация движения спрайта «Запускаем котика в космос». Команды «плыть ... секунд к случайное положение» (группа *Движение*), «следующий костюм» (группа *Внешний вид*), «когда флажок нажат» (группа *События*), «повторять всегда» (группа *Управление*), «включить звук» (группа *Звук*). Файл. Имя файла. Сохранение созданной анимации в личной папке.

Анализ анимации «Запускаем котика в космос» по плану:

- 1) Кто? Что? (Какие спрайты (объекты, персонажи, герои, исполнители) задействованы в анимации?)
- 2) Где? (Как выглядит сцена, на которой разворачивается действие: какой выбран фон? изменяется ли фон? предусмотрено ли звуковое сопровождение?)

- 3) Как? (Что делает каждый из спрайтов на сцене? Взаимодействуют ли спрайты между собой?)

Понятие о сценарном плане анимации.

Проект «Морские обитатели». Создание простой анимации «Морские обитатели» по заданному сценарному плану. Сохранение созданной анимации в личной папке.

Интернет. Безопасность в сети Интернет. Запуск среды программирования Скретч (онлайн). Регистрация аккаунтов.

Разработка сценарного плана простой анимации по собственному замыслу. Создание анимации по разработанному сценарному плану. Сохранение анимации, созданной по собственному замыслу, в личной папке.

2. Знакомство с графическим редактором Скретч (2 часа).

Графический редактор — компьютерное приложение для создания и редактирования (изменения) изображений на экране компьютера. Растровые и векторные графические редакторы.

Векторное изображение как совокупность линий и фигур. Знакомство с инструментами графического редактора: векторный режим.

Растровое изображение как совокупность разноцветных точек. Знакомство с инструментами графического редактора: растровый режим.

Создание фона. Редактирование фона.

Редактирование костюма. Центр костюма. Создание костюма.

Команды для смены внешности. Команды «следующий костюм», «следующий фон» (группа *Внешний вид*).

Анимация со сменой фонов по заданному сценарному плану. Сохранение созданной анимации в личной папке.

Анимация со сменой костюмов по заданному сценарному плану. Сохранение созданной анимации в личной папке.

3. Создание мультимедийной открытки (4 часа).

Исследование возможностей изменения костюма.

Команды «установить размер», «изменить размер на», «установить эффект», «изменить эффект», «убрать

графические эффекты», «показаться», «спрятаться» (группа *Внешний вид*).

Создание мультимедийной открытки по образцу. Сохранение созданной мультимедийной открытки в личной папке.

Анализ сценарного плана мультимедийной открытки.

Проект «Живое имя». Создание проекта по заданному сценарному плану. Сохранение созданного проекта в личной папке.

Разработка сценарного плана мультимедийной открытки по собственному замыслу. Создание мультимедийной открытки по разработанному сценарному плану. Сохранение мультимедийной открытки, созданной по собственному замыслу, в личной папке.

4. Как думают и говорят спрайты (4 часа).

Команды «говорить», «сказать», «думать» (группа *Внешний вид*).

Расширение *Текст в речь*. Команды «установить язык», «установить голос», «сказать».

Проект «Гобо читает стихотворение». Разработка сценарного плана, создание и сохранение созданного проекта в личной папке.

Команда «спросить и ждать» (группа *Сенсоры*). Планирование последовательности высказываний.

Проект «Диалог двух героев». Разработка сценарного плана, создание и сохранение созданного проекта в личной папке.

5. Планирование последовательности действий (3 часа).

Алгоритм. Базовые алгоритмические конструкции. Следование.

Онлайн-практикум «Классический лабиринт» (<https://studio.code.org/hoc/1>).

Команды «идти ... шагов», «перейти на», «плыть секунд к», «вернуться к» (группа *Движение*). Изменение скорости передвижения.

Команда «ждать» (группа *Управление*).

Проект «Ожившая история (сказка)». Разработка сценарного плана, создание и сохранение созданного проекта в личной папке.

6. Компьютерная игра — своими руками (4 часа).

Управление спрайтом с помощью клавиш (вверх, вниз, влево, вправо).

Событие — сигнал, по которому запускаются определенные скрипты. Стандартные (системные) события: нажатие на зеленый флажок, на клавишу.

Команда «когда клавиша нажата» (группа *События*).

Создание игры «Догонит ли кошка мышку?» по образцу.

Анализ сценарного плана игры «Догонит ли кошка мышку?»

Разработка сценарного плана аналогичной игры с другими персонажами.

Создание игры по разработанному сценарному плану. Сохранение созданной игры в личной папке.

7. Интерактивный плакат (3 часа).

Команда «когда спрайт нажат» (группа *События*).

Анимация спрайта в результате щелчка по нему мышью: спрайт говорит или воспроизводит звук, меняет

внешний вид (цвет, размер), исчезает, к спрайту применяется выбранный эффект.

Управление перемещением спрайта нажатием клавиш.

Создание интерактивного плаката «Красная площадь» по образцу.

Анализ сценарного плана плаката «Красная площадь».

Разработка сценарного плана интерактивного плаката по собственному замыслу. Поиск информации в сети Интернет. Создание интерактивного плаката по разработанному сценарному плану. Сохранение созданного интерактивного плаката в личной папке.

8. Взаимодействие объектов (4 часа).

Команды «передать», «передать и ждать», «когда я получу» (группа *События*). Диалог между спрайтами: после своей реплики спрайт передает сообщение второму спрайту и т. д.

Ветвление. Выбор той или иной последовательности действий в зависимости от выполнения заданного условия. Примеры ситуаций выбора в жизни.

Команды «если ... то», «повторять всегда» (группа *Управление*).

Команды «касается», «касается цвета», «цвет касается цвета» (группа *Сенсоры*).

Взаимодействие двух спрайтов. Обработка касания спрайтов.

Создание игры «Берегись голодной акулы!» по образцу.

Анализ сценарного плана игры «Берегись голодной акулы!»

Разработка сценарного плана аналогичной игры с другими персонажами.

Создание игры по разработанному сценарному плану. Сохранение созданной игры в личной папке.

9. Движение и рисование. Инструмент *Перо* (3 часа).

Расширение *Перо*. Команды «стереть все», «печать», «опустить перо», «поднять перо», «установить для пера цвет», «изменить (цвет, насыщенность, яркость, прозрачность) пера на», «установить (цвет, насыщенность, яркость, прозрачность) пера», «изменить размер пера на», «установить цвет пера» (группа *Перо*). Настройка линий при рисовании.

Линейные алгоритмы. Программа рисования для спрайта.

Базовая программа рисования круга. Рисунки «Радужные круги», «Мишень», «Светофор». Композиция из кругов по собственному замыслу. Сохранение созданных рисунков и композиций в личной папке.

Бесконечный цикл. Команда «повторять всегда» (группа *Управление*). Команда «перейти на (случайное положение, указатель мыши)» (группа *Движение*).

Рисунки «Разноцветные линии», «Разноцветный клубок».

Команда «идти ... шагов» (группа *Движение*). Базовая программа рисования линии. Рисунки из линий «Пирамидка», «Штанга», «Стадион».

Композиция из линий по собственному замыслу.

Повороты. Команды «вернуть в направлении», «вернуть по часовой стрелке», «вернуть против

часовой стрелки» (группа *Движение*). Базовая программа рисования квадрата. Рисунки из квадратов и прямоугольников. Сохранение созданных рисунков и композиций в личной папке.

10. Презентация проектов (1 час).

Презентация проектов, выполненных обучающимися в рамках занятий по модулю.

Предметные результаты освоения модуля 1.

Учащиеся будут иметь представление:

- об основных устройствах компьютера;
- о правилах работы за компьютером;
- о правилах безопасной работы в сети Интернет;
- о назначении среды программирования Скретч и об основных элементах ее интерфейса;
- об алгоритме и исполнителях;
- о сценарном плане;
- о программном коде и составляющих его командах;
- о правилах именования и сохранения документа;
- об объектах авторского права в сети Интернет.

Учащиеся будут уметь:

- запускать среду программирования Скретч офлайн;
- выбирать спрайты и фоны из библиотек среды программирования Скретч;
- создавать и редактировать свои спрайты и фоны в графическом редакторе;
- разрабатывать сценарные планы и создавать на их основе анимации, мультимедийные открытки, интерактивные плакаты и простые игры в программной среде Скретч.

6.2. Модуль 2. Азы программирования в Скретч

1. Повторение (2 часа).

Правила работы и поведения в компьютерном классе.

Интернет. Безопасность в сети Интернет.

Онлайн-практикум «Час кода с Minecraft» (<https://studio.code.org/s/mc/>).

Запуск среды программирования Скретч (онлайн). Регистрация учетной записи (аккаунта).

Работа по совершенствованию любого из своих проектов, созданных в рамках модуля 1.

Сохранение проекта в разделе «Мои работы», публикация проекта.

2. Циклы (4 часа).

Цикл — многократное выполнение группы команд. Циклические алгоритмы. Команды «повторять всегда», «повторять раз» (группа *Управление*).

Спрайт-художник. Команды движения и рисования.

Рисование пунктирной линии. Рисование квадрата. Рисование равностороннего треугольника. Рисование правильного пятиугольника. Рисование правильного шестиугольника.

Орнамент. Виды орнаментов. Технология создания геометрического орнамента в Скретч (определение исходной позиции, создание повторяющегося фрагмента, переход на исходную позицию). Ряд одинаковых квадратов. Ряд одинаковых правильных многоугольников.

Проект «Геометрический орнамент». Создание геометрического орнамента по собственному замыслу. Сохранение проекта в разделе «Мои работы».

Последовательные и одновременные действия исполнителей. Параллельные алгоритмы.

Проект «Олимпийские кольца». Одинаковые действия исполнителей. Дублирование спрайтов. Сохранение проекта в разделе «Мои работы».

3. Переменные (3 часа).

Переменная — ячейка памяти, имеющая имя и значение. Имя переменной. Создание переменной. Команды «задать значение», «изменить на», «показать переменную», «скрыть переменную» (группа *Переменные*).

Создание игры с подсчетом очков «Сможет ли призрак сыграть в мяч?» по образцу.

Анализ сценарного плана игры «Сможет ли призрак сыграть в мяч?»

Разработка сценарного плана аналогичной игры с другими персонажами. Создание аналогичной игры по разработанному сценарному плану. Сохранение проекта в разделе «Мои работы».

4. Механика движения (3 часа).

Команды «смена костюма» (группа *Внешность*), «идти ... шагов», «если касается края оттолкнуться», «установить способ вращения» (группа *Движение*).

Движение по сцене спрайта «Балерина».

Движение Кота по сцене.

Работа в графическом редакторе (векторный режим). Создание новых костюмов по дополнительным фазам движения. Сохранение (экспорт) спрайта с дополнительными костюмами в личную папку.

Программирование реалистичного движения спрайта по собственному выбору. Сохранение проекта в разделе «Мои работы».

5. Координаты (4 часа).

Координаты — числа, определяющие положение точки на сцене.

Система координат в Скретч.

Команды «изменить x на», «изменить y на», «установить x в», «установить y в», «перейти в x, y», «плыть секунд в точку x, y» (группа *Движение*).

Создание игры с использованием координат «Любят ли ежики мячики?» по образцу.

Анализ сценарного плана игры «Любят ли ежики мячики?»

Разработка сценарного плана аналогичной игры с другими персонажами. Создание аналогичной игры по разработанному сценарному плану. Сохранение проекта в разделе «Мои работы».

Создание геометрического орнамента по собственному замыслу с использованием координат. Сохранение проекта в разделе «Мои работы».

Вложенные циклы. Проект «Дизайн ткани». Сохранение проекта в разделе «Мои работы».

6. Спрайты обучаются (2 часа).

Разбиение задачи на подзадачи. Вспомогательные алгоритмы.

Создание собственных блоков. Блоки для изображения цифр «0», «1» и «2». Мини-проект «Год 2021 (2022)».

Проект «Мой почтовый индекс». Сохранение проекта в разделе «Мои работы».

7. Ветвления (4 часа).

Алгоритмы с ветвлениями.

Команды «если — то», «если — то — иначе» (группа *Управление*), «клавиша нажата», «мышь нажата» (группа *Сенсоры*), «когда я получу сообщение», «передать сообщение» (группа *События*).

Проект «Времена года». Смена фонов сцены при передаче-получении сообщений. Сохранение проекта в разделе «Мои работы».

Управление объектами. Управление движением персонажа с помощью мыши. Управление движением с помощью клавиш.

Создание игры «Постреляем по тарелочкам?» по образцу. Сохранение проекта в разделе «Мои работы».

Анализ сценарного плана игры «Постреляем по тарелочкам?»

Разработка сценарного плана аналогичной игры с другими персонажами. Создание аналогичной игры по разработанному сценарному плану. Сохранение проекта в разделе «Мои работы».

8. Диалоги и списки (4 часа).

Команды «говорить», «сказать», «думать» (группа *Внешний вид*), «спросить и ждать», «ответ» (группа *Сенсоры*), «установить язык», «установить голос», «сказать» (группа *Текст в речь*), «перевести на» (группа *Перевести*).

Создание программы-переводчика по образцу.

Анализ сценарного плана программы-переводчика.

Разработка сценарного плана аналогичной программы. Создание аналогичной программы по разработанному сценарному плану. Сохранение проекта в разделе «Мои работы».

Списки.

Создание программы «Пообщаемся с чат-ботом?» по образцу. Сохранение проекта в разделе «Мои работы».

Анализ сценарного плана игры «Пообщаемся с чат-ботом?»

Разработка сценарного плана аналогичной игры с другими персонажами. Создание аналогичной игры по разработанному сценарному плану. Сохранение проекта в разделе «Мои работы».

Создание аналогичной программы по собственному замыслу.

9. Тренажеры и викторины (4 часа).

Случайные числа.

Обсуждение сценарного плана тренажера устного счета. Создание тренажера устного счета. Сохранение проекта в разделе «Мои работы».

Правила создания викторин. Создание викторины по образцу. Сохранение проекта в разделе «Мои работы».

Анализ сценарного плана викторины.

Создание викторины по разработанному сценарному плану. Сохранение проекта в разделе «Мои работы».

10. Презентация проектов (1 час).

Презентация проектов, выполненных обучающимися в рамках занятий по модулю.

Предметные результаты освоения модуля 2.

Учащиеся будут иметь представление:

- о правилах работы за компьютером;
- о правилах безопасной работы в сети Интернет;
- о назначении среды программирования Скретч и об основных элементах ее интерфейса;
- об основных правилах работы в сети и на сайте: <https://scratch.mit.edu>;
- о базовых алгоритмических конструкциях (ветвления и циклы) и их реализации в среде программирования Скретч;
- об этапах разработки программы (проекта в среде программирования Скретч): постановка задачи, разработка сценарного плана, алгоритмизация, кодирование, тестирование, отладка.

Учащиеся будут уметь:

- запускать среду программирования Скретч онлайн;
- использовать переменные и списки;
- работать с координатами и случайными числами;
- создавать вспомогательные алгоритмы;
- использовать ветвления и циклы различного вида;
- создавать и редактировать свои спрайты в графическом редакторе;
- разрабатывать сценарный план анимации, игры, тренажера, викторины;
- создавать анимации, игры, тренажеры и викторины в среде программирования Скретч.

6.3. Модуль 3. Навыки программирования в Скретч

1. Повторение (3 часа).

Правила работы и поведения в компьютерном классе.

Интернет. Безопасность в сети Интернет.

Запуск среды программирования Скретч (онлайн).

Игра «Пинг-понг». Изучение инструкции. Пошаговое выполнение инструкции.

Сохранение проекта в разделе «Мои работы».

Анализ сценарного плана игры «Пинг-понг». Приемы, используемые при ее создании.

Игра «Угадайка». Обсуждение сюжета игры. Самостоятельная работа по созданию игры «Угадайка».

2. Черепаха графика. Процедуры с параметрами (5 часов).

История Черепахи.

Добавление внешнего спрайта. Спрайт Черепаха, костюмы Черепахи. Начальное положение. Смена направления. Черепаха и *Перо*. Система команд Черепахи.

Линейные алгоритмы. Черепаха рисует по координатам.

Проект «Мой рисунок по координатам». Подготовка эскиза рисунка на бумаге. Программирование рисунка. Сохранение проекта в разделе «Мои работы».

Циклические алгоритмы. Циклические алгоритмы с использованием координат.

Обучение Черепахи. Процедуры. Процедуры с параметрами. Процедура «Многоугольник» с параметрами. Организация ввода данных (параметров) в диалоге. Вложенные многоугольники. Концентрические многоугольники.

Проект «Мои многоугольники». Сохранение проекта в разделе «Мои работы».

Арт-проект «Центрический орнамент». Сохранение проекта в разделе «Мои работы». Публикация проекта. Комментирование арт-проектов одноклассников.

Онлайн-практикум «Программируй с Анной и Эльзой» (<https://studio.code.org/s/frozen/>).

3. Создание и использование собственных фонов и спрайтов (6 часов).

Информация в сети Интернет и правила ее использования.

Основные приемы создания и редактирования изображений в графическом редакторе Gimp. Выделение фрагментов. Работа с инструментом *Лассо*.

Проект «Буктрейлер». Буктрейлер — видеоряд, представляющий книгу. Виды буктрейлеров и их содержание.

Выбор литературного произведения. Разработка сценарного плана буктрейлера. Создание буктрейлера по литературному произведению на основе иллюстраций из этого произведения (фон и спрайты), спрайтов и музыки из библиотек, самостоятельно созданных текстовых спрайтов.

Публикация проекта. Знакомство с проектами одноклассников и их комментирование.

4. Музыка и звукозапись (3 часа).

Расширение *Музыка*. Команды «барабану играть», «пауза в течение бит», «играй ноту», «изменить инструмент на», «установить темп», «изменить темп».

Вкладка *Звуки*. Библиотека звуков. Выбор звуков из библиотеки, запись звуков, загрузка звуков. Команды «играть звук до конца», «включить звук», «остановить все звуки», «изменить эффект», «установить эффект», «убрать звуковые эффекты», «изменить громкость», «установить громкость» (группа *Звук*).

Проект «Создай музыку». Изучение инструкции. Пошаговое выполнение инструкции. Сохранение проекта в разделе «Мои работы».

Проект «Музыка чисел». Создание скрипта, проигрывающего гамму в пределах первой октавы на основе линейного алгоритма. Создание по готовой нотной записи скрипта, проигрывающего простое музыкальное произведение.

Проект «Басни Крылова». Беседа о баснях И. А. Крылова. Обсуждение возможного сценарного плана визуализации и озвучки басни. Работа в парах по подготовке фона, спрайтов и звуковых фрагментов для визуализации и озвучки выбранной басни. Публикация проекта.

5. Дублирование и клонирование (2 часа).

Дублирование спрайтов. Дубль — копия спрайта (вместе с его скриптами), созданная до работы программы. Проект «Снегопад-1». Просмотр видеоролика. Самостоятельное создание проекта «Снегопад-1» по образцу.

Клонирование — создание копии спрайта в процессе выполнения скрипта. Клонирование и дублирование. Команды «создать клон», «удалить клон», «когда я начинаю как клон» (группа *Управление*).

Проект «Снегопад-2». Просмотр видеоролика. Самостоятельное создание проекта «Снегопад-2» по образцу.

Создание проектов «Листопад», «Дождик», «Шарики» по аналогии.

Сборка проекта «Времена года» на основе проектов «Снегопад», «Дождик», «Шарики», «Листопад». Добавление музыки. Создание титульного фона. Публикация проекта.

6. Управление временем. Таймер (2 часа).

Таймер и особенности его работы (начало работы с момента включения программы, невозможность остановки или паузы). Команда «сбросить таймер» (группа *Сенсоры*).

Команды со встроенным таймером.

Планирование последовательности событий. Использование таймера для запуска определенных скриптов через заданное число секунд. Проект «Солнечные сутки Земли». Обсуждение и реализация сценарного плана проекта. Публикация проекта.

7. Включаем таймер. Игры на время (3 часа).

Использование таймера в играх: сброс-запуск таймера на каждом новом уровне игры, этапе викторины для фиксации времени; ограничение времени, отводимого на игру или ее этап.

Игры с таймером.

Игра «Шерлок Холмс»: за ограниченное время игрок должен найти в комнате как можно больше определенных предметов. Обсуждение и реализация сценарного плана проекта. Публикация проекта.

Игра «Собери урожай»: за ограниченное время герой под управлением игрока должен собрать как можно больше яблок. Обсуждение и реализация сценарного плана проекта. Публикация проекта.

Игра по собственному замыслу с использованием таймера и клонов. Публикация проекта.

8. Лабиринты (2 часа).

Лабиринт. Игра «Яблоко раздора»: два героя под управлением игроков пытаются добраться до яблока в центре лабиринта. Обсуждение и реализация сценарного плана проекта. Публикация проекта.

Игра «Выход из лабиринта». «Правило одной руки» для прохождения лабиринта: двигаясь по лабиринту, надо все время касаться правой или левой рукой его стены. Использование «правила одной руки» для программирования героя, ищущего выход из лабиринта. Обсуждение и реализация сценарного плана проекта. Публикация проекта.

Игра «Похитители сокровищ». Лабиринт с дополнительными препятствиями. Обсуждение и реализация сценарного плана проекта. Публикация проекта.

9. Многоуровневые игры. Командная работа над проектом (4 часа).

Рюкзак Скретч — инструмент для переноса кода, спрайтов, костюмов, фонов, музыки в другие проекты.

Создание многоуровневой игры «Лабиринт». Распределение работ в команде.

Публикация проекта.

10. Презентация проектов (1 час).

Презентация проектов, выполненных обучающимися в рамках занятий по модулю.

Предметные результаты освоения модуля 3.

Учащиеся будут иметь представление:

- о правилах работы за компьютером;
- о правилах безопасной работы в сети Интернет;
- о назначении среды программирования Скретч и об основных элементах ее интерфейса;
- об основных правилах работы в сети и на сайте: <https://scratch.mit.edu>;
- о базовых алгоритмических конструкциях (ветвления и циклы) и их реализации в среде программирования Скретч;
- о возможностях дублирования и клонирования спрайтов;
- об этапах разработки программы (проекта в среде программирования Скретч): постановка задачи, разработка сценарного плана, алгоритмизация, кодирование, тестирование, отладка;
- о возможностях коллективной работы над проектом.

Учащиеся будут уметь:

- использовать переменные и списки;
- работать с координатами и случайными числами;
- создавать вспомогательные алгоритмы;
- использовать ветвления и циклы различного вида;
- пользоваться таймером;
- делать звукозапись и использовать музыкальные возможности Скретч;
- создавать дубли и клоны;
- создавать и редактировать свои спрайты в графическом редакторе;
- разрабатывать сценарный план анимации, игры, тренажера, викторины;
- создавать анимации, игры, тренажеры и викторины в среде программирования Скретч;
- строить правильные многоугольники и композиции из них с использованием циклических алгоритмов и процедур с параметрами;
- осуществлять командную разработку проектов.

6.4. Модуль 4. Разработка цифровых продуктов

1. Повторение (3 часа).

Правила работы и поведения в компьютерном классе. Интернет. Безопасность в сети Интернет.

Ремикс — модифицированная и опубликованная версия существующего проекта.

Лицензионное соглашение, регулирующее обмен материалами внутри Скретч-сообщества.

Анализ игры «Охотник за золотом». Создание ремикса игры.

Знакомство с проектом «Раскраска» и его доработка.

Знакомство с проектом «Составные условия» и разработка на его основе программы рисования собственного рисунка на координатной плоскости.

Публикация созданных проектов.

2. Ремиксы популярных игр (9 часов).

Компьютерные игры. История компьютерных игр. Классификация компьютерных игр. Основные составляющие компьютерной игры. Игровая зависимость и пути ее предупреждения.

Игры «Морской бой», «Змейка», «Марио», «Ну, погоди!», «Гонки». Знакомство с историями их создания, сюжетами и алгоритмами реализации.

Программирование одной или нескольких игр в среде Скретч. Тестирование и отладка игр(ы). Доработка игр(ы).

Публикация созданных проектов.

Портирование на Скретч.

3. Рекурсия (2 часа).

Рекурсия как процесс повторения элементов самоподобным образом. Примеры рекурсии. Фракталы. Снежинка Коха. Треугольник Серпинского.

Рекурсия в Скретч. Разработка функции, которая запускает сама себя с разными параметрами. Программирование рекурсии.

Публикация проекта.

4. Параллельные алгоритмы. Проект «Стройка» (4 часа).

Алгоритмы и исполнители. Последовательное исполнение алгоритма.

Планирование работы нескольких исполнителей. Параллельный алгоритм — алгоритм, который может быть реализован по частям на множестве различных исполнителей (вычислительных устройств) с последующим объединением полученных результатов и получением корректного результата.

Зависимость скорости работы от количества исполнителей; критический путь; оптимальное количество исполнителей; оптимальный порядок действий исполнителей, конвейерная обработка данных.

Исполнитель «Директор строительства» и решаемые им задачи.

Разработка параллельного алгоритма и его визуализация в Скретч. Пример реализации параллельного алгоритма в Скретч.

Публикация проекта.

5. Многоуровневые квесты. Командная работа над проектом (4 часа).

Квест как интерактивная история, в которой герой, управляемый игроком, исследует мир, решая головоломки и задачи и переходя с одного уровня игры на другой.

Этапы разработки квеста:

- 1) выбор сюжетной линии, главного героя, цели и названия игры;
- 2) разработка эскизов локаций для каждого уровня игры;
- 3) подбор заданий для каждого уровня игры;
- 4) создание эскизов стартового и финального экранов игры;
- 5) реализация сценария (замысла) квеста в среде программирования.

Рассмотрение примера разработки сценария.

Групповая работа над проектом. Мозговой штурм для выбора сюжетной линии, главного героя, цели и названия игры. Распределение ролей в группе:

- дизайнер — работа над стартовым и финальным экраном;
- художник — подготовка фонов-локаций и спрайта-героя;
- логик — подбор задач и головоломок;
- программист — написание скриптов;

- звукорежиссер — обеспечение музыкального сопровождения;
 - руководитель — координация работы и т. д.
- Публикация созданных проектов.

6. Дополненная реальность (2 часа).

Дополненная реальность как способ соединения реального и компьютерного миров. Примеры дополненной реальности.

Расширение *Видео распознавание*. Команды «включить видео», «установить прозрачность видео», «когда скорость видео». Исследование команд.

Создание мини-игр «Лопни шарик» и «Прогони шарик».

Публикация созданных проектов.

7. Создание приложений в App Inventor (6 часов).

Основы создания программ для мобильных устройств.

Введение в среду программирования приложений для мобильных устройств MIT App Inventor. Основные структурные блоки программирования. Принципы разработки мобильных приложений. Интерфейс программной среды MIT App Inventor. Режимы *Дизайн* и *Блоки*. Основные компоненты среды программирования. Свойства компонент. Блоки программирования в среде Blockly. Сохранение и установка приложений на мобильные устройства. Разработка приложения «Игральный кубик».

Основные компоненты приложения. Дизайн приложения и программирование компонент. Разработка приложений, содержащих мультимедийные объекты (изображения и аудиоресурсы). Компоненты *Звук* и *Кнопка*. Разработка приложений «Колокольная галерея», «Виртуальный кот».

Экраны приложения и передача данных между ними. Экран приложения и его свойства. Принципы создания приложений с несколькими экранами. Передача данных между экранами. Использование компонента *Tiny DB* и начального значения экрана. Разработка приложения «Сказочные превращения».

Рисование. Компонент *Холст*. Способы создания приложений с использованием компонента *Холст*. Холст и координатная сетка. Настройка параметров холста. Рисование круга и линий. Вывод текста на холст. Разработка приложений «Рисование», «Раскраска».

8. Презентация проектов (1 час).

Презентация проектов, выполненных обучающимися в рамках занятий по модулю.

Предметные результаты освоения модуля 4.

Учащиеся будут иметь представление:

- о правилах работы за компьютером;
- о правилах безопасной работы в сети Интернет;
- об основных правилах работы в сети и на сайте: <https://scratch.mit.edu>;
- о том, что такое ремикс;
- о правилах, регулирующих обмен материалами внутри Скретч-сообщества;
- о базовых алгоритмических конструкциях (ветвления и циклы) и их реализации в среде программирования Скретч;
- о возможностях дублирования и клонирования спрайтов;

- о сущности дополненной реальности;
- о понятии рекурсии;
- об особенностях последовательного и параллельного алгоритмов;
- об этапах разработки программы (проекта в среде программирования Скретч): постановка задачи, разработка сценарного плана, алгоритмизация, кодирование, тестирование, отладка;
- о назначении и основных возможностях среды MIT App Inventor;
- о технологии коллективной работы над проектом.

Учащиеся будут уметь:

- использовать переменные и списки;
- работать с координатами и случайными числами;
- создавать вспомогательные алгоритмы;
- использовать ветвления и циклы различного вида;
- пользоваться таймером;
- делать звукозапись и использовать музыкальные возможности Скретч;
- создавать дубли и клоны;
- создавать и редактировать свои спрайты в графическом редакторе;
- разрабатывать сценарный план анимации, игры, тренажера, викторины;
- создавать анимации, игры, тренажеры и викторины, в том числе ремиксы в среде программирования Скретч;
- строить рекурсивные алгоритмы;
- создавать параллельные алгоритмы для управления групповой работой исполнителей;
- осуществлять командную разработку проектов.

7. Методика проведения внеурочных занятий с младшими школьниками по курсу «Программируем, учимся и играем!»

В основу методики проведения внеурочных занятий с младшими школьниками по курсу «Программируем, учимся и играем!» положена идея создания ситуации успеха, заключающаяся в том, что практически на каждом занятии обучающийся имеет возможность при поддержке преподавателя или самостоятельно создать законченный программный продукт (анимацию, игру, интерактивный плакат и т. д.) той или иной степени сложности. При этом выполнение работы любого уровня сложности позволяет учащемуся продвигаться в освоении курса дальше, каждый раз самостоятельно определяя уровень погружения в изучаемый материал.

Основной используемый нами метод называется «**использование перед разработкой**», или **метод пар «пример — задача**». Он подразумевает на первом шаге обучения программированию знакомство с законченным (или почти законченным) программным продуктом или его элементами, после чего обучающийся получает аналогичную задачу из рассматриваемого класса задач, решая которую он закрепляет и развивает имеющиеся у него умения (процедурные знания). Эффективность такой методики в получении первичных умений программирования доказана в исследованиях зарубежных и отечественных исследователей [4, 14].

В рамках каждой темы курса обучающиеся:

- 1) знакомятся со специально разработанным примером и воспроизводят его;
- 2) анализируют пример по предложенной схеме и создают аналогичный продукт;
- 3) изменяют, дополняют и развивают продукт, созданный по аналогии;
- 4) выходят на разработку собственного оригинального примера.

Следует отметить, что систематически используемая нами **схема «использование — изменение — дополнение — создание»** не только обеспечивает возможность освоения обучающимися каждой следующей все более сложной темы, но и позволяет через нормы, принятые в Скретч-сообществе, обсудить ряд вопросов, связанных с информационными этикой и правом.

В заключение коротко охарактеризуем учебно-методическое обеспечение курса, размещаемое в открытом доступе **на сайте bosova.ru**. Прежде всего, это сама программа, а также обучающие видеоролики [1], содержащие детальный разбор технологии создания основных примеров продуктов. Кроме того, к каждой теме разработаны методические рекомендации по проведению занятий для преподавателя и презентации по изучаемому материалу. Учебные материалы для обучающихся находятся в стадии разработки.

Список использованных источников

1. Босова Информатика (YouTube-канал). <https://www.youtube.com/channel/UCTn1twdHTQQyFZbVi-4UxNg>
2. Босова А. Ю., Аквилянов Н. А. Визуализация параллельных алгоритмов в программной среде Scratch 2.0 // Информатика в школе. 2018. № 6. С. 27–31.
3. Босова Л. Л., Сорокина Т. Е. Методика применения интерактивных сред для обучения младших школьников про-

граммированию // Информатика и образование. 2014. № 7. С. 61–68.

4. Бутарев К. В. Теория когнитивной нагрузки при обучении программированию - принцип «использование перед разработкой» // Актуальные проблемы обучения математике и информатике в школе и вузе. Материалы VI международной научной интернет-конференции (г. Москва, 11–12 декабря 2020 года) / под общ. ред. Л. И. Боженовой, М. В. Егуповой. М.: Московский педагогический государственный университет, 2021. С. 34–43.

5. Елисеева О. Е. Обучение детей основам создания компьютерных игр на языке программирования Scratch: пособие для учителей учреждений общ. средн. образования с белорус. и рус. яз. обучения: 5–6 классы. Минск: Народная асвета, 2017. 166 с.

6. Лаборатория информационных технологий. Программирование игр и анимации в Scratch. <http://scratch.aelit.net/>

7. Пащковская Ю. В. Творческие задания в среде Scratch: рабочая тетрадь для 5–6 классов. 3-е изд., перераб. и доп. М.: Лаборатория знаний, 2018. 192 с. (Школа программиста.)

8. Примерная рабочая программа основного общего образования предмета «Информатика» (проект). http://www.instra.ru/images/p_r_p-po-uchebnym-predmetam-proekty.pdf

9. Сорокина Т. Е., Босова А. Ю. Информатика. 5–6 класс. Практикум по программированию в среде Scratch / под ред. Л. Л. Босовой. М.: БИНОМ. Лаборатория знаний, 2019. 144 с.

10. Тарапата В. В., Прокофьев Б. В. Учимся вместе со Scratch. Программирование, игры, робототехника. М.: Лаборатория знаний, 2019. 228 с. (Школа юного программиста.)

11. Федеральный государственный образовательный стандарт начального общего образования. <https://fgos.ru/>

12. Федеральный государственный образовательный стандарт основного общего образования. <https://fgos.ru/>

13. Scratch — Идеи. Руководства. <https://scratch.mit.edu/ideas>

14. Trafton J. G., Reiser B. J. The contributions of studying examples and solving problems to skill acquisition // Proceedings of the 15th Annual Conference of the Cognitive Science Society, 1993. Hillsdale, NJ: Erlbaum, 1993. P. 1017–1022. https://www.academia.edu/270647/The_Contributions_of_Studying_Examples_and_Solving_Problems_to_Skill_Acquisition

НОВОСТИ

Российские школьники завоевали четыре золотые медали на 5-й Европейской юниорской олимпиаде школьников по информатике

Юные программисты показали блестящие результаты на 5-й Европейской юниорской олимпиаде школьников по информатике. Российская сборная стала единственной командой, получившей более одной золотой медали, — все четыре школьника, входившие в российскую сборную, получили золотые медали, а двое ребят заняли абсолютные первое и третье места. Нашу страну представляли: Алексей Васильев («Пятьдесят седьмая школа», Москва), Дарья Грекова (физико-математическая школа № 2007, Москва), Иван Пискарев (школа № 2086, Москва) — абсолютное третье место, Ренат Каримов (лицей имени Н. И. Лобачевского Казанского (Приволжского) федерального университета) — абсолютное первое место. Команда для участия в олимпиаде формировалась из числа школьников не старше 15 лет, хорошо показавших себя на всероссийской олимпиаде школьников по информатике. Далее ребята проходили углубленную подготовку и отбор в сборную юниоров на учебно-тренировочных сборах.

В этом году олимпиада объединила рекордное число школьников — 107 официальных участников из 30 стран,

а также 64 внеконкурсных участника. Соревнования принимал город Плоешти (Румыния). В связи с пандемией школьники участвовали в олимпиаде дистанционно из своих стран.

По условиям олимпиады в этом году каждая страна могла представить две команды — основную и дополнительную, для которой участие в состязании является тренировочным. Участники российской дополнительной команды также показали отличный результат: Андрей Цой (образовательный центр — гимназия № 6 «Горно-стай», Новосибирск) получил эквивалент золотой медали, а остальные три участника — эквиваленты серебряной медали.

Руководителем сборных выступила Елена Андреева, зав. кафедрой информатики Специализированного учебно-научного центра (факультета) — школы-интерната имени А. Н. Колмогорова МГУ имени М. В. Ломоносова, зам. руководителя — Андрей Станкевич, доцент факультета информационных технологий и программирования НИУ ИТМО.

(По материалам федерального портала «Российское образование»)