



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
ГУМАНИТАРНО-ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУрГГПУ»)

ФАКУЛЬТЕТ МАТЕМАТИКИ, ФИЗИКИ, ИНФОРМАТИКИ

КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
МЕТОДИКИ ОБУЧЕНИЯ ИНФОРМАТИКЕ

Методические особенности изучения основ алгоритмизации на уроках
информатики

Выпускная квалификационная работа по направлению
по направлению 44.03.05, Педагогическое образование (с двумя
профилями подготовки)

Направленность программы бакалавриата

«Информатика. Математика»

Форма обучения заочная


Проверка на объем заимствований:

86,6% % авторского текста

Работа рецензирована к защите
рекомендована/не рекомендована

14 «Июль» 2024.

зав. кафедрой ИИТиМОИ

 Рузаков А.А.

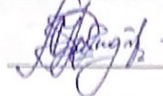
Выполнила:

Студентка группы ЗФ-613-111-5-1
Нигаметьянова Мария Андреевна



Научный руководитель:

к.п.н., доцент кафедры ИИТиМОИ

 Давыдова Н. А.

Челябинск
2024



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
ГУМАНИТАРНО-ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУрГГПУ»)

ФАКУЛЬТЕТ МАТЕМАТИКИ, ФИЗИКИ, ИНФОРМАТИКИ

КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
МЕТОДИКИ ОБУЧЕНИЯ ИНФОРМАТИКЕ

**Методические особенности изучения основ алгоритмизации на уроках
информатики**

**Выпускная квалификационная работа по направлению
по направлению 44.03.05, Педагогическое образование (с двумя
профилями подготовки)**

Направленность программы бакалавриата

«Информатика. Математика»

Форма обучения заочная

Проверка на объем заимствований:
_____ % авторского текста

Работа _____ к защите
рекомендована/не рекомендована
« ___ « _____ 20__ г.
зав. кафедрой ИИТиМОИ

_____ Рузаков А.А.

Выполнила:

Студентка группы ЗФ-613-111-5-1
Нигаметьянова Мария Андреевна

Научный руководитель:

к.п.н., доцент кафедры ИИТиМОИ

_____ Давыдова Н. А.

Челябинск
2024

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ ИЗУЧЕНИЯ ОСНОВ АЛГОРИТМИЗАЦИИ	6
1.1 Понятие алгоритма и основные принципы алгоритмизации.....	6
1.2 Роль алгоритмизации в обучении информатике.....	9
1.3 Основные методы и подходы к изучению алгоритмов	11
1.4 Интеграция алгоритмизации в учебный процесс информатики.....	12
1.5 Организация учебного процесса	14
Выводы по главе 1.....	15
ГЛАВА 2. МЕТОДИЧЕСКИЕ ПОДХОДЫ К ПРЕПОДАВАНИЮ ОСНОВ АЛГОРИТМИЗАЦИИ	17
2.1 Выбор методик и учебных пособий	17
2.2 Ролевая игра и практические задания на уроках.....	22
2.3 Разработка урока с учетом рассмотренных методических особенностей.....	24
Выводы по главе 2.....	47
ЗАКЛЮЧЕНИЕ	49
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	52

ВВЕДЕНИЕ

Изучение основ алгоритмизации на уроках информатики в школе с учетом методических особенностей, таких как индивидуализация образовательного процесса, активное использование интерактивных форм работы на уроках, использование проблемного обучения. Сочетание различных технологий обучения [1], позволяет подойти к процессу обучения систематически и структурировано. Ученики могут лучше понимать принципы работы алгоритмов и развивать навыки логического мышления. Такой подход также способствует формированию у учащихся умения решать задачи эффективно и креативно.

Учет методических особенностей при изучении основ алгоритмизации формируют у учащихся следующие навыки и умения:

1) цифровая грамотность – в век цифровых технологий всё больше задач и проблем решается с помощью компьютеров и программного обеспечения, понимание основ алгоритмизации позволяет учащимся эффективнее использовать компьютерные ресурсы и разрабатывать программы для решения различных задач;

2) понимание основных принципов алгоритмизации и их применение на практике включает в себя знание основных концепций алгоритмов, таких как последовательность действий, условия, циклы и так далее, а также умение применять их для решения конкретной задачи;

3) развитие логического мышления и умения строить последовательные действия для решения задач способствует логическому мышлению, анализированию проблем и построению последовательных шагов для ее решения – навык, позволяющий эффективно решать сложные задачи, разбивая их на более мелкие и управляемые части;

4) навык использования различных методов и приемов для построения эффективных алгоритмов;

5) умение оценивать эффективность и корректность разработанных алгоритмов – важный навык, который позволяет оценивать работу алгоритма с точки зрения его временной сложности, использования ресурсов и правильности решения задачи;

6) развитие навыков программирования для реализации алгоритмов на компьютере – умение писать код на языках программирования, реализующий разработанные алгоритмы, и тестировать их на компьютере;

7) повышение уровня абстрактного мышления и способности к обобщению и систематизации информации – абстрактное мышление помогает видеть общие закономерности и применять их к различным задачам, а систематизация информации позволяет структурировать знания для более эффективного использования;

8) развитие умения работать в команде и совместно решать задачи, требующие алгоритмического подхода – навык, который позволяет эффективно сотрудничать с другими людьми для достижения общей цели при решении сложных задач [3].

В связи с этим сочетание различных технологий обучения: индивидуализация образовательного процесса, активное использование интерактивных форм работы на уроках, использование проблемного обучения, при изучении основ алгоритмизации на уроках в школе становятся все более актуальными, поскольку они помогают подготовить учащихся к цифровому миру и помогут сделать изучение основ алгоритмизации более интересным, полезным и эффективным для учащихся.

Целью работы является разработка и теоретическое обоснование методики обучения основам алгоритмизации в базовом курсе информатики в школе.

Для достижения цели поставлены следующие задачи:

1) проанализировать существующие учебно-методические подходы по изучению курса информатики в школе;

2) изучить современный курс информатики по теме «Основы алгоритмизации», его задачи и цели;

3) разработать вариант проведения урока по основам алгоритмизации;

4) проанализировать результаты применения разработанных методических рекомендаций на практике.

Объектом исследования процесс обучения информатике в средней школе.

Предметом исследования является изучение основ алгоритмизации в курсе информатики.

Теоретическая значимость работы и научная новизна заключается в разработке цифровых образовательных ресурсов (далее ЦОР) для облегчения процесса восприятия информации учащимися в изучении школьного курса информатики с сочетанием различных технологий обучения, таких как индивидуализация образовательного процесса, активное использование интерактивных форм работы на уроках, использование проблемного обучения.

Практическая значимость заключается в возможности применения разработанных ЦОРов за счет внедрения их в учебный процесс на уроках информатики в школе.

Структура работы. Выпускная квалифицированная работа состоит из введения, двух глав, заключения, списка использованных источников. В работе приведено 27 рисунков и 1 таблица. Список использованных источников содержит 12 наименований. Полный объем работы составляет 53 страницы.

ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ ИЗУЧЕНИЯ ОСНОВ АЛГОРИТМИЗАЦИИ

1.1 Понятие алгоритма и основные принципы алгоритмизации

Алгоритм – это последовательность инструкций или шагов, предназначенных для решения определенной задачи или проблемы [2].

Алгоритмы широко используются в различных областях для описания процессов обработки данных и выполнения операций, рассмотрим некоторые из них:

1) в информатике и программировании – алгоритмы используются для решения задач обработки данных, сортировки, поиска, оптимизации и многих других;

2) в математике – алгоритмы применяются для решения задач дискретной математики, комбинаторики, теории чисел и других областей;

3) в инженерии – алгоритмы используются для проектирования и оптимизации систем управления, обработки сигналов, машинного обучения и других технических задач;

4) в биоинформатике – алгоритмы помогают в анализе геномных данных, предсказании структуры белков, выявлении генетических взаимосвязей и других задачах;

5) в финансах – алгоритмы используются для прогнозирования рыночных трендов, оптимизации портфелей инвестиций, управления рисками и других финансовых операций;

6) в медицине – алгоритмы используются для анализа медицинских данных, диагностики заболеваний, прогнозирования эффективности лечения, создания систем поддержки принятия решений для врачей;

7) в телекоммуникациях – алгоритмы применяются для обработки сигналов, управления сетями передачи данных, оптимизации качества связи и других задач;

8) в логистике и транспорте – алгоритмы используются для оптимизации маршрутов доставки, управления запасами, планирования грузоперевозок и других логистических операций;

9) в науке и исследованиях – алгоритмы помогают в обработке экспериментальных данных, моделировании сложных систем, анализе результатов исследований;

10) в образовании – алгоритмы используются для разработки образовательных программ, оценки учебных достижений, индивидуализации обучения и других образовательных процессов.

Алгоритмы могут быть записаны на различных языках программирования и выполняться компьютером или другим устройством.

Важными характеристиками алгоритма являются точность, эффективность и корректность его работы.

Алгоритмы играют ключевую роль в современном мире, помогая автоматизировать процессы, улучшать эффективность работы систем и принимать обоснованные решения на основе данных. Они являются основой многих технологий и инноваций, помогают автоматизировать задачи, улучшать качество жизни людей и развивать различные отрасли экономики.

Алгоритмизация – это процесс разработки алгоритмов для решения задач с использованием компьютера или других устройств [2].

Основные принципы алгоритмизации включают в себя следующие этапы: выходные и входные данные, дискретность, однозначность, конечность, эффективность, общность, модульность, понятность и тестирование.

Разберем каждый из этих этапов алгоритмизации.

Входные (или исходные) данные – определение и описание данных, которые будут обрабатываться алгоритмом. Входные данные помогают определить, какие данные нужно получить и как их обработать.

Выходные данные – определение и описание ожидаемых результатов работы алгоритма. Они дают понять, какие данные должны быть получены в результате работы алгоритма.

Дискретность – разбиение задачи на отдельные шаги или операции, которые могут быть выполнены последовательно. Этап помогает разделить сложную задачу на более простые подзадачи.

Однозначность – каждый шаг алгоритма должен быть четко определен и понятен, что позволяет избежать двусмысленности и неоднозначности при выполнении алгоритма.

Конечность – алгоритм должен завершаться за конечное количество шагов, это гарантирует, что алгоритм будет завершен и не заикнется на бесконечном выполнении.

Эффективность – алгоритм должен решать задачу за разумное время и с использованием разумных ресурсов. Данный этап важен для оптимизации процесса решения конкретной задачи.

Общность – алгоритм должен быть применим к различным ситуациям или задачам, что позволяет использовать один и тот же алгоритм для решения различных задач.

Модульность – разделение алгоритма на независимые модули для упрощения понимания и поддержки кода, делает алгоритм более структурированным и легким для поддержки.

Понятность – алгоритм должен быть понятен для человека, который будет его реализовывать или использовать. Чем проще и понятнее алгоритм, тем легче его будет использовать и поддерживать.

Тестирование – необходимо проводить тестирование алгоритма на различных входных данных для проверки его корректности и эффективности. Тестирование помогает убедиться, что алгоритм работает правильно и даст ожидаемые результаты.

Следуя этим этапам алгоритмизации, можно разработать эффективный и надежный алгоритм для решения задачи.

1.2 Роль алгоритмизации в обучении информатике

Алгоритмизация играет важную роль в обучении информатике в школе, поскольку она помогает ученикам развивать навыки логического мышления, решения проблем и программирования. Изучение алгоритмизации позволяет учащимся развивать умение анализировать задачи, выявлять основные этапы и шаги для их решения, а также оптимизировать процессы [6].

Кроме того, изучение алгоритмизации помогает ученикам развивать творческое мышление, поскольку при создании алгоритмов для решения задач необходимо находить новые и нестандартные подходы. Это способствует развитию креативности и инновационности, что является важным фактором в современном мире.

В процессе изучения основ алгоритмизации учащиеся учатся разрабатывать последовательности действий для решения поставленных задач, структурировать информацию, понимать основные принципы программирования. При составлении алгоритмов необходимо строить последовательные цепочки действий, учитывать условия и принимать решения на основе логических операций, что помогает развивать умение анализировать информацию, выделять главное, определять причинно-следственные связи и делать выводы [3].

Немаловажно, что изучение основ алгоритмизации способствует формированию навыков командной работы и сотрудничества. Зачастую при решении сложных задач часто необходимо совместное усилие нескольких людей для разработки эффективных алгоритмов, что способствует развитию коммуникативных навыков, умению объяснять свои мысли и идеи, а также слушать и учитывать мнение других [1].

Обучение алгоритмизации помогает ученикам понять, как компьютер обрабатывает информацию и какие шаги необходимо выполнить для

достижения определенной цели. Эти навыки являются основой для дальнейшего изучения программирования и информационных технологий.

Изучение основ алгоритмизации на уроках информатики в школе может быть разделено на несколько этапов, которые помогут учащимся усвоить ключевые концепции и навыки. Рассмотрим возможные этапы изучения данной темы [11]:

1) введение в понятие алгоритма – на этом этапе ученикам объясняется, что такое алгоритм, какие задачи он решает и почему важно уметь разрабатывать и использовать алгоритмы;

2) основные элементы алгоритма – ученики изучают основные элементы алгоритма, такие как последовательность действий, условия и циклы, учатся составлять простейшие алгоритмы для решения простых задач;

3) структуры данных – на данном этапе происходит знакомство с различными структурами данных, такими как массивы, списки, стеки и очереди;

4) алгоритмические методы – ученики изучают различные алгоритмические методы, такие как поиск, сортировка, рекурсия и другие;

5) практические задания – на этом этапе ученики выполняют практические задания, где им предлагается решить конкретные задачи с помощью алгоритмов, позволяющие закрепить полученные знания и навыки;

6) самостоятельное творчество – ученики разрабатывают собственные алгоритмы для решения интересных задач или создания своих проектов.

Каждый из этих этапов играет важную роль в формировании навыков в изучении раздела на уроках информатики в школе «основ алгоритмизации» у учащихся и подготовке их к дальнейшему изучению информатики и программирования.

1.3 Основные методы и подходы к изучению алгоритмов

Изучение основ алгоритмизации на уроках информатики в школе может включать в себя различные методы и подходы, направленные на формирование понимания и навыков работы с алгоритмами. Некоторые из основных методов и подходов включают в себя [2]:

1) объяснение теории – учитель начинает с объяснения основных понятий и принципов работы алгоритмов, включая последовательность действий, условия, циклы и т.д.;

2) демонстрация – учитель демонстрирует выполнение алгоритмов на примерах, используя компьютер или доску, чтобы проиллюстрировать их работу;

3) практические задания – ученики выполняют практические задания, например, написание алгоритмов на бумаге, в текстовом редакторе или с помощью специальных программных средств;

4) групповая работа – ученики могут работать в группах, чтобы обсудить и разработать алгоритмы вместе, обмениваясь идеями и решениями;

5) решение задач – ученики решают задачи, которые требуют применения алгоритмов для поиска решений.

Данные методы и подходы способствуют формированию понимания учеников о работе алгоритмов и их практическом применении. Они могут использоваться методы как отдельно, так и в комбинации друг с другом для более эффективного изучения алгоритмов на уроках информатики [8].

Все эти методы играют важную роль в эффективном обучении основ алгоритмизации на уроках информатики, помогая учащимся лучше понять и применять основные принципы работы алгоритмов.

Использование разнообразие методов в процессе обучения, таких как групповая работа, ролевые игры и интерактивные онлайн-ресурсы, способствует активизации участия учащихся в процессе обучения. Это

помогает им лучше усваивать материал и развивать навыки работы с алгоритмами.

Различные методы обучения способствуют повышению мотивации учащихся к изучению алгоритмов, что помогает создать благоприятную обучающую среду и стимулирует учеников к достижению улучшения результатов.

Групповая работа и обсуждение результатов проектов друг с другом способствуют развитию коммуникативных навыков, учит эффективно взаимодействовать учеников между собой, обсуждать свои идеи и аргументировать свои решения.

Обратная связь и рефлексия после выполнения заданий помогают учащимся развивать навыки саморегуляции, анализировать свои ошибки и выявлять области для улучшения. Это способствует развитию самостоятельности и ответственности за свое обучение.

1.4 Интеграция алгоритмизации в учебный процесс информатики

Интеграция алгоритмизации в учебный процесс информатики в школе является важным шагом для развития компьютерной грамотности у учащихся. Для успешной интеграции алгоритмизации в учебный процесс существуют следующие методы.

Обучение основам алгоритмизации. Для удовлетворительной работы необходимо начинать с простых задач, таких как сортировка чисел или поиск элемента в массиве.

Использование специализированных программных средств. Следует начать с использования специальных программных средств, таких как блочные языки программирования, которые позволяют учащимся создавать алгоритмы без необходимости изучения сложного синтаксиса и не потерять интерес к теме.

Интеграция алгоритмизации в другие предметы. Например, в таких дисциплинах как математика, физика или биология есть возможность использования алгоритмизации в качестве инструмента для решения задач.

Использование интерактивных онлайн-ресурсов. Существует большое множество онлайн-платформ и ресурсов, которые предлагают интерактивные уроки и задачи по алгоритмизации, позволяющие помочь учащимся освоить концепции алгоритмизации через интерактивное обучение [4].

Работа в группах. Организация совместной работы учащихся над созданием алгоритмов или работа в группах позволяет ученикам обмениваться идеями, учиться друг у друга и развивать навыки командной работы.

Привлечение профессионалов или посещение информационных выставок и музеев. Приглашение специалистов из области информационных технологий для проведения мастер-классов по алгоритмизации может быть стимулирующим опытом для учащихся, а также поможет им понять, какие возможности открывает алгоритмизация в реальном мире. Некоторым ученикам это поможет в определении будущей профессии, ведь наглядный пример вовлечения в процесс немаловажен для каждого ребенка.

Регулярные практические задания, на которых учащиеся могут применять свои знания в создании алгоритмов закрепят полученную информацию по теме, а обратная связь от учителей и сверстников мотивирует на продолжение своего обучения в данной сфере.

Использование игровых подходов. Разработка игровых задач и проектов, требующие от учащихся создания алгоритмов для решения различных игровых ситуаций, позволяет сделать процесс обучения более увлекательным для учащихся.

Делая выводы, можно сказать, что интеграция алгоритмизации в учебный процесс информатики поможет учащимся развить навыки

логического мышления, проблемного решения и программирования, а также поможет создать эффективную и интересную образовательную среду, способствующую развитию навыков алгоритмизации у учащихся.

1.5 Организация учебного процесса

Для эффективной организации учебного процесса на уроках информатики следует придерживаться следующего плана урока:

1) планирование урока, то есть определение целей и задач урока, выбор методов обучения и форм работы с учащимися, разработка учебного материала;

2) введение в тему, привлечение внимания учащихся к теме урока, объяснение актуальности изучаемого материала;

3) объяснение нового материала – систематическое и последовательное изложение информации, использование наглядных примеров и демонстраций;

4) дача практических заданий, то есть предоставление учащимся возможности применить полученные знания на практике, выполнение заданий в рабочих тетрадях, на компьютере или на доске;

5) контроль знаний – проверка понимания материала, проведение тестов, задач, контрольных работ;

6) рефлексия, то есть обсуждение результатов урока, выявление проблемных моментов, анализ ошибок и способов их исправления;

7) домашнее задание – назначение заданий для самостоятельной работы дома, чтобы закрепить изученный материал.

Важно также создать поддерживающую атмосферу в классе, поощрять активное участие учащихся, индивидуализировать подход к каждому ученику и стимулировать их интерес к изучению информатики.

Методы и приемы организации учебного процесса на уроках информатики помогут не только эффективно передать знания учащимся, но

развить их критическое мышление, творческие способности и навыки работы с информацией в цифровой эпохе.

Рассмотрим некоторые методы и приемы организации учебного процесса.

Дифференцированный подход – учитывание индивидуальных особенностей учащихся, их уровня знаний и способностей при планировании и проведении урока.

Интерактивные методы обучения – использование интерактивных досок, онлайн-ресурсов, игровых технологий для привлечения внимания учащихся и повышения их мотивации к изучению информатики.

Проектная деятельность – организация учебного процесса в форме проектов, которые позволяют учащимся применять полученные знания на практике, развивать творческое мышление и коммуникативные навыки.

Коллективная работа – проведение уроков в форме групповых заданий, дискуссий и обсуждений, что способствует развитию социальных навыков учащихся и сотрудничеству в коллективе.

Использование разнообразных образовательных ресурсов – использование электронных учебников, видеоуроков, онлайн-курсов для обогащения учебного материала и разнообразия форм обучения.

Обратная связь – регулярное обсуждение прогресса учащихся, выявление проблемных моментов и поиск путей их решения, а также поощрение успехов и достижений.

Самостоятельная работа – организация заданий для самостоятельного изучения материала, выполнения практических заданий и развития навыков самообучения.

Выводы по главе 1

Таким образом, мы выяснили, что алгоритмизация играет важную роль в обучении информатике, так как помогает структурировать и систематизировать процессы решения задач. В главе рассматриваются

основные подходы к изучению алгоритмов. Понимание и использование алгоритмов является ключевым элементом успешного обучения информатике и развития компьютерных навыков.

Также мы подчеркнули важность понимания базовых понятий алгоритмов. Выявили значимость развития навыков анализа, проектирования и реализации алгоритмов в контексте обучения информатике.

Все это способствует формированию у учеников компетенций в области алгоритмизации и программирования.

ГЛАВА 2. МЕТОДИЧЕСКИЕ ПОДХОДЫ К ПРЕПОДАВАНИЮ ОСНОВ АЛГОРИТМИЗАЦИИ

2.1 Выбор методик и учебных пособий

При изучении основ алгоритмизации на уроках информатики используются следующие нормативные и рекомендуемые документы и учебники [7]:

1. Федеральный компонент государственного образовательного стандарта начального общего образования (ФГОС НОО) и основного общего образования (ФГОС ООО), который содержит требования к изучению информатики и основ алгоритмизации [11].

2. Учебники по информатике для основной школы, соответствующие ФГОС ООО. Например, «Информатика. 5 – 9 классы» под редакцией Л.М. Захаровой и др.; «Информатика и ИКТ: учебник для 8 класса» под редакцией Н.И. Морозовой, М.В. Лисовой и др. [1].

3. Методические рекомендации «Функциональная грамотность в контексте обновленных ФГОС ООО по информатике» [6].

4. Учебные пособия и методические материалы, разработанные опытными педагогами и специалистами в области информатики и программирования для школьников [2].

Во время освоения темы можно использовать как отдельно, так и в совокупности методики обучения и учебные пособия, включая:

1) индивидуализация образовательного процесса – предлагает адаптацию содержания и методик обучения к индивидуальным особенностям каждого ученика для этого педагог должен проводить диагностику знаний и способностей каждого ребёнка, чтобы определить его уровень развития и потребностей. На основе полученных данных педагог может выбрать соответствующие методы обучения, задания материалы для каждого ученика;

2) активное использование интерактивных форм работы на уроках. Вместо традиционной модели лекции педагог ставит перед собой задачу активизировать участие каждого ученика в процессе обучения. Это можно достичь с помощью таких форм работы, как групповые проекты, мозговые штурмы, дискуссии, игры и другие интерактивные методы;

3) разнообразные технологии в обучении. Современные ученики активно используют технику и интернет в своей повседневной жизни, поэтому педагог должен уметь принимать эти средства в образовательном процессе. Это может быть использование интерактивных досок, компьютерных программ, интернет ресурсов и других средств;

4) использование проблемного обучения. При этом подходе ученик не просто получает готовые знания от учителя, а самостоятельно исследует предметную область и решает возникшие проблемы;

5) учебники и рабочие тетради, охватывающие основы информатики, алгоритмы, программирование, работу с различными программными средствами и технологиями [2].

Например:

– «Основы информатики: Учебник для 7 класса» – авторы: Босова Л.И., Бурмистрова Н.А., Горячев А.В. Рабочая тетрадь: «Основы информатики: Рабочая тетрадь для 7 класса» – авторы: Босова Л.И., Бурмистрова Н.А., Горячев А.В.

– «Информатика и ИКТ: Учебник для 8 класса» – авторы: Петров И.Т., Сидорова Е.А. Рабочая тетрадь: «Информатика и ИКТ: Рабочая тетрадь для 8 класса» – авторы: Петров И.Т., Сидорова Е.А.

– «Основы информатики и программирования: Учебник для 9 класса» – авторы: Николенко А.Г., Семакин В.Н. Рабочая тетрадь: «Основы информатики и программирования: Рабочая тетрадь для 9 класса» – авторы: Николенко А.Г., Семакин В.Н.;

б) онлайн-курсы и образовательные платформы, предлагающие интерактивные уроки, видеоуроки, задания и тесты для самостоятельного

обучения. Например: «Единая коллекция цифровых образовательных ресурсов (URL: <http://school-collection.edu.ru/>);

7) учебные пособия и методические материалы для учителей, помогающие им разработать уроки, лабораторные работы и другие виды занятий по информатике.

Выбор методик и учебных пособий зависит от уровня подготовки учащихся, их интересов и целей обучения.

В зависимости от выбранного метода обучения, в своей работе можно принимать следующие методики обучения.

Методика «ученик – учитель» – позволяет ученикам самостоятельно изучать основы алгоритмизации с помощью учебников, онлайн-курсов и интерактивных обучающих программ [7]:

1) использование учебников по информатике, например, «Информатика. 5 – 9 классы» автора И. Г. Семакина;

2) онлайн-курсы, на таких платформах как Stepik или Coursera, к которым есть свободный доступ для каждого учащегося. А для учителя можно выполнить простой способ отслеживания успеваемости и усвоения материала;

3) интерактивные обучающие программы, например, «CodeCombat», «Code.org» или Scratch.

Онлайн-платформы предоставляют возможность создания алгоритмов с помощью блоков кода.

Code.org – предоставляет бесплатные учебные материалы и интерактивные уроки по программированию для школьников.

Scratch – позволяет создавать интерактивные истории, анимации и игры, используя блочное программирование.

Методика «проектное обучение» – ученики могут изучать алгоритмизацию, работая над проектами, разрабатывая программы и решая задачи [7]:

1) разработка проектов, связанных с созданием программ или игр с использованием алгоритмов;

2) работа над задачами, связанными с алгоритмизацией, например, создание алгоритма для решения задачи поиска кратчайшего пути на графе.

Использование учебных пособий, таких как «Азбука программирования» или «Основы алгоритмизации и программирования», которые содержат теоретический материал и практические задания:

1) «Азбука программирования» автора В.В. Петрова и Н.В. Петровой;

2) «Основы алгоритмизации и программирования» автора В.П. Краснова.

Выбор методик и учебных пособий при изучении материала зависит от потребностей и возможностей конкретной школы, а также от интересов и уровня подготовки учащихся.

На примере разберем методику «учитель – ученик» на базе онлайн-курса платформы Stepik.

Для начала учителю необходимо выполнить регистрацию на платформе Stepik (URL: <https://stepik.org/>), и выполнить переход на вкладку «Преподавание» в верхней части экрана, для создания собственного онлайн-курса.

1. Название курса: «Основы алгоритмизации».

2. Описание курса:

В этом онлайн-курсе вы познакомитесь с основами алгоритмизации, изучите различные методы решения задач, научитесь писать эффективные алгоритмы и реализовывать их на практике. Курс предназначен для начинающих, не требует специальных знаний программирования и подходит как для школьников, так и для взрослых, желающих освоить основы алгоритмизации.

3. Модули курса:

– введение в алгоритмизацию;

- основные структуры данных;
- алгоритмы сортировки;
- поиск и обход графов;
- программирование на языке программирования: Python.

4. Формат курса:

- видеолекции с объяснением материала;
- тесты для проверки знаний;
- практические задания для самостоятельной работы;
- форум для обсуждения вопросов.

5. Примерные задания:

- написать алгоритм сортировки массива чисел;
- реализовать поиск кратчайшего пути в графе;
- написать программу на Python для решения конкретной задачи.

6. Завершение курса:

По завершении курса участники получают сертификат о прохождении, подтверждающий их знания в области алгоритмизации.

Это лишь примерный план разработки онлайн-курса по основам алгоритмизации на платформе Stepik. Конечно, разработка курса требует более детального изучения темы, создания контента, тестирования и доработок.

Современные образовательные технологии играют важную роль в изучении информатики в школе в целом и значительно обогащают процесс изучения основ алгоритмизации, такие как: интерактивные уроки, онлайн-курсы, программирование и робототехника, визуализация алгоритмов, коллективная работа и проекты.

Существует множество онлайн-курсов и платформ, которые предлагают интерактивные уроки по основам алгоритмизации. Учителя могут использовать эти ресурсы для демонстрации концепций и примеров алгоритмов, а также для домашних заданий и самостоятельного обучения.

Использование программирования и робототехники в учебном процессе поможет ученикам увидеть практическое применение алгоритмов. Учащиеся могут создавать и тестировать свои собственные алгоритмы на компьютерах или программируемых роботах, что поможет им лучше понять основы алгоритмизации.

Следует заметить, что, к сожалению, еще не все школы внедрили в процесс обучения робототехнику, несмотря на то что данное представление информации является очень эффективным и интересным для учащихся.

Визуализация алгоритмов. Существуют специальные программы и инструменты, которые позволяют визуализировать алгоритмы в виде блок-схем или графов. Конкретная технология помогает учащимся лучше понять структуру алгоритмов и их последовательность выполнения.

Использование современных образовательных технологий при изучении основ алгоритмизации делает уроки более интересными, доступными и эффективными для учащихся.

2.2 Ролевая игра и практические задания на уроках

Ролевая игра и практические задания могут быть отличным способом обучения информатике в школе.

Создание ролевой игры, в которой учащиеся будут играть роли различных персонажей, связанных с информатикой. Например, один ученик может играть роль программиста, другой – системного администратора, третий – дизайнера веб-сайтов и т.д. В ходе игры они могут решать разные задачи и проблемы, связанные с их профессиональной деятельностью [4].

Пример такой игры для лучшего изучения основ алгоритмизации выглядит следующим образом: ролевая игра «Путешествие по алгоритмическому лесу»:

Учитель объявляет, что ученики отправляются в путешествие по алгоритмическому лесу, где каждый участник будет представлять определенный алгоритм сортировки.

Ученики разделяются на группы, каждая из которых представляет определенный алгоритм (например, сортировка пузырьком, сортировка выбором, быстрая сортировка и т. д.). Группы получают набор случайных чисел, который им нужно отсортировать с помощью своего алгоритма.

Ученики в каждой группе обсуждают и разрабатывают план действий, чтобы продемонстрировать работу своего алгоритма на практике.

После того как все группы завершили сортировку, учитель проводит обсуждение эффективности каждого алгоритма, его сложности и применимости в различных ситуациях.

Такая ролевая игра поможет ученикам лучше понять основы алгоритмизации, а также на практике увидеть различия между разными методами сортировки.

Можно рассмотреть еще один пример ролевой игры: «Алгоритмический детектив»:

1) учитель создает «загадку» – неправильно написанный алгоритм для решения конкретной задачи;

2) ученики делятся на группы и получают этот «загадочный» алгоритм;

3) задача каждой группы – выявить ошибки в алгоритме и исправить их, чтобы он работал правильно.

Проведение практических заданий, таких как создание веб-сайтов, разработка компьютерных программ, анализ данных с использованием электронных таблиц и т.д. Учащиеся могут работать в группах или индивидуально, чтобы решать реальные проблемы с использованием информационных технологий [10].

Пример практического задания «Создание алгоритма для решения конкретной задачи»:

1) учитель предлагает группе учеников конкретную задачу, например, поиск определенного элемента в массиве;

2) группа учеников работает вместе, чтобы разработать алгоритм для решения этой задачи, используя псевдокод или блок-схемы;

3) после этого группы представляют свои алгоритмы перед классом, и все вместе обсуждают их эффективность и правильность.

Рассмотрим более детально пример практического задания по созданию алгоритма для поиска определенного элемента в массиве:

Задача: написать алгоритм для поиска числа 7 в массиве.

Предварительные шаги:

- 1) определить, что нужно найти число 7 в массиве;
- 2) задать начальное значение переменной для хранения индекса найденного элемента.

Алгоритм поиска:

- 1) начать с первого элемента массива (с индексом 0);
- 2) сравнить текущий элемент с числом 7;
- 3) если текущий элемент равен 7, сохранить его индекс и завершить поиск;
- 4) если текущий элемент не равен 7, перейти к следующему элементу и повторить шаги сравнения.

Такие методы обучения помогут учащимся лучше понять и применить знания по информатике на практике, а также развить навыки командной работы, критического мышления и решения проблем.

2.3 Разработка урока с учетом рассмотренных методических особенностей

Знакомство с основами алгоритмизации начинается уже в начальной школе. Далее в основной школе и старших классах углубляется изучение алгоритмов и структур данных.

В начальной школе дети могут изучать простые алгоритмы, такие как последовательность действий для решения задачи или игры. Например, они

могут создавать алгоритм для решения задачи на сложение чисел или для перемещения персонажа в игре по определенному пути.

В основной школе ученики могут изучать более сложные алгоритмы, такие как сортировка данных или поиск элемента в массиве. Они также могут познакомиться со структурами данных, такими как массивы, списки, очереди и деревья.

В старших классах школьники могут изучать более продвинутые концепции программирования, такие как рекурсия, динамическое программирование и алгоритмы на графах. Они также могут углубляться в изучение различных структур данных и их применение в реальных задачах.

Раздел «Основы алгоритмизации» в 8 классе рассчитаны на 9 часов. Календарно-тематическое планирование представлено в таблице [2].

Планирование составлено на основе:

– авторской программы основного общего образования по Информатике: 7 – 9 классы / Л.Л. Босова, А.Ю. Босова – М.: БИНОМ. Лаборатория знаний, 2015 г.;

– учебника по информатике: Информатика – 8 класс: учебник для учащихся общеобразовательных учреждений/ Л.Л. Босова, А.Ю. Босова М.: БИНОМ. Лаборатория знаний, 2018 г.

Таблица – Календарно-тематическое планирование 8 класса

№	Тема	Кол-во часов	Формы контроля	Результаты развития		
				Личностные	Метапредметные	Предметные
1	2	3	4	5	6	7
14	Алгоритмы и исполнители	1		алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе	уметь: – понимать смысл понятия «алгоритм» и широты сферы его применения; – понимать ограничения, накладываемые средой исполнителя и системой команд на круг задач,	- смысл понятия «алгоритм»; – умение анализировать предлагаемые последовательности команд на предмет наличия у них таких свойств алгоритма как дискретность, детерминированность, понятность, результативность, массовость; –

Продолжение таблицы

1	2	3	4	5	6	7
					решаемых исполнителем	термины «исполнитель», «формальный исполнитель», «среда исполнителя», «система команд исполнителя» и др.; – умение исполнять алгоритм для формального исполнителя с заданной системой команд
15	Способы записи алгоритмов	1	теоретический диктант, практическая работа на тему: «Запись алгоритмов различными способами»	алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе	уметь: – анализировать предлагаемые последовательности команд на предмет наличия у них таких свойств алгоритма как дискретность, детерминированность, понятность, результативность, массовость; – понимание преимущества и недостатков той или иной формы записи алгоритмов; – Умение переходить от одной формы записи алгоритмов к другой; – умение выбирать форму записи алгоритма, соответствующую решаемой задаче	- различные способы записи алгоритмов
16	Записи алгоритмов	1		алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе	уметь: – понимать сущность понятия «величина»; – понимать границы применимости величин того или иного типа	- представление о величинах, с которыми работают алгоритмы; – правила записи выражений на алгоритмическом языке; операции присваивания

Продолжение таблицы

1	2	3	4	5	6	7
17	Алгоритмическая конструкция следование	1		алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе	уметь: – выделять линейные алгоритмы в различных процессах; – понимать ограниченности возможностей линейных алгоритмов	представление об алгоритмической конструкции «следование»; – исполнение линейного алгоритма для формального исполнителя с заданной системой команд; – составление простых (коротких) линейных алгоритмов для формального
18	Алгоритмическая конструкция ветвление. Полная форма ветвления. Сокращённая форма ветвления	1		алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе	уметь: – выделять алгоритмы с ветвлением в различных процессах; – понимать ограниченности возможностей линейных алгоритмов	исполнителя с заданной системой команд – представление об алгоритмической конструкции «ветвление»; – исполнение алгоритма с ветвлением для формального исполнителя с заданной системой команд; – составление простых (коротких) алгоритмов с ветвлением для формального исполнителя с заданной системой команд
19	Алгоритмическая конструкция повторение. Цикл с заданным условием продолжения работы	1	практическая работа на тему: «Алгоритм цикла с заданным условием продолжения работы»	алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе	уметь: – выделять циклические алгоритмы в различных процессах	- представления об алгоритмической конструкции «цикл», о цикле с заданным условием продолжения работы; – исполнение циклического алгоритма для формального исполнителя с заданной системой команд; – составление простых

Продолжение таблицы

1	2	3	4	5	6	7
						циклических алгоритмов для формального исполнителя с заданной системой команд
20	Алгоритмическая конструкция повторение. Цикл с заданным условием окончания работы	1	практическая работа на тему: «Алгоритм цикла с заданным условием окончания работы»	алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе	уметь: – выделять циклические алгоритмы в различных процессах	– представления об алгоритмической конструкции «цикл», о цикле с заданным условием окончания работы; – исполнение циклического алгоритма для формального исполнителя с заданной системой команд; – составление простых циклических алгоритмов для формального исполнителя с заданной системой команд
21	Алгоритмическая конструкция повторение. Цикл с заданным числом повторений	1	практическая работа на тему: «Алгоритм цикла с заданным числом повторений»	алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе	уметь: – выделять циклические алгоритмы в различных процессах	– представления об алгоритмической конструкции «цикл», о цикле с заданным числом повторений; – исполнение циклического алгоритма для формального исполнителя с заданной системой команд; – составление простых циклических алгоритмов для формального исполнителя с заданной системой команд
22	Обобщение и систематизация основных понятий темы Основы алгоритмизации. Проверочная работа «Основы алгоритмизации»	1	Компьютерное тестирование	алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе.	уметь: – самостоятельно о планировать пути достижения целей; – соотносить свои действия с планируемыми	– основные понятия темы «Основы алгоритмизации»

Продолжение таблицы

1	2	3	4	5	6	7
					результатами, осуществлять контроль своей деятельности, определять способы действий в рамках предложенных условий, корректиро- вать свои действия в соответствии с изменяющейся ситуацией; – оценивать правильность выполнения учебной задачи; – владеть владеть основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательн ой деятельности	

Для проведения урока 8 класса «Алгоритмы и исполнители», с целью эффективного изучения материала мною был создан цифровой образовательный ресурс в виде веб-сайта, который включает в себя теоретическую и практическую часть. Для увлеченности учеников в процесс обучения и лучшего усвоения материала, были учтены следующие методические особенности:

- 1) использование интерактивных форм работы на уроке: дискуссия, групповая работа, игры;
- 2) применение разнообразных технологий: визуальное пособие, интерактивная доска, схемы алгоритмов, веб-сайт.
- 3) использование проблемного обучения: практические задания после изучения теоретической части урока.

Для просмотра созданного сайта, необходимо перейти по ссылке URL:
<http://nigametuanova.tilda.ws/>.

Тема урока: «Алгоритмы и исполнители» 8 класс.

Урок составлен по учебнику «ФГОС ООО. Информатика. 8 класс» авторы: Босова Л.Л., Босова А.Ю.

Цель урока: Ознакомить учащихся с основными принципами алгоритмов и научить применять их на практике с помощью исполнителей.

Ход урока:

1. Введение:

- приветствие учащихся;
- переход на цифровой образовательный ресурс по ссылке.

При открытии веб-сайта, в шапке указана тема урока «Алгоритмы и исполнители», далее представлено понятие алгоритма, чтобы учащиеся ознакомились с понятием «Алгоритм».

Начальная страница веб-сайта представлена на рисунке 1.

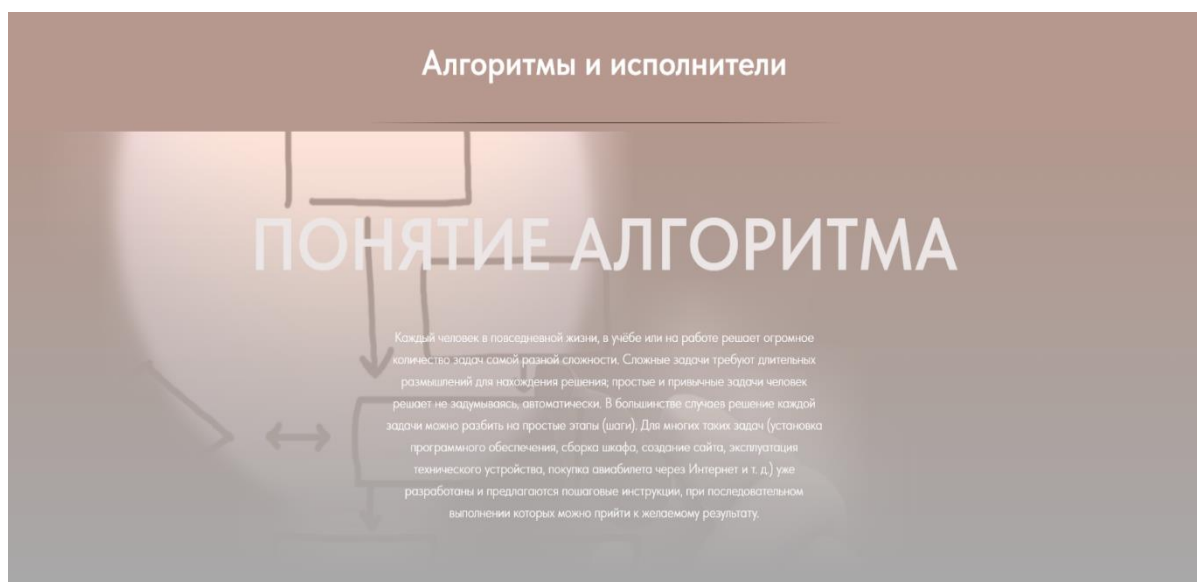


Рисунок 1 – Начальная страница веб-сайта

2. Теоретическая часть:

- примеры алгоритмов из повседневной жизни;

При скроллинге веб-сайта, ученикам представляется три варианта алгоритмов из повседневной жизни. Учащейся должны ответить на вопрос: «Что общего у этих трех картинок». При нажатии на картинку, она

открывается на весь экран, с возможностью просмотреть следующие изображения. Данное задание позволяет понять, что алгоритм присутствует в жизни каждого человека и не является чем-то новым и непонятным. После ответов учеников, необходимо нажать на кнопку «Правильный ответ», чтобы собрать воедино все названные варианты. На новой странице откроется текст с правильным ответом. Задание «Что общего у трех картинок» представлено на рисунке 2.



Рисунок 2 – Задание «Что общего у этих трех картинок»

Страница с правильным ответом представлена на рисунке 3.

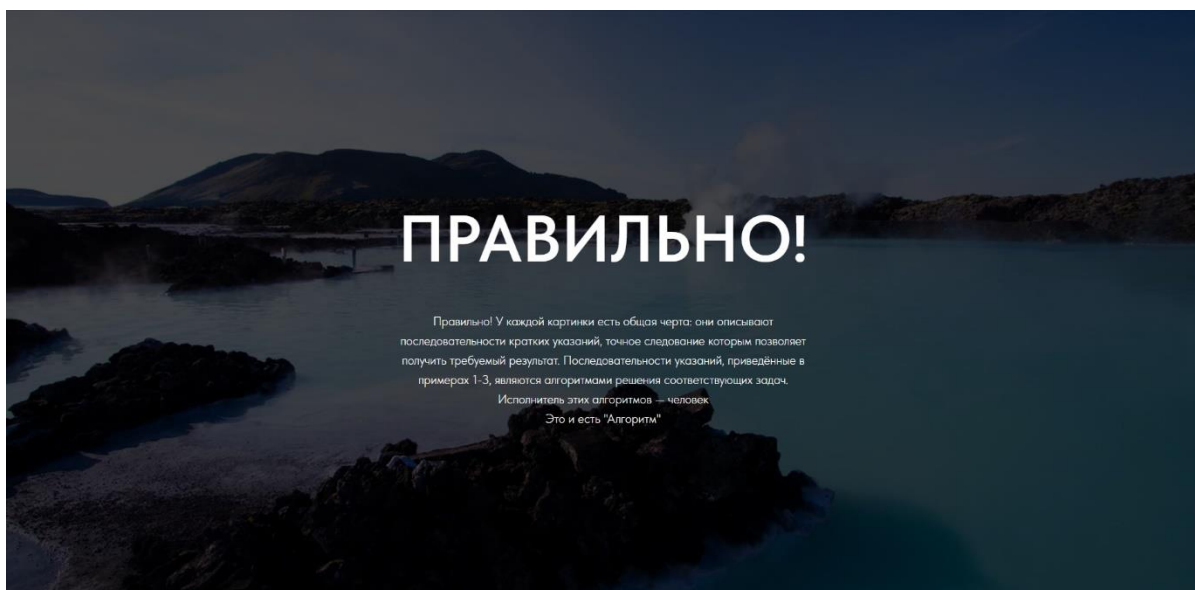


Рисунок 3 – Правильный ответ на задание

– объяснение основных понятий: алгоритм, исполнитель, свойства алгоритмов, формы представления алгоритмов.

Прокручивая страницу далее, мы знакомимся с основным понятием алгоритма, с общим видом-схемой работы алгоритма и интересный факт – рисунок 4.

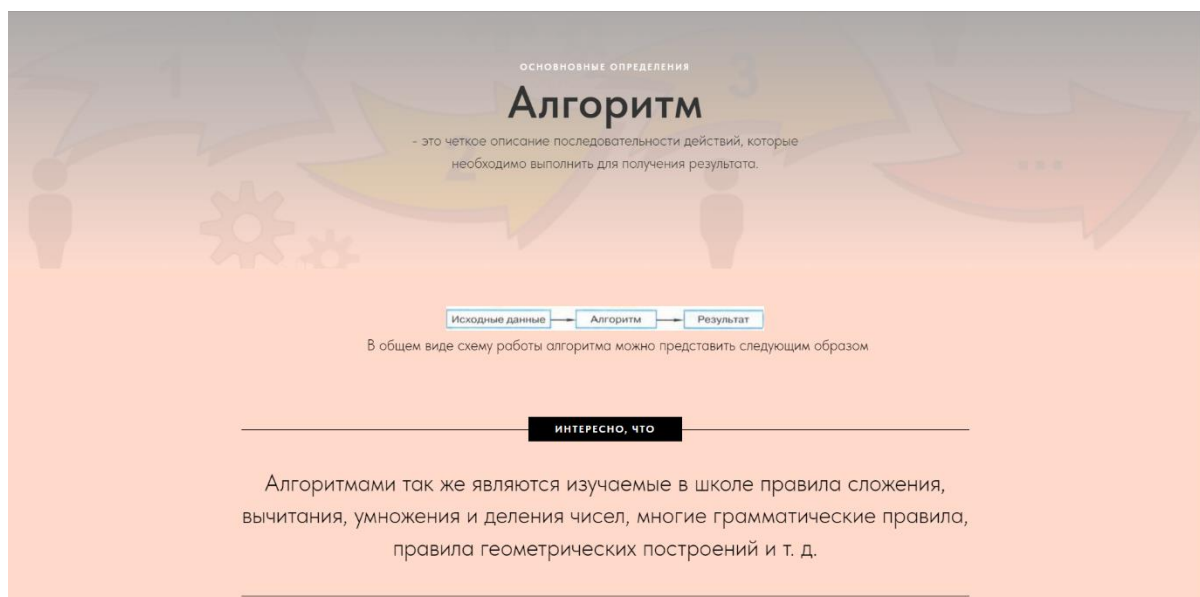


Рисунок 4 – Общий вид понятия алгоритма

Далее происходит знакомство с исполнителями алгоритма: что такое исполнитель и какими они бывают. Понятие исполнителя алгоритма представлено на рисунке 5.

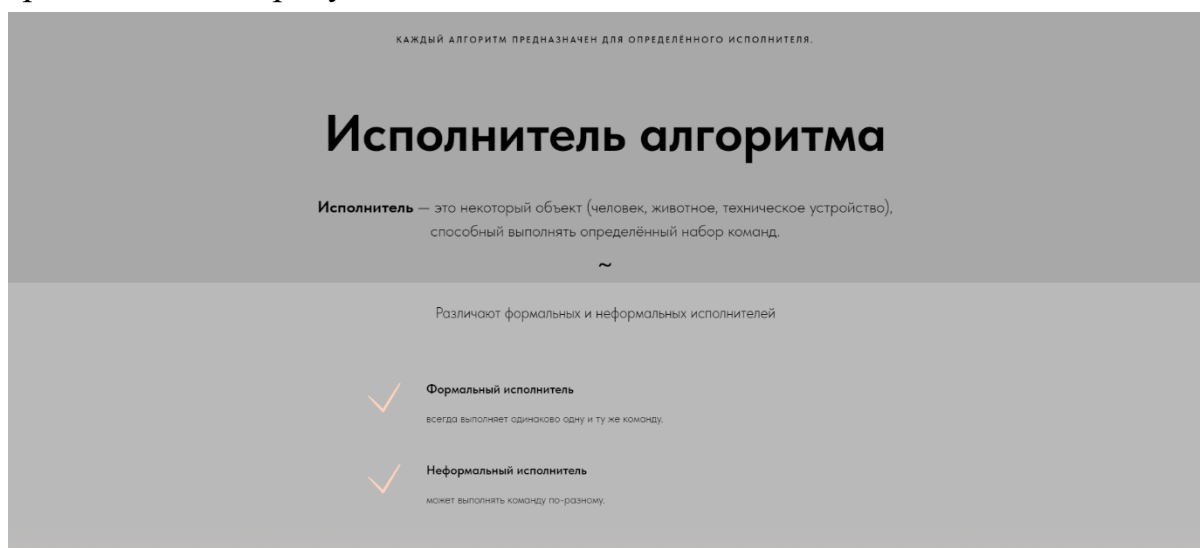


Рисунок 5 – Основные понятия исполнителя

3. Практическая часть:

- разбиение учащихся на группы;
- задание – игра «Исполнитель «Черепашка»;

Для закрепления пройденного материала по исполнителям, учащимся предлагается попробовать на практике в игровой форме исполнителя

«Черепашка», суть которого состоит в написании последовательности команд, алгоритма. При правильном подборе команд, «черепашка» двигается, оставляя след в виде линии. Чтобы запустить задание – игру, необходимо нажать на кнопку «Исполнитель «Черепашка»».

Здание представлено на рисунке 6.

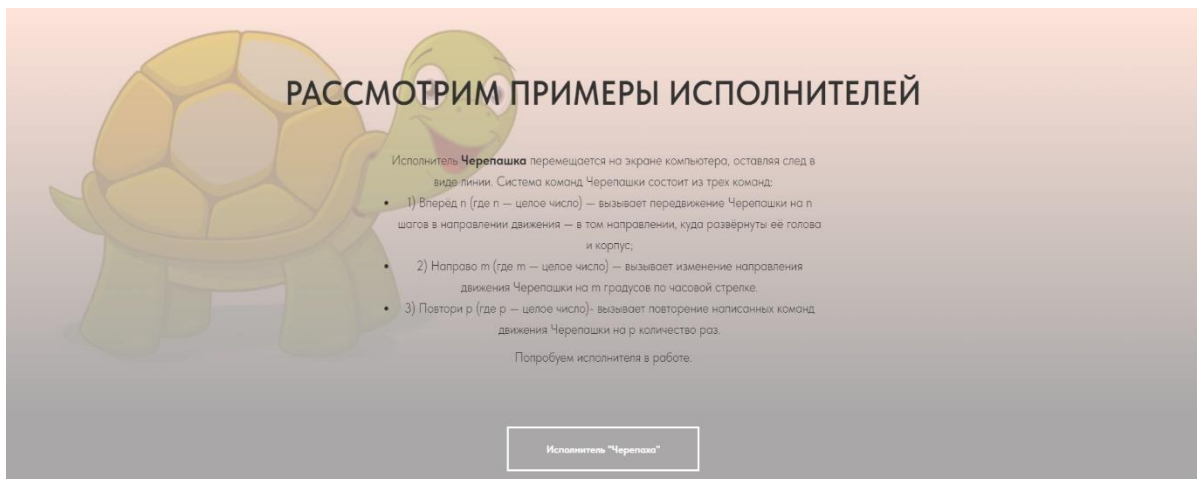


Рисунок 6 – Задание – игра «Исполнитель «Черепашка»»

После нажатия на кнопку, задание откроется в новом окне. Практическое задание представлена в 10 уровнях. В течении урока, для понятия алгоритма, можно пройти 1–2 уровня, если учащимся понравится задание – игра, они для самостоятельного закрепления материала, могут дома продолжить другие уровни. Задние – игра представлена на рисунке 7.

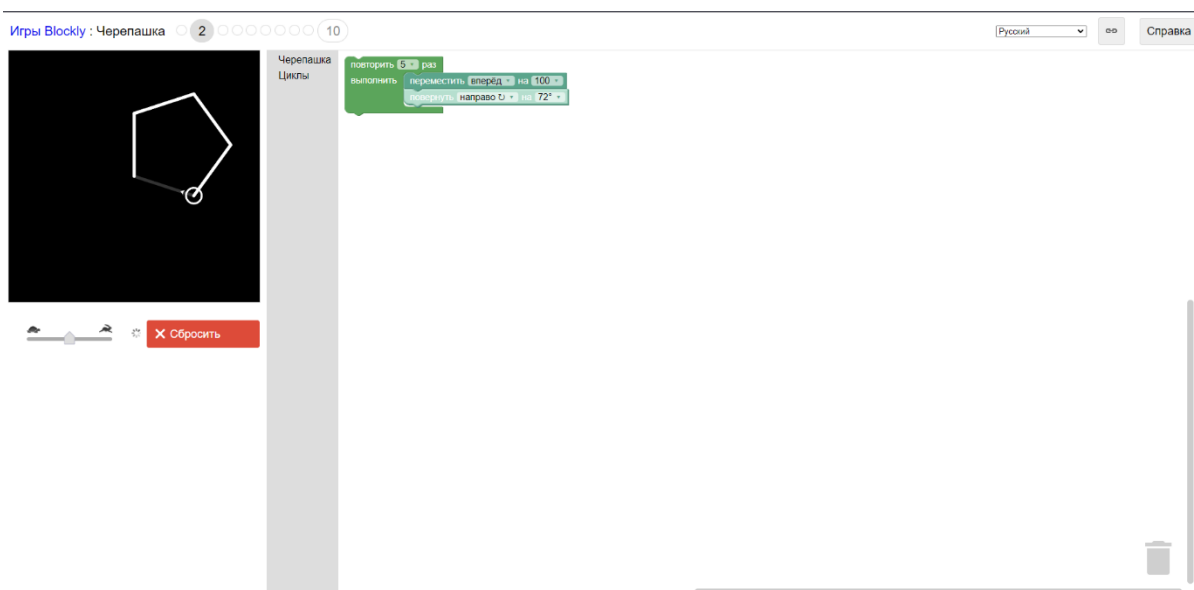


Рисунок 7 – Задание – игра «Исполнитель «Черепашка», уровень 2

Если была допущена ошибка при выполнении программы, в правом нижнем углу есть кнопка удаления. Один и тот же уровень можно выполнить разными способами. Можно попробовать с учениками их все.

– свойства алгоритма;

После небольшой игровой части, рассматриваются свойства алгоритма, такие как:

- 1) свойство дискретности;
- 2) свойство понятности;
- 3) свойство определенности;
- 4) свойство результативности;
- 5) свойство массовости.

Свойства алгоритма представлены на рисунке 8.

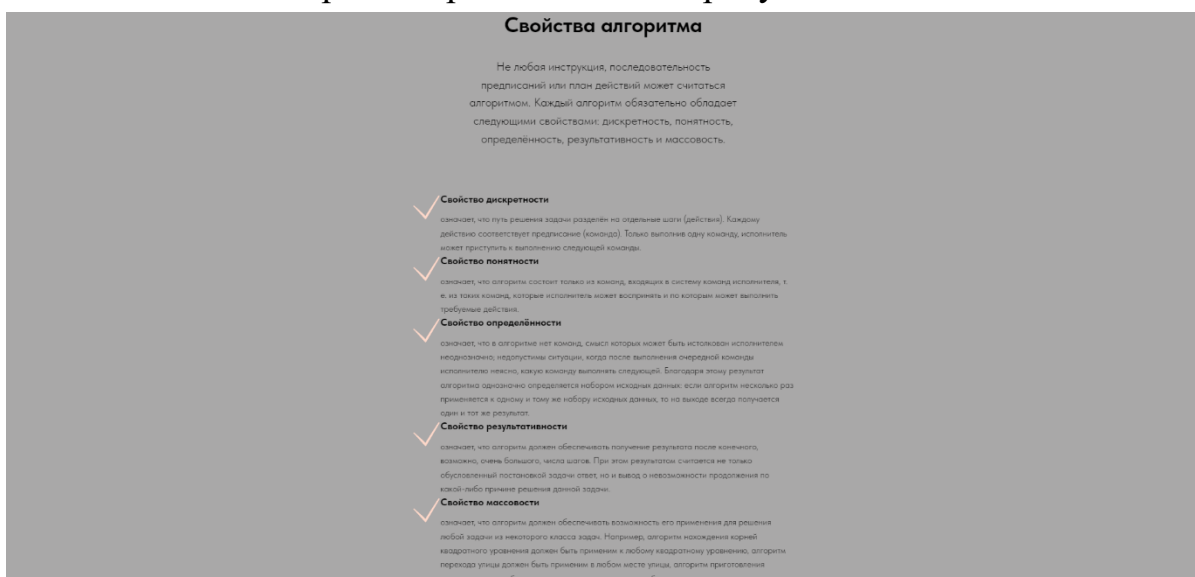


Рисунок 8 – Свойства алгоритма

– задание на составление алгоритма для решения конкретной задачи;

Теоретическая часть урока закончена. Завершение практической части урока представлено на рисунке 9.

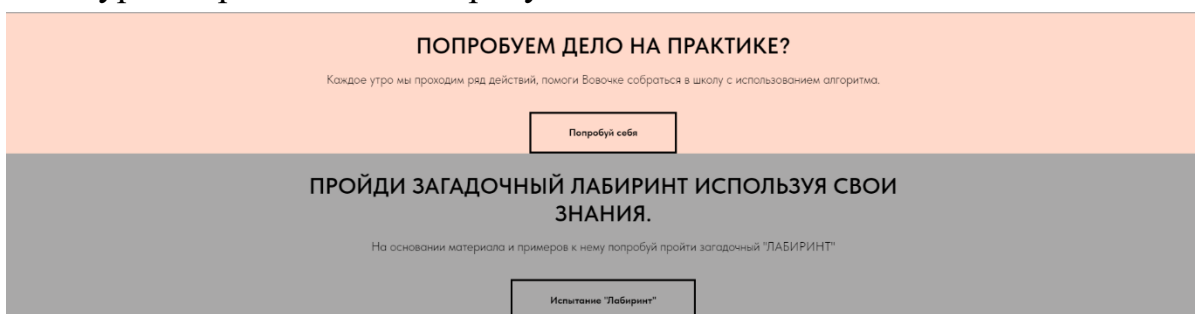


Рисунок 9 – Практические задания

Для окончательного подведения итогов, мною было создано интерактивное мультимедийное упражнение с помощью сайта «Learningapps» [12]. Задание упражнения представлено на рисунке 10.

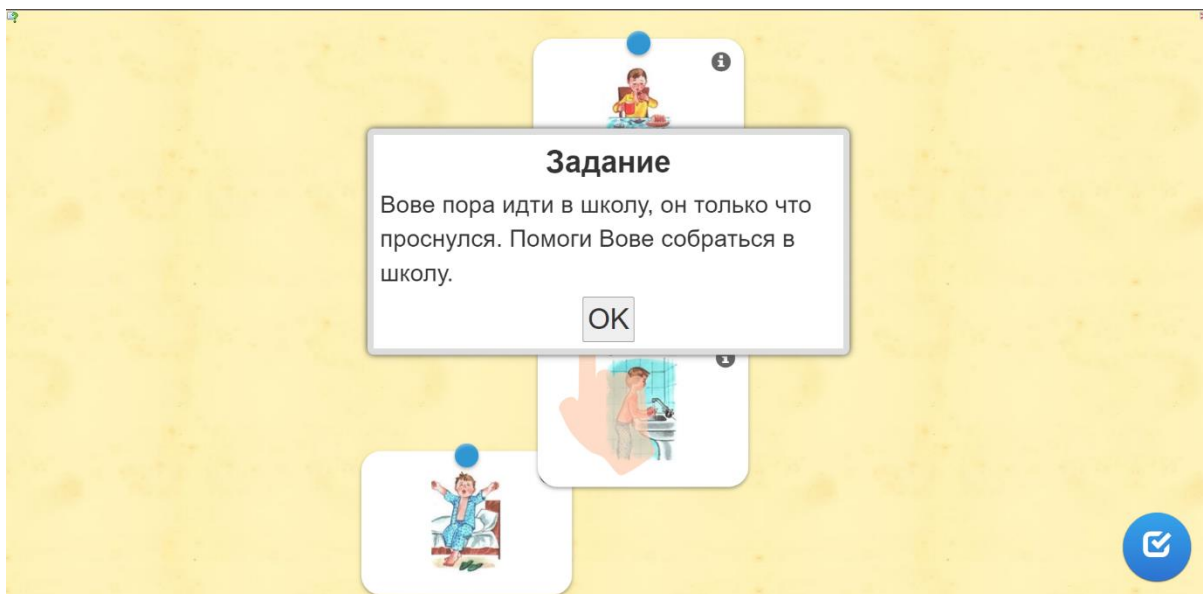


Рисунок 10 – Интерактивное мультимедийное упражнение – задание. Чтобы выполнить задание, необходимо перетащить картинки в правильном порядке, затем нажать на галочку в правом нижнем углу. Если задание выполнено верно – карточки загорятся зеленым цветом.

Правильное выполнение задание изображено на рисунке 11 [11].

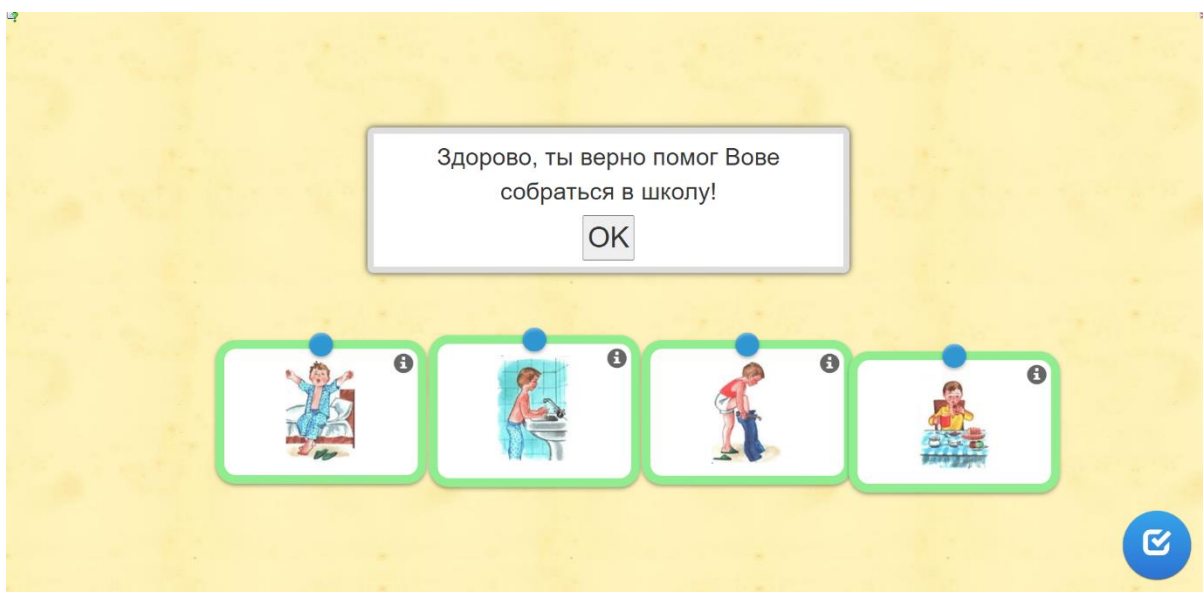


Рисунок 11 – Правильное выполнение упражнения

Упражнение «прохождение лабиринта», выполненное при помощи сайта «Игра-интернет.РФ», представлено на рисунке 12 [5].

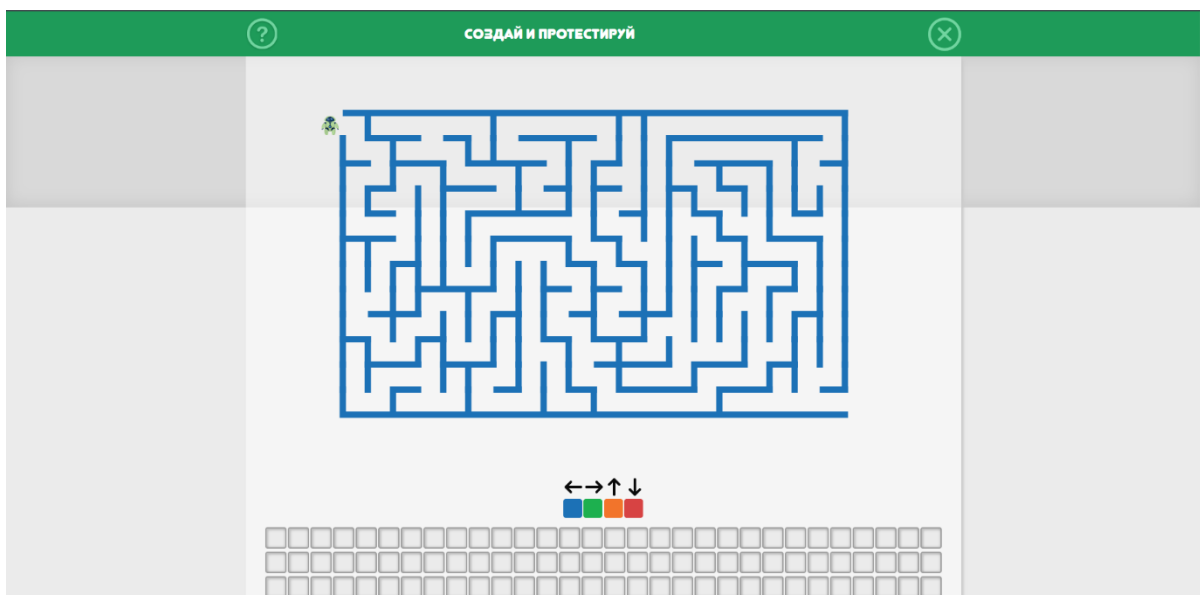


Рисунок 12 – Упражнение «Прохождение лабиринта»

Данное упражнение предназначено для игрового закрепления пройденного материала в качестве необязательного домашнего задания. Необходимо при помощи стрелок разного цвета довести робота до конца лабиринта. Как показывает практика – ученикам очень нравится данное упражнение, за счет простоты и увлеченности в процесс.

- обсуждение результатов в группах.

5. Заключение:

- подведение итогов урока;
- домашнее задание: составить алгоритм для решения задачи из повседневной жизни и принести на следующий урок.

Материалы, используемые для проведения урока на тему «Алгоритмы и исполнители»:

- 1) веб-сайт с визуальными пособиями;
- 2) доска и маркеры – для объяснения основных понятий и примеров алгоритмов;
- 3) компьютеры – для решения практических заданий.

Разработанный веб-сайт, во-первых, позволяет учителю наглядно и доходчиво предоставить материал, причём данный сайт может быть использован как на уроке, так и при подготовке к нему. Так же он может быть полезен при подготовке раздаточного материала.

Во-вторых, цифровой образовательный ресурс дает больше возможности для выполнения самостоятельной работы учеников. Учащиеся могут пользоваться веб-сайтом при подготовке домашних заданий, к проверочным и контрольным работам.

В-третьих, способом использования веб-сайта является дистанционное изучение материала, дающее возможность учащимся и родителям, самостоятельно ознакомиться с теоретическим материалом.

В-четвертых, использование веб-сайта является хорошим способом отвлечься от теоретического освоения материала, так как в нем присутствует игровая часть.

Основой данного цифрового образовательного ресурса, является разнообразие представления материала, но не полной его замены. Так как тема «алгоритмы и исполнители» является очень важной основой для дальнейшего изучения информатики, необходимо, чтобы каждый ученик понял суть алгоритмов. Метод визуализации и игры позволяет ученикам лучше понять пройденный материал.

Следующая тема раздела «Основ алгоритмизации» в 8 классе называется: «Способы записи алгоритмов». Доступ к разработанному цифровому образовательному ресурсу по теме урока доступен по ссылке URL: <http://nigametuanova.tilda.ws/sposobyzapisialgoritmov>.

Тема урока: «Способы записи алгоритмов» 8 класс.

Урок составлен по учебнику «ФГОС ООО. Информатика. 8 класс» авторы: Босова Л.Л., Босова А.Ю.

Цель урока:

- 1) познакомить учащихся с основными способами записи алгоритмов;
- 2) обучить учащихся правильно описывать последовательность действий в алгоритме, используя различные способы записи;
- 3) показать, как использование различных способов записи алгоритмов может помочь при решении задач и программировании;

4) сформировать у учащихся навыки анализа и создания алгоритмов с использованием различных методов записи.

Ход урока:

1. Введение:

- приветствие учащихся;
- обсуждение пройденного материала на прошлом уроке: что такое алгоритм? Что такое исполнитель алгоритма? Свойства алгоритма?
- переход на цифровой образовательный ресурс по ссылке.

При открытии веб-сайта, в шапке указана тема урока «Способы записи алгоритмов», далее перечисленные основные способы записи алгоритмов.

Начальная страница веб-сайта представлена на рисунке 13.



Рисунок 13 – Начальная страница веб-сайта

2. Теоретическая часть:

Основные способы записи алгоритмов:

- словесный способ;

Первый способ записи алгоритма – это словесная запись. Интерфейс словесного способа записи алгоритма представлен на рисунке 14.

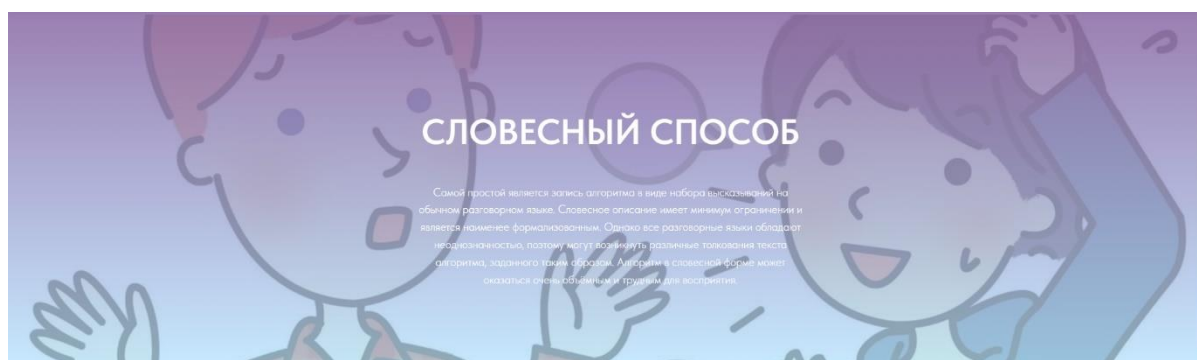


Рисунок 14 – Интерфейс словесного способа записи алгоритма

Существует 2 варианта словесной записи алгоритма:

- 1) словесное описание;
- 2) построчная запись.

Каждый из вариантов подробно описан с предоставлением примером.

Для наглядного представления примеры словесной и построчной записи представлен на рисунке 15.

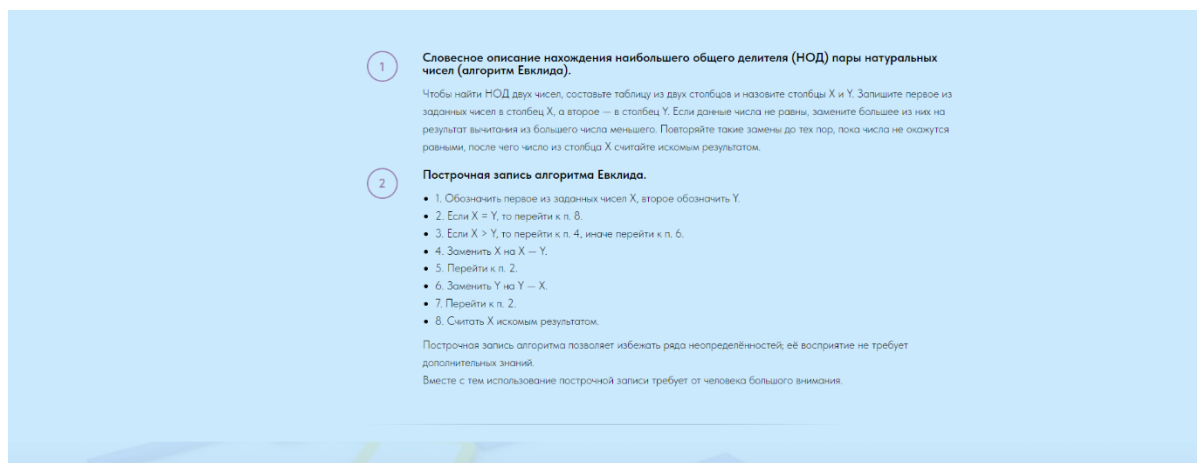


Рисунок 15 – Пример словесной и построчной записи алгоритма – графический способ;

Второй способ записи алгоритма – это графический способ.

Интерфейс графического способа представлен на рисунке 16.

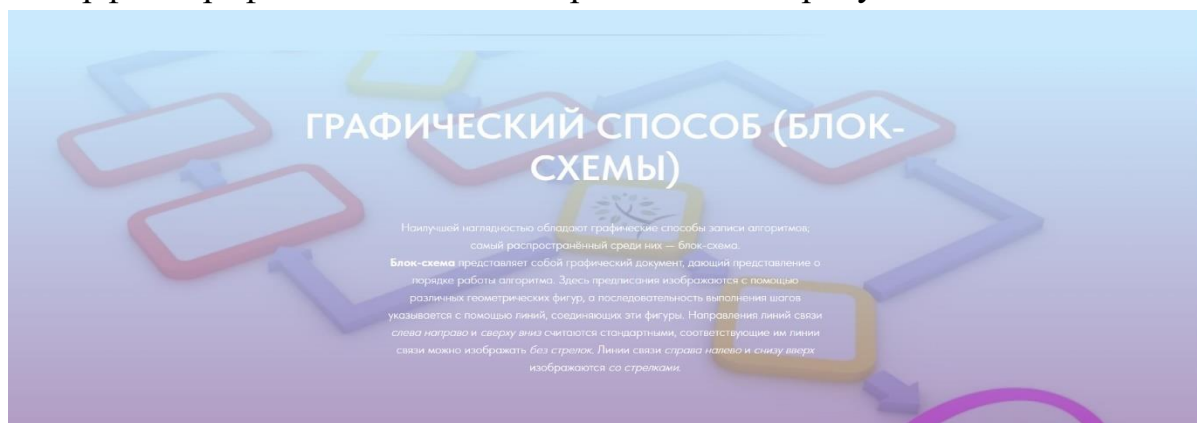


Рисунок 16 – Интерфейс графического способа записи алгоритма

Графический способ отличается от словесного не только визуально, но и тем, что при графическом способе, каждый блок имеет свой вид. Например, для обозначения начала или конца алгоритма – блок имеет овальную форму, а для обозначения обработки данных – прямоугольную. Описание и обозначение каждого графического блока алгоритма представлен на рисунке 17.

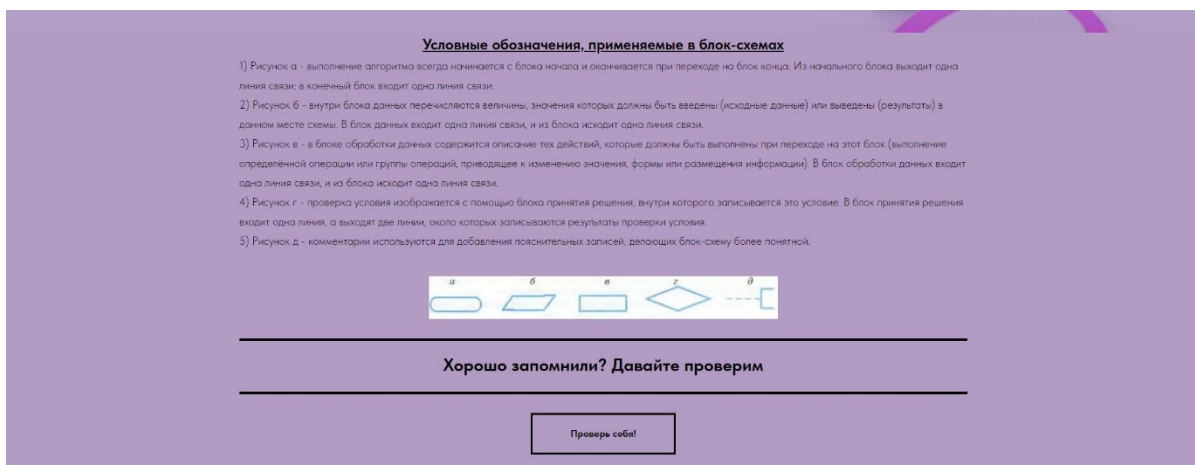


Рисунок 17 – Описание и обозначение графических блоков

Так как при графическом способе записи алгоритма важно, чтобы каждый блок имел свою геометрическую форму, учащимся необходимо выучить форму записи каждого блока. Для более эффективного запоминания данной теоретической информации мною было создано интерактивное мультимедийное упражнение с помощью сайта «Learningapps».[5] Для перехода к заданию, необходимо нажать на кнопку «Проверь себя!», которая находится под теоретической информацией.

Задание упражнения представлено на рисунке 18.

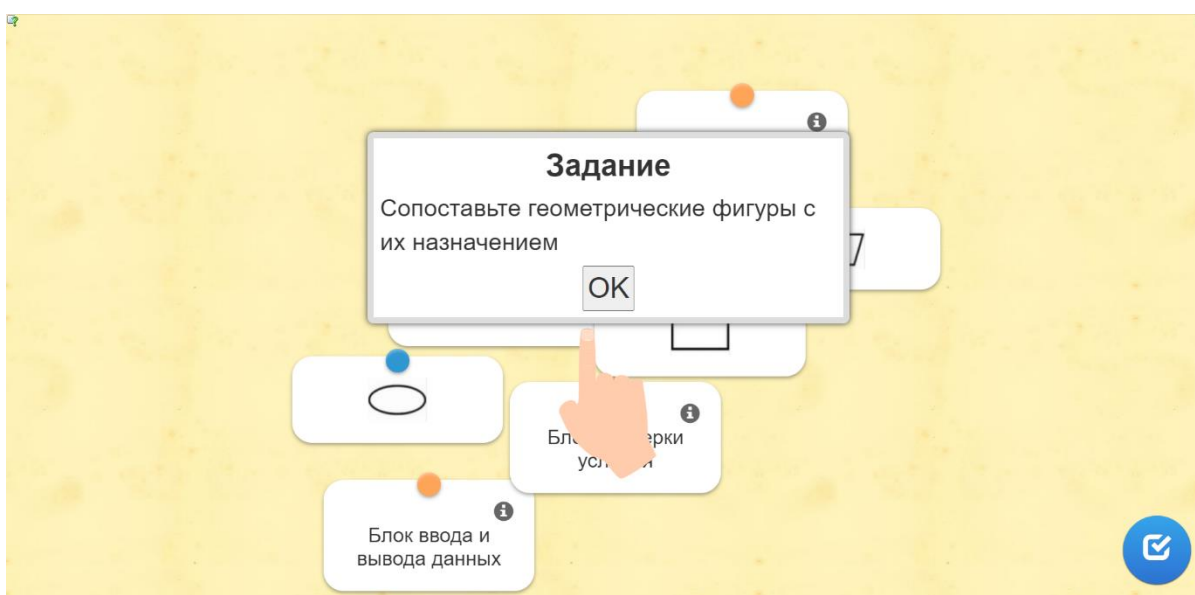


Рисунок 18 – Интерактивное мультимедийное упражнение

Если ученик забыл, какая геометрическая фигура описывает то или иное название блок, для получения подсказки, необходимо навести курсор мыши на значок «i». Подсказка представлена на рисунке 19.

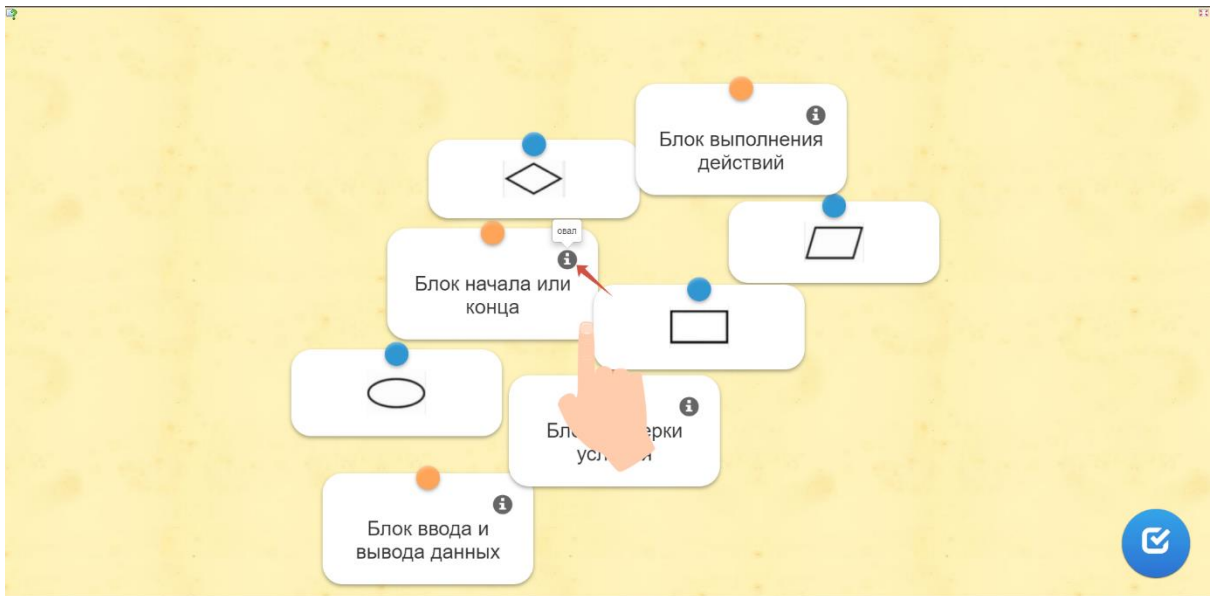


Рисунок 19 – Подсказка к выполнению задания

Готовый правильный вариант выполнения задания представлен на рисунке 20.

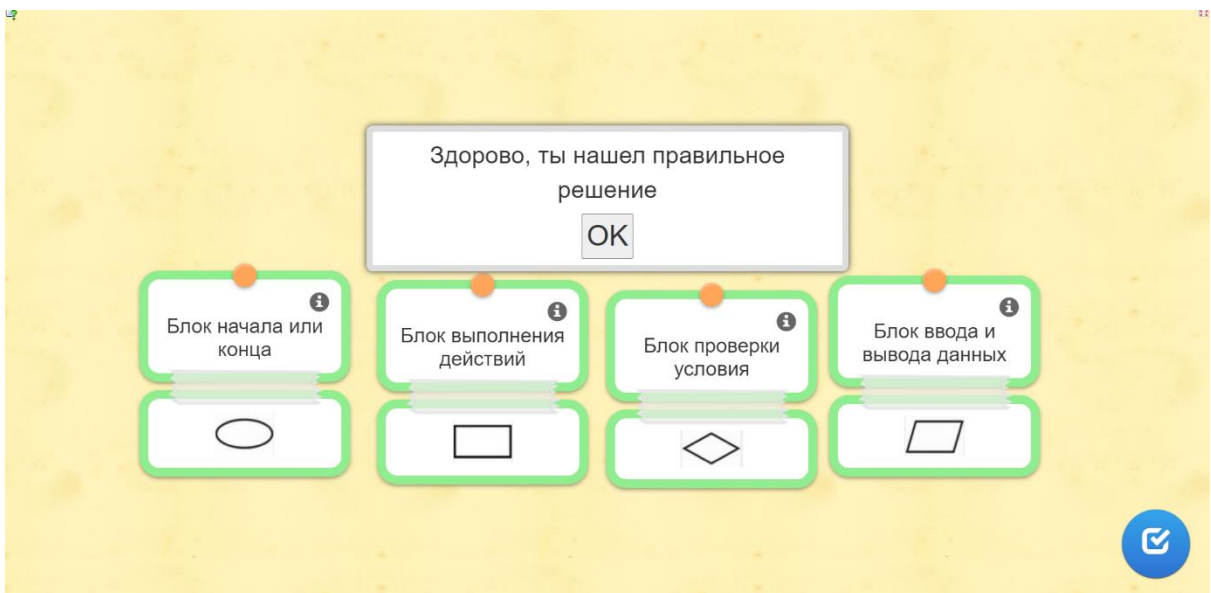


Рисунок 20 – Правильно выполненное задание

Если ученик правильно выполнил задание, то пары обретут зеленую рамку. Если хотя бы 1 пара будет сопоставлена не верно – рамки пар окрасятся в красный цвет, необходимо будет определить пары еще раз.

Данное интерактивное задание не только поможет ученикам лучше усвоить данный материал, но и увлечет в процесс изучения темы, за счет игровой формы выполнения.

Ранее разобранный пример записи алгоритма Евклида можно так же представить в графическом исполнении. На рисунке 21 представлен графический способ записи данного алгоритма.

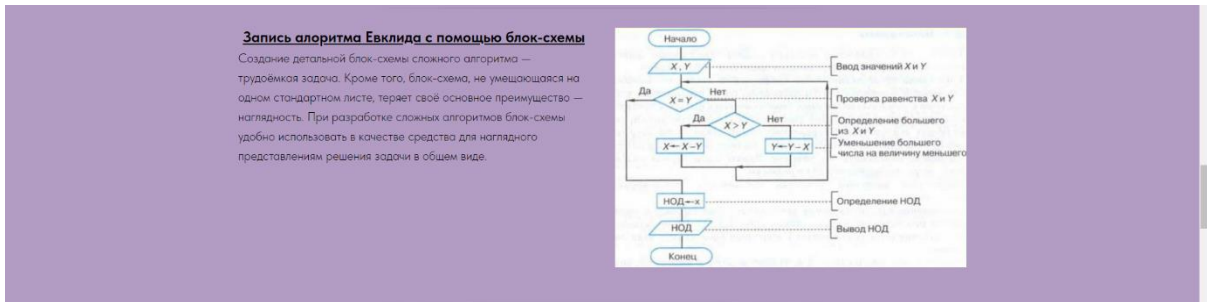


Рисунок 21 – Графическая записи алгоритма Евклида

– алгоритмические языки, или по-другому программный способ записи алгоритма.

Последний разбираемый способ записи алгоритма – это программный способ.

Описание программного способа записи алгоритма представлен на рисунке 22.

**АЛГОРИТМИЧЕСКИЕ ЯЗЫКИ
(ПРОГРАММНЫЙ СПОСОБ)**

```
function sortBubble(data) {
  var tmp;
  for (var i = data.length - 1; i > 0; i--) {
    var counter=0;
    for (var j = 0; j < i; j++) {
      if (data[j] > data[j+1]) {
        tmp = data[j]; data[j] = data[j+1];
        data[j+1] = tmp; counter++;
      }
    }
    if(counter==0){ break; }
  }
  return data;
};
```

JavaScript

C++

```
int bubble(int* a, int n) {
  for (int i = n - 1; i >= 0; i--) {
    for (int j = 0; j < i; j++) {
      if (a[j] > a[j + 1]) {
        int tmp = a[j];
        a[j] = a[j + 1];
        a[j + 1] = tmp;
      }
    }
  }
}
```

Алгоритмические языки — формальные языки, предназначенные для записи алгоритмов. Каждый из них характеризуется:

- алфавитом — набором используемых символов;
- синтаксисом — системой правил, по которым из символов алфавита образуются правильные конструкции языка;
- семантикой — системой правил, строго определяющей смысл и способ употребления конструкций языка.

Класс алгоритмических языков очень широк. При изучении курса информатики в школах используются различные версии школьного (учебного) алгоритмического языка.

Рисунок 22 – Описание программного способа записи алгоритма

Программный способ записи алгоритма – это запись алгоритма на языке программирования, позволяющем на основе строго определенных правил формировать последовательность предписаний, однозначно отражающих смысл и содержание алгоритма, с целью его последующего исполнения на компьютере.

В школьной программе изучение записи алгоритма на алгоритмическом языке происходит с помощью школьного алгоритмического языка.

Объяснение – что такое «Школьный алгоритмический язык» и его пример представлен на рисунках 23 и 24.

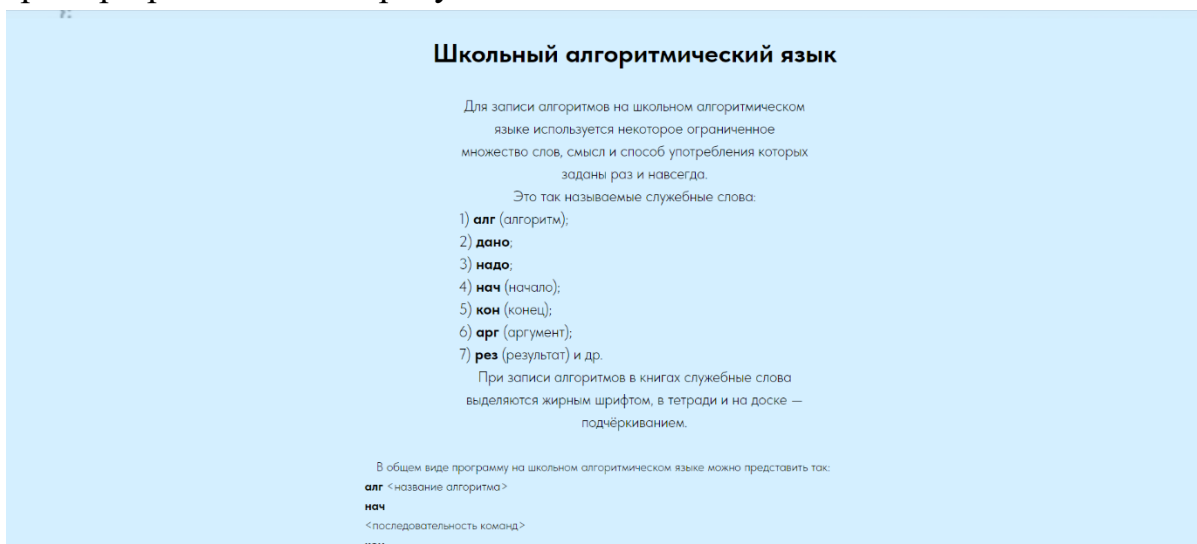


Рисунок 23 – Школьный алгоритмический язык

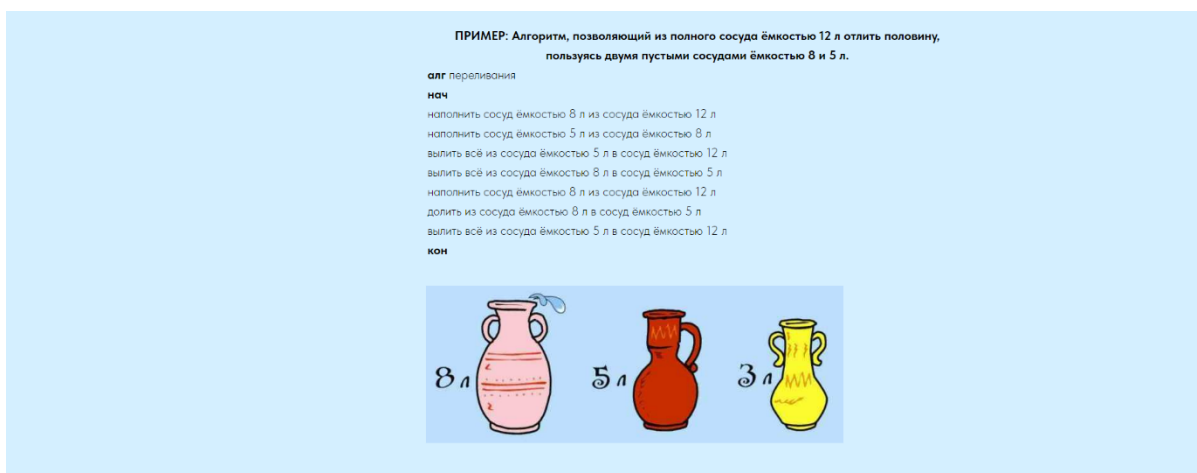


Рисунок 24 – Пример школьного алгоритмического языка

После изучения теоретической части урока, для закрепления полученных знаний, необходимо выполнить созданное интерактивное мультимедийное упражнение с помощью сайта «Learningapps».[5]

Упражнение представлено на рисунках 25, 26 и 27.

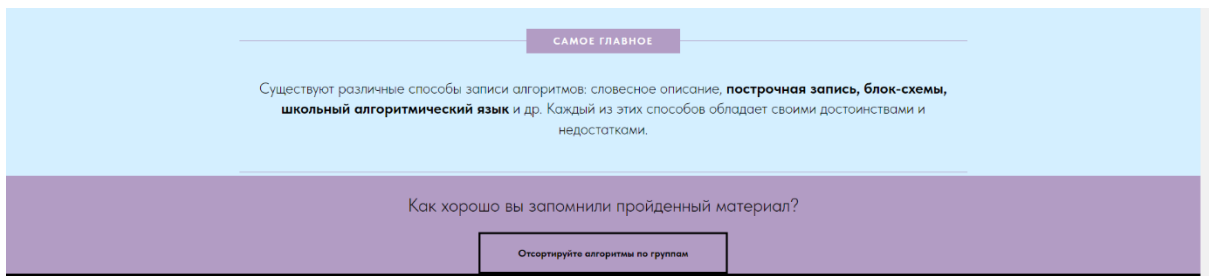


Рисунок 25 – Интерфейс перехода к интерактивному упражнению

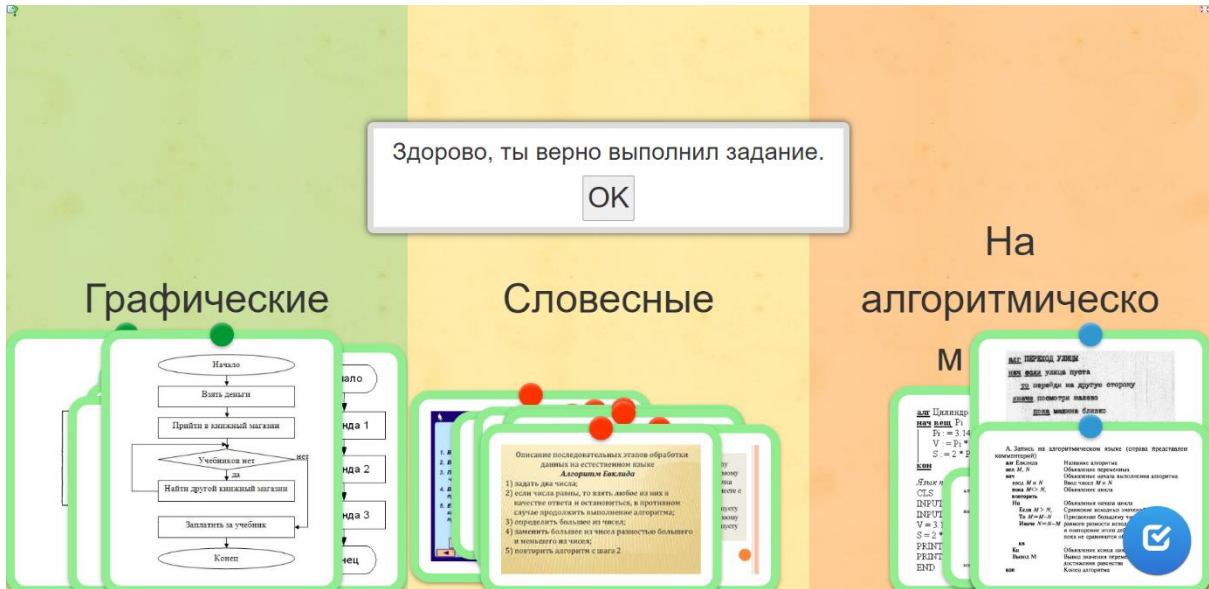


Рисунок 26 – Интерактивное мультимедийное упражнение – задание

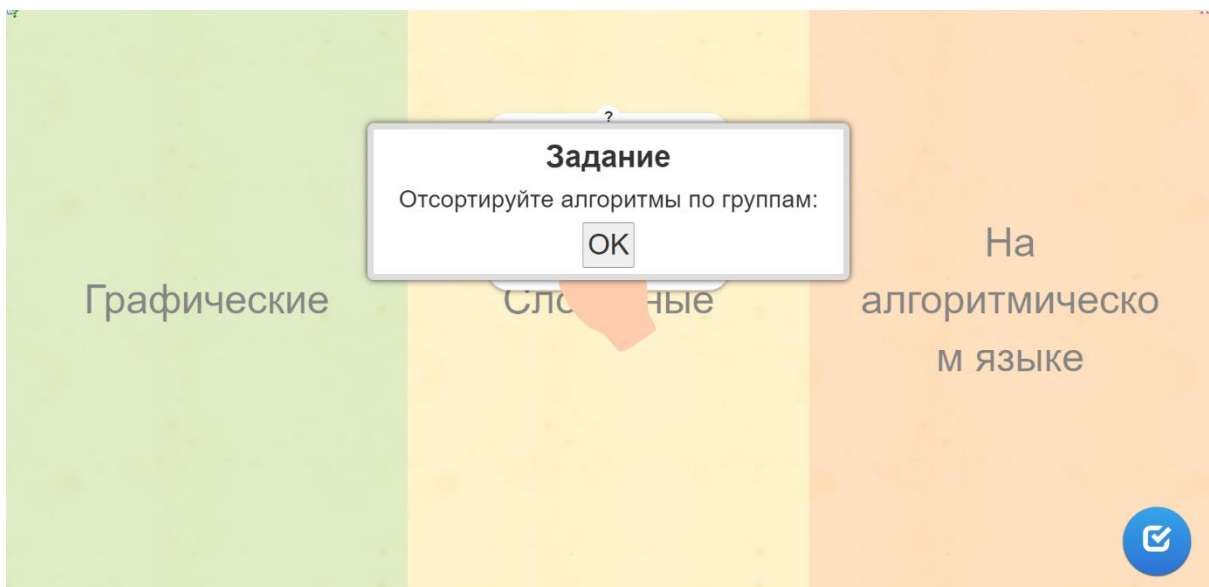


Рисунок 27 – Правильно выполненное интерактивное задание

5. Заключение:

– подведение итогов урока.

Материалы, используемые для проведения урока на тему «Алгоритмы и исполнители»:

- 1) веб-сайт с визуальными пособиями;

2) доска и маркеры – для объяснения основных понятий и примеров записи алгоритма;

3) компьютеры – для решения практических заданий.

Анализ результатов проведения уроков «Алгоритмы и исполнители» и «Способы записи алгоритмов» с учетом методических особенностей на уроках по информатике:

1. Введение – приветствие учащихся и объяснение темы урока создали благоприятную атмосферу для обучения.

2. Теоретическая часть – объяснение основных понятий и примеры алгоритмов из повседневной жизни помогли учащимся разобраться в теме урока.

3. Практическая часть – разбиение на группы и задания на составление алгоритма позволили учащимся применить полученные знания на практике. Обсуждение результатов в группах способствовало обмену опытом.

4. Методические особенности – использование интерактивных методов обучения и визуальных пособий улучшило усвоение материала и удержание внимания учащихся.

5. Заключение – подведение итогов и задание домашнего задания закрепило полученные знания.

В заключении необходимо отметить, что уроки были организованы эффективно, и структура уроков позволила достичь поставленной цели.

Апробация:

В рамках педагогической практики была проведена апробация разработанных уроков в Муниципальном бюджетном общеобразовательном учреждении «Школа № 14» села Кочкарь.

В рамках урочной деятельности было проведено 2 занятия в 8 классе по разделу «Основы алгоритмизации», а именно:

1) алгоритмы и исполнители;

2) способы записи алгоритмов.

Каждый урок сопровождался веб-сайтом, который находится в свободном доступе при переходе по ссылке.

Апробация прошла успешно. Все учащиеся усвоили учебный материал, о чем говорит высокий процент выполнения заданий в рамках промежуточной аттестации.

Результаты исследования представлены ниже:

1. Эффективность ресурса: в процессе апробации было выявлено, что цифровые образовательные ресурсы на тему «Алгоритмы и исполнители» и «Способы записи алгоритмов» оказались эффективными инструментами для обучения учащихся. Ученики проявили интерес к изучаемым темам, активно взаимодействовали с ресурсами и продемонстрировали хорошие результаты в выполнении заданий.

2. Удобство использования: учителя отметили удобство использования цифровых ресурсов в учебном процессе. Интерфейсы ЦОРов интуитивно понятны как для учителей, так и для учащихся, что способствовало более эффективному взаимодействию с материалом.

3. Обратная связь: собранная обратная связь от учителей и учащихся позволила выявить как положительные стороны ресурсов (например, интерактивность, доступность информации), так и возможные улучшения (например, добавление дополнительных заданий или разнообразных форматов).

4. Рекомендации: на основе результатов апробации можно сделать ряд рекомендаций по дальнейшему развитию и улучшению цифровых образовательных ресурсов. Например, добавление дополнительных модулей, расширение функционала или улучшение адаптивности под разные уровни обучения.

5. Рекомендации для учителей информатики по применению методических особенностей:

1) использование конкретных примеров из повседневной жизни, могут помочь ученикам понять алгоритмы. Например, алгоритм приготовления кофе или поиска информации в интернете;

2) проведение практических занятий, на которых ученики смогут самостоятельно создавать и тестировать свои алгоритмы, поможет учащимся лучше усвоить материал;

3) использование игр и задач с логическими элементами для обучения алгоритмизации – помогают ученикам развить логическое мышление и умение решать сложные задачи;

4) объяснение основных понятий и принципов алгоритмизации доступным и понятным языком, избегая сложных терминов и формулировок, позволяет ученикам не потерять интерес к изучению данной темы;

5) поощрение учеников к творческому мышлению и поиску нестандартных решений задач, помогает им осознать, что существует несколько способов решения одной и той же задачи;

6) использование различных методов оценки знаний учеников, такие как тесты, проекты, практические задания – поможет вам оценить уровень понимания материала каждым учеником;

7) поддержание открытого диалога с учениками, стимулирует их задавать вопросы;

8) позволение ученикам работать в группах или партнерстве над задачами по созданию алгоритмов, способствует сотрудничеству, обмену идеями и развитию коммуникативных навыков.

Выводы по главе 2

В главе 2 дипломной работы были рассмотрены методические особенности при изучении основ алгоритмизации на уроках в школе. Был проведен анализ существующих методик изучения алгоритмизации и

выявлены основные принципы, которые должны учитываться при разработке уроков по данной теме.

Были разработаны уроки на тему «Алгоритмы и исполнители» и «Способы записи алгоритмов» с учетом особенностей учащихся и их уровня подготовки. А также было уделено особое внимание выбору подходящих методов обучения, использованию интерактивных методов и средств обучения.

ЗАКЛЮЧЕНИЕ

Обобщение результатов исследования: результаты исследования показывают, что изучение раздела «Основы алгоритмизации» с учетом названных в работе методических особенностей на уроках информатики в школе способствует эффективному обучению учащихся.

В частности, использование интерактивных методов обучения, применение практических заданий и проектной деятельности, а также индивидуальный подход к каждому ученику способствуют лучшему усвоению материала. Кроме того, важно учитывать уровень подготовки учащихся, их интересы и способности при планировании занятий по алгоритмизации.

В целом, оптимальное сочетание теоретического материала с практическими заданиями и активными формами работы помогает достичь более высоких результатов в обучении основам алгоритмизации на уроках информатики.

Использование интерактивных методов обучения, таких как обсуждение вопросов с учащимися, групповая работа, игровые задания и т.д. – позволяют активизировать учебный процесс, повысить вовлеченность учащихся и помочь им лучше усвоить материал.

Применение практических заданий и проектной деятельности – позволяют учащимся применять полученные знания на практике, развивать навыки решения задач и креативное мышление.

Индивидуальный подход к каждому ученику, учитывая его уровень подготовки, интересы и способности – позволяет адаптировать материал к потребностям каждого ученика и обеспечить более эффективное обучение.

Сочетание теоретического материала с практическими заданиями и активными формами работы, такими как лабораторные работы, создание алгоритмов на компьютере, решение задач на программирование – помогает

закрепить теоретические знания на практике и развить умения применять их в реальных ситуациях.

Важно обеспечить структурированное и последовательное изучение основ алгоритмизации, начиная с простых концепций и постепенно переходя к более сложным задачам, чтобы ученики могли постепенно развивать свои навыки.

Систематическое использование различных методов контроля и оценки знаний позволяет учителю отслеживать прогресс каждого ученика и корректировать учебный процесс в соответствии с его потребностями.

Важно создавать поддерживающую и стимулирующую обстановку на уроках информатики, чтобы мотивировать учеников к активному участию в учебном процессе и развитию своих навыков алгоритмизации.

В ходе выполнения выпускной квалифицированной работы на тему «Методические особенности изучения основ алгоритмизации на уроках информатики» были рассмотрены основные принципы организации образовательного процесса по данной теме. Были выявлены основные методические подходы к обучению алгоритмизации, рассмотрены современные методики и технологии, способствующие эффективному усвоению материала.

Анализ показал, что для успешного изучения основ алгоритмизации на уроках информатики в школе необходимо использовать комбинацию различных методов обучения, таких как интерактивные занятия, практические задания, групповые проекты и т.д. Также важно учитывать индивидуальные особенности учащихся и создавать условия для развития их творческого мышления.

В результате исследования были разработаны рекомендации по оптимизации процесса обучения алгоритмизации на уроках информатики в школе. Предложенные методические подходы и технологии могут быть использованы учителями для повышения качества образования и

формирования у школьников навыков программирования и решения алгоритмических задач.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Босова Л. Л. Методика преподавания курса информатики и ИКТ в 5 – 9 классах (по УМК Босовой Л. Л. по информатике и ИКТ): метод. пособие / Л. Л. Босова. – Москва : [б. и.], 2012. – 146 с. – ISBN 978-5-9963-1462-1
2. Босова Л. Л. Информатика 7–9 классы.: базовый уровень: метод. Пособие к учебникам / Л. Л. Босова, А. Ю. Босова. – Москва : Просвещение, 2022. – 69 с. – ISBN 978-5-09-094379-6
3. Давыдова Л. В. Аттестационная работа. Методика изучения основ алгоритмизации и программирования: веб-ресурс. – Майский, 2020 – . – URL: <https://znanio.ru/media/attestatsionnaya-rabota-metodika-izucheniya-osnov-algoritmizatsii-i-programmirovaniya-2585580> (дата обращения 10.12.2023).
4. Златопольский Д. М. Интеллектуальные игры в информатике.: учеб. пособие / Д. М. Златопольский. – Санкт-Петербург : БХВ-Петербург, 2004. – 400 с.: ил. – ISBN 5-94157-505-X.
5. Интернет-игра.рф – изучи интернет – управляй им : интерактивный проект. – Москва. – Обновляется в течении суток. – URL: <https://игра-интернет.рф/games> (дата обращения 23.12.2023).
6. Лапчик М. П. Методика преподавания информатики.: учеб. пособие / М. П. Лапчик – Москва : Академия, 2001. –624 с. – ISBN 5-7695-0825-6.
7. Мациевский С. В. Учебное пособие.: учеб. пособие / С. В. Мациевский, С. А. Ишанов, С. В. Клевцур – Калининград: КГУ, 2003. – 140 с. – ISBN 5-88874-385-2.
8. Место и содержание в школьном курсе информатики алгоритмизации : веб-ресурс. – Пермь, 2004 – . – URL: https://lbz.ru/metodist/authors/informatika/2/files/Algoritmizatsiya_i_programmirovaniye.pdf (дата обращения 17.12.2023).

9. Панкратова Л. П. Контроль знаний по информатике: тесты, контрольные задания, экзаменационные вопросы, компьютерные проекты.: метод. пособие / Л. П. Панкратова, Е. Н. Челак – Москва : БХВ-Петербург, 2004. – 448 с. – ISBN 5-94157-371-5.
10. Потапов Ю. В. Компьютерная грамота для начинающих : веб-ресурс. – Томск, 2001 – . – URL: <https://znanio.ru/media/kompyuternaya-gramota-dlya-nachinayuschih-tomskij-gu2001pdf-2563466> (дата обращения 15.12.2023).
11. Семакин И. Г. Информатика. Программа для основной школы 7 – 9 классы.: метод. пособие / И. Г. Семакин, М. С. Цветкова – Москва : Бином, 2013. – 184 с.: табл. – ISBN 978-5-9963-1468-3/
12. Создание мультимедийных интерактивных упражнений : мультимедийный, интерактивный проект. – Майнц, 2012 – . – URL: <https://learningapps.org/> (дата обращения 23.12.2023).