Муниципальное бюджетное образовательное учреждение дополнительного образования «Чувашская воскресная школа им. П. М. Миронова» городского округа город Уфа Республики Башкортостан

г. Уфа, Кировский район, ул. Пушкина, д. 110

ОКПО 71852573, ОГРН 1030203925510, ИНН 0274080343 / КПП 027401001

Утверждаю
директор МБОУ ДО
ЧВШ им. П.М. Миронова
_____ И.Г. Тарасов
Приказ №48 от 13.06.2019 г.

Дополнительная общеобразовательная общеразвивающая программа технической направленности «Компьютер – инструмент творчества (языки программирования)»

Программа рассчитана
на возраст детей 10-17 лет.
Срок реализации: 3 года.
Обучение очное.
Уровень сложности: базовый.
Вид программы: модифицированный.

Составитель:
Кондратьев Денис Васильевич,
кандидат физико-математических наук,
педагог дополнительного образования 1категории
МБОУ ДО ЧВШ им. П.М. Миронова.

Рекомендовано к утверждению
Педагогическим советом
Протокол № 3 от 31.05.2019 г.

Пояснительная записка

Формирование алгоритмической культуры является одной из актуальных задач последнего времени.

Развитие логики и алгоритмического мышления при использовании фундаментальных понятий программирования соответствует плану мероприятий по реализации Концепции математического образования в РФ (Приказ Минобрнауки России № 265 от 3 апреля 2014 г.).

Сегодня наука и технология развиваются столь стремительно, что образование зачастую не успевает за ними. Например, для того, чтобы успешно выступать на Российских олимпиадах по информатике, надо серьёзно заниматься хотя бы с седьмого, а ещё лучше — с шестого или даже пятого класса. Ведущие вузы и фирмы, работающие в области информационных технологий, — МГУ, МФТИ, МИФИ, Intel и др. — стараются привлечь школьников к научной деятельности, причём со всё более раннего возраста. Для этого проводятся конкурсы, викторины, олимпиады и другие мероприятия. Наш опыт в подготовке детей к такой работе показывает, что начиная с некоторого момента (совпадающего приблизительно с 4–5 классом) интересы ребят претерпевают существенные изменения и у большинства из них сильно удаляются от учебной деятельности вообще и научно-познавательной — в частности. Это можно объяснить разными причинами, но одной из наиболее серьёзных таких причин, несомненно, является неуспешность наших детей в учебной деятельности или боязнь такой неуспешности. В такой ситуации одной из важнейших задач педагогов следует считать создание комфортной учебно-воспитательной среды, в которой возможна наиболее полная самореализация ребёнка.

Другой причиной снижения интереса учеников, имеющей отношение собственно к информатике, является очень небольшое количество часов, предусмотренное федеральными образовательными стандартами на изучение данного предмета. Более того, стандарты второго поколения вообще не включают дисциплину «Информатика», заменяя её «Информационными технологиями». Основной акцент предполагается сделать на приобретении учениками информационной грамотности, подразумевающей общие навыки обработки информации различных видов. Теоретическая же подготовка по информатике вынесена во внеучебную деятельность и, таким образом, носит факультативный характер.

Изучение курса «Компьютер – инструмент творчества» является одним из возможных подходов к организации проектной внеучебной образовательной деятельности по информатике в средней школе. Такая деятельность позволяет, с одной стороны, организовать среду для самореализации и самоутверждения учеников, и, с другой стороны, сформировать у них тягу к творчеству и знаниям и дать подходящие средства её реализации. Предлагаемый курс посвящен пропедевтике объектно-ориентированного программирования, а также элементов (идей) параллельного программирования на первой и второй ступенях средней общеобразовательной

школы (соответственно, 1-4 и 5-9 классы) при помощи языка программирования Scratch. Интерес именно к параллельному программированию вызван, во-первых, широким распространением этой технологии, а во-вторых, — наличием подходящего инструмента для изучения соответствующих техник.

Подготовка школьников к изучению программирования во многих странах мира начинается с различных программных сред, одна из которых – программная среда SCRATCH, разработанная MIT (http://scratch.mit.edu/).

При дальнейшем изучении языков программирования высокого уровня рассматривается Python. Приобретенные навыки программирования могут быть в дальнейшем использованы при подготовке как к ОГЭ-9, так и ЕГЭ-11.

Данный курс ориентирован на обучающихся 4-7 классов и продолжает развитие логических способностей школьников, знакомство с основными алгоритмическими конструкциями и ориентирован также на получение первоначальных навыков программирования на языке высокого уровня Python.

Программа может быть использована как дополнительный модуль к основной программе по информатике при использовании часов учебного плана из предметной области технология.

Ценностные ориентиры учебного предмета «Компьютер – инструмент творчества» связаны:

с нравственно-этическим поведением и оцениванием, предполагающем, что обучающийся знает и применяет правила поведения в компьютерном классе и этические нормы работы с информацией коллективного пользования и личной информацией; выделяет нравственный аспект поведения при работе с информацией;

с возможностью понимания ценности, значимости информации в современном мире и ее целесообразного использования, роли информационно-коммуникативных технологий в развитии личности и общества;

с развитием логического, алгоритмического и системного мышления, созданием предпосылок формирования компетентности в областях, связанных с информатикой, ориентацией обучающихся на формирование самоуважения и эмоционально-положительного отношения к окружающим.

Актуальность.

В рамках, отводимых «Программой» в базовом курсе информатики на алгоритмизацию и программирование, овладение даже основами программирования на современных алгоритмических языках представляется невозможным. Тем не менее, контингент школьников, у которых интерес именно к изучению, а не знакомству с программированием высок, несомненно, существует. В первую очередь, это учащиеся физико-математических классов, гимназий, лицеев и гимназических классов общеобразовательных школ. У большинства из них есть как мотивация, так и способности к освоению программирования. Учебные планы подобных образовательных учреждений, в которых на освоение информатики и информационных технологий отводится не менее часа, начиная с 5-го класса, не менее двух – с 8-го и до четырех в 10-

11-х классах, также играют положительную роль. Мотивация есть и у учителя – ведь большинство современных олимпиад по информатике являются по своей сути олимпиадами по программированию, а по успехам учеников в олимпиадах зачастую судят о квалификации учителя, хотя в случае с информатикой это далеко не бесспорно.

Начинать преподавание программирования можно как в 7-8-х классах, так и в 9-10-х классах, при этом незначительно меняются задачи, решаемые на уроках, которые должны быть адаптированы к уровню математической подготовки учащихся.

В рамках курса «Компьютер – инструмент творчества» предлагается обучать основам программирования уже с 4-5 класса, что приведет к лучшему пониманию курса «Информатика и ИКТ» в общеобразовательной школе и будет являться своего рода пропедевтикой. Предлагаемый курс будет подготавливать детей к изучению довольно сложного учебного модуля «Основы алгоритмизации и программирования» в случае обучающихся 10-13 лет, дополнять, расширять и углублять знания по программированию в случае детей 13-17 лет.

Новизна данной программы

1. Курс дисциплины «Компьютер – инструмент творчества» предполагает с раннего школьного возраста обучать основам объектно-ориентированного программирования.

2. Предлагаемая для изучения среда программирования Scratch предполагает решение сложных задач программирования с раннего школьного возраста: создание анимации, интерактивных приложений, простейших игр, что повышает мотивацию к изучению программирования в общеобразовательной школе.

3. Занятия, посвященные программированию, чередуются с занятиями, целью которых является повышение общей компьютерной грамотности. Общеизвестные программные продукты используются для выполнения творческих заданий.

Цель и задачи образовательной программы:

Цели:

- получить представление об алгоритмах и исполнителях, основных алгоритмических конструкциях языков программирования;
- обеспечить высокую мотивацию к проектной деятельности и дальнейшему изучению языков программирования;
- обеспечить получение первоначальных навыков программирования на языке высокого уровня;
- формировать компьютерную грамотность.

Для реализации поставленных целей обучающийся научится:

• самостоятельно устанавливать программную среду на домашний компьютер;

• использовать возможности отладки программ;

• использовать команды организации цикла для оптимизации программ исполнителей;

• составлять программы для выполнения параллельных алгоритмов;

• использовать интерактивные возможности среды Scratch для создания программ и игр;

• применять различные формы ветвления алгоритмов при моделировании ситуаций, включая цикл по условию;

• планировать и создавать анимацию заданного сюжета;

• моделировать и создавать обучающие программы, иллюстрирующие материал других предметных областей;

• моделировать интерактивное взаимодействие с исполнителями для создания простейших тренажеров;

• использовать переменные и списки для реализации творческих проектов;

• использовать творческий подход к построению моделей различных объектов и систем.

Образовательный процесс состоит из следующих частей:

- **учебная:** организуется посредством передачи теоретических знаний, выработки практических умений и навыков – мини-лекции, разбор готовых примеров, лабораторные занятия по написанию своих программ, созданию анимации и простейших игр;

- **воспитательная:** реализуется через организацию индивидуальной и коллективной деятельности, решение проблемных ситуаций, вовлечение учащихся в ситуации, требующие самостоятельности и самоорганизации;

- **развивающая**: осуществляется через активизацию познавательного интереса, выбор индивидуальной образовательной траектории для учащихся в соответствии с уровнем знаний и интересом, проявляемым к предмету.

Основные принципы организации образовательного процесса:

Курс построен с учетом принципов последовательности, преемственности, системности в формировании знаний, навыков и умений.

- обучение детей на добровольных началах;
- морально-психологический комфорт на занятиях;
- организация индивидуально-личностного и дифференцированного подхода на занятиях с учетом возрастных особенностей и личной мотивации.

Образовательные задачи:

- изучить язык программирования и создание основных алгоритмических конструкций на языке программирования;
- обучить учащихся работе в среде программирования;
- дать базовые навыки по созданию проектов в данной среде программирования.

Развивающие задачи:

- формировать у учащихся стремление к саморазвитию и личностному росту через познавательную деятельность, развивать информационную речь, пространственное воображение, логическое и системное мышление, развивать

память, внимание учащихся, развивать умение использовать полученные теоретические знания в практической деятельности.

Воспитательные задачи:

- воспитывать у учащихся личностные качества, способствующие успешной социальной адаптации: целеустремленность, объективность в самооценке, ответственность, познавательный интерес, стремление доводить начатое дело до конца, аккуратность, коммуникативность.

Межпредметные связи:

Межпредметные связи - важнейший принцип обучения в школе. Их использование способствует целостному восприятию мира, формированию основных учебных компетенций, способствующих развитию умения обнаруживать скрытые зависимости и связи, переносить ранее усвоенный материал на новый, а так же позволяет активизировать уже существующий интерес ученика к предмету или способствует развитию такого интереса.

В курсе «Компьютер – инструмент творчества» прослеживаются межпредметные связи между математикой, физикой, биологией, историей и информатикой. Ряд задач, рассматриваемых на уроках демонстрирует интеграцию литературы, русского языка и информатики.

Основными формами занятий являются:

Курс рекомендуется реализовывать как в урочной, так и во внеурочной формах деятельности обучающихся. Программа курса имеет инженерно-техническую направленность. Многие задания составлены таким образом, чтобы они решались методами учебно-исследовательской и проектной деятельности.

Большинство заданий встречаются в различных темах для того, чтобы показать возможности решения одной и той же задачи или проблемы различными средствами, обеспечивающими достижение требуемого результата, что в итоге должно привести к выработке способности оптимизировать решение поставленной задачи или проблемы.

Занятия сочетают элементы обучения с развитием художественно-творческих способностей. Каждая тема адаптируется к возрасту детей и строится с учетом их интересов, предпочтений и возможностей.

Программа рассчитана на 3 года обучения. Первое полугодие, в основном, ознакомительного характера, второе полугодие углубленное изучение. Год обучения состоит из 108 учебных часов. Занятия проводятся в течение 3 часов 1 раз в неделю в подгруппах по 4-5 человек.

В первый год обучения акцент делается на среду программирования Scratch и общую компьютерную грамотность; второй год обучения – среда программирования Python; третий год обучения – язык программирования C++.

Возраст обучающихся: в основном, от 10 до 15 лет.

Проверка результативности обучения

Личностные результаты:

– формирование ответственного отношения к учению, способности планировать и доводить до конца начатое дело благодаря завершённым творческим учебным проектам;

– формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе визуальной среды программирования Scratch и графической библиотеки Pyturtle;

– мотивации к обучению и познанию, приобретение опыта участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;

– формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;

– формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;

– формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;

– формирование ценности здорового и безопасного образа жизни, усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Метапредметные результаты:

– умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;

– умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата, понимая принципы оптимизации программ;

– умение формулировать критерии эффективности конечного продукта (программы) и критически оценивать решение учебно-исследовательской задачи;

– умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;

– владение основами самоконтроля, принятия решений;

– умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;

– формирование и развитие компетентности в области использования информационно- коммуникационных технологий (ИКТ-компетенция);

– умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Предметные результаты:

– осознание значения математики и информатики в повседневной жизни человека;

– формирование представлений об основных предметных понятиях («информация», «алгоритм», «исполнитель», «модель») и их свойствах;

– развитие логических способностей и алгоритмического мышления, умения составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами – линейной, условной и циклической;

– развитие представлений о числах, числовых системах;

– овладение символьным языком алгебры, умение составлять и использовать сложные алгебраические выражения для моделирования учебных проектов, моделировать реальные ситуации на языке алгебры;

– развитие пространственных представлений, навыков геометрических построений и моделирования таких процессов, развитие изобразительных умений с помощью средств ИКТ;

– формирование информационной и алгоритмической культуры, развитие основных навыков использования компьютерных устройств и программ;

– формирование умения соблюдать нормы информационной этики и права.

Календарно-тематическое планирование: 1 год обучения

№	***Тема***	***Теория***	***Практика***	***Всего***
1.	Правила ТБ и поведения в компьютерном классе	1		1
2.	Знакомство со средой Scratch. Внешний вид среды, поля. Анимация		1	1
3.	Исполнитель Scratch, цвет и размер пера		1	1
4.	Основные инструменты встроенного растрового графического редактора	1		1
5.	Линейный алгоритм. Создание блок-схемы. Основные графические примитивы векторного редактора LibreOffice Draw		1	1
6.	Линейный алгоритм. Рисование линий исполнителем Scratch		1	1
7.	Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно	1		1
8.	Конечный цикл. Исполнитель Scratch рисует квадраты, линии. Блок-схема цикла		1	1
9.	Конечный цикл. Исполнитель		1	1

	Scratch рисует несколько линий и фигур. Копирование фрагментов программы			
10.	Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы	1		1
11.	Цикл в цикле. Повторение пунктирной линии с поворотом		1	1
12.	Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов		1	1
13.	Сцена как исполнитель. Создаем модель таймера	1		1
14.	Бесконечный цикл. Одна программа для исполнителя Scratch, но разные костюмы		1	1
15.	Одинаковые программы для нескольких исполнителей		1	1
16.	Несколько исполнителей. Параллельное выполнение действий исполнителями	1		1
17.	Разбиение программы на части для параллельного выполнения исполнителями. Таймер. Звук		1	1
18.	Два исполнителя со своими программами. Мини-проект «Часы»		1	1
19.	Алгоритмы с ветвлением. Условие ЕСЛИ. Два исполнителя. Блок-схема алгоритма с условием	1		1
20.	Цикл при условии. Мини-проект «Шарики в лабиринте»		1	1
21.	Цикл при условии. Исполнитель определяет цвета		1	1
22.	Цикл при условии. Исполнители в разных слоях. Мини-проект «Самолет сквозь облака»	1		1
23.	Перемещение исполнителя из одного слоя в другой. Действия исполнителей в разных слоях. Мини-проект «Дорога»		1	1
24.	Алгоритмы с ветвлением. Условие ЕСЛИ. Взаимодействие исполнителей		1	1
25.	Сцена как исполнитель.	1		1

	Последовательное выполнение команд исполнителями			
26.	Алгоритмы с ветвлением. Программирование клавиш		1	1
27.	Алгоритмы с ветвлением. ЕСЛИ касается цвета		1	1
28.	Интерактивность исполнителей. Создание мини-проекта «Лабиринт»	1		1
29.	Игра «Лабиринт». Усложнение		1	1
30.	Моделирование ситуации. Мини-проект «Пешеходный переход»		1	1
31.	Моделирование ситуации. Интерактивность исполнителей. Мини-проект «Водолей»	1		1
32.	Моделирование. Учебные модели «Рисующий карандаш», «Затухание»		1	1
33.	Моделирование. Модель теста «Комнатные растения»		1	1
34.	Моделирование. Обучающий проект по маршрутам географических открытий	1		1
35.	Обобщение пройденного за курс		1	1
36.	Обзор разработанных моделей		1	1
37.	Модель «Координатная плоскость». Геометрические фигуры	1		1
38.	Модель «Координатная плоскость». Повторение фигур в цикле		1	1
39.	Создание модели «Координатная плоскость»		1	1
40.	Списки. Создание викторины	1		1
41.	Переменные. Мини-проект «Периметр прямоугольника»		1	1
42.	Переменные. Мини-проект «Площадь прямоугольника»		1	1
43.	Использование подпрограмм	1		1
44.	Мини-проект «Многоугольники»		1	1
45.	Мини-проект «Палиндром»		1	1
46.	Обобщение по теме «Переменные и списки»	1		1
47.	Среда для управления черепашкой. Основные команды. Формы		1	1
48.	Решение задач с использованием		1	1

	линейных алгоритмов			
49.	Цвет и размер формы. Решение задач	1		1
50.	Возможности рисования в PyTurtle. Цвет и размер пера		1	1
51.	Циклические конструкции в PyTurtle		1	1
52.	Создание геометрических фигур в PyTurtle	1		1
53.	Рисование спиралей в PyTurtle		1	1
54.	Конструкции ветвления алгоритмов в PyTurtle		1	1
55.	Решение задач	1		1
56.	Обобщение по теме «Черепашья графика»		1	1
57.	Язык программирования Python. Структура программы в языке Python. Создание и запуск простых программ		1	1
58.	Арифметические операции. Числовые и строковые типы данных. Преобразование типов данных	1		1
59.	Понятие переменной. Оператор присваивания. Таблица значений переменной		1	1
60.	Вывод данных. Функция print		1	1
61.	Ввод данных с клавиатуры. Функция input	1		1
62.	Целочисленная арифметика. Решение задач		1	1
63.	Программирование линейных алгоритмов. Решение задач		1	1
64.	Программирование разветвляющихся алгоритмов. Решение задач	1		1
65.	Условный оператор. Решение задач		1	1
66.	Программирование циклов с заданным числом повторений. Цикл for		1	1
67.	Программирование циклов с заданных числом повторений. Решение задач	1		1
68.	Обобщение и систематизация		1	1

	основных понятий темы «Начала программирования»			
69.	Повторение пройденного		1	1
70.	Свободное программное обеспечение.	1		1
71.	Знакомство с текстовыми и графическими редакторами		1	1
72.	Сканирование изображений		1	1
73.	Обработка изображений	1		1
74.	Импорт и экспорт объектов		1	1
75.	Notepad		1	1
76.	Текстовый процессор Word	1		1
77.	Форматирование текста		1	1
78.	Текст и графика в таблицах		1	1
79.	Создание объектов WordArt	1		1
80.	Диаграммы в Word		1	1
81.	Графические возможности текстового процессора Word		1	1
82.	MS Paint	1		1
83.	Импортирование изображений из приложений Windows в MS Paint		1	1
84.	Экспорт изображений из MS Paint		1	1
85.	Понятие информационной модели	1		1
86.	Представление данных в виде таблиц		1	1
87.	Представление данных в виде схем		1	1
88.	Табличные процессоры	1		1
89.	Представление данных в виде диаграмм		1	1
90.	Выполнение вычислений средствами Excel		1	1
91.	Импорт (экспорт) данных в (из) Excel	1		1
92.	Решение задачи анализа расходов средствами Excel		1	1
93.	Построение диаграмм		1	1
94.	Создание презентационных материалов	1		1
95.	Назначение и описание PowerPoint		1	1
96.	Создание презентаций		1	1
97.	Разработка структуры презентации	1		1
98.	Размещение текста в презентациях		1	1
99.	Сопровождение публичных выступлений средствами			

	компьютера			
100	Анализ функциональности PowerPoint	1		1
101	Работа с графическими объектами в PowerPoint		1	1
102	Работа с таблицами в PowerPoint		1	1
103	Создание диаграмм в PowerPoint	1		1
104	Создание презентации, посвященной П.М. Миронову		1	1
105	Создание презентации, И.Я. Яковлеву		1	1
106	Обзор звуковых и визуальных эффектов	1		1
107	Создание презентации ко Дню России. Подбор материала		1	1
108	Создание презентации ко Дню России. Выбор дизайна		1	1
Итого		36	72	108

Календарно-тематическое планирование: 2 год обучения

№	*Тема*	*Теория*	*Практика*	*Всего*
1.	Правила ТБ и поведения в компьютерном классе. История вычислительной техники	1		1
2.	Представление информации в ЭВМ		1	1
3.	Файлы и папки		1	1
4.	Создание, копирование и перемещение	1		1
5.	Поиск файлов и папок		1	1
6.	Двоичное кодирование цифровой информации		1	1
7.	Перевод целых десятичных чисел в двоичный код	1		1
8.	Клавиатурный тренажер		1	1
9.	Программа «Блокнот»		1	1
10.	Программа «Калькулятор»	1		1
11.	Получение десятичного числа по двоичному коду		1	1
12.	Тексты в памяти компьютера		1	1
13.	Программа «Проводник»	1		1
14.	Текстовый процессор Word		1	1

15.	Редактирование и формат текстов		1	1
16.	Изображения в памяти компьютера	1		1
17.	История счета и систем счисления		1	1
18.	Создание надписей		1	1
19.	Нумерованные списки	1		1
20.	Маркированные списки		1	1
21.	Единицы измерения информации		1	1
22.	Информация и знания	1		1
23.	Создание таблиц		1	1
24.	Текст и графика в таблицах		1	1
25.	Диаграммы в Word	1		1
26.	Графические редакторы		1	1
27.	Чувственное познание окружающего мира		1	1
28.	MS Paint	1		1
29.	Графические возможности текстового процессора Word		1	1
30.	Рисунки в Word		1	1
31.	Формы представления графической информации	1		1
32.	Обзор графических редакторов		1	1
33.	Создание растровых изображений		1	1
34.	Импортирование изображений из приложений Windows в MS Paint	1		1
35.	Экспортирование изображений в документы Word		1	1
36.	Конвертирование файлов из одного формата в другой		1	1
37.	Онлайн-конвертеры	1		1
38.	Введение в язык программирования Python		1	1
39.	Установка Python и сред разработки		1	1
40.	Целочисленная арифметика	1		1
41.	Операторы //, %		1	1
42.	Условная инструкция в языке Python		1	1
43.	Оператор if	1		1
44.	Цикл for в языке Python		1	1
45.	Оператор for		1	1
46.	Оператор for	1		1
47.	Цикл while в языке Python		1	1
48.	Оператор while		1	1
49.	Инструкции управления циклом	1		1
50.	Программы с циклом while		1	1

51.	Строки в языке Python		1	1
52.	Строки, срезы строк	1		1
53.	Методы строк		1	1
54.	Коды символов		1	1
55.	Программы на обработку строк	1		1
56.	Списки в языке Python		1	1
57.	Списки, срезы списков		1	1
58.	Методы split и join для списка строк	1		1
59.	Генераторы списков		1	1
60.	Работа со списками		1	1
61.	Словари и множества. Файловый ввод- вывод	1		1
62.	Множества в Python		1	1
63.	Словари (ассоциативные массивы)		1	1
64.	Работа с текстовыми файлами	1		1
65.	Многомерные списки		1	1
66.	Генераторы таблиц		1	1
67.	Функции. Рекурсия	1		1
68.	Локальные переменные		1	1
69.	Глобальные переменные		1	1
70.	Функции	1		1
71.	Рекурсия		1	1
72.	Рекурсия. Ханойские башни		1	1
73.	Ханойские башни	1		1
74.	Ханойские башни		1	1
75.	Поверка числа на простоту		1	1
76.	Алгоритм Евклида	1		1
77.	Наибольший общий делитель		1	1
78.	Наименьшее общее кратное		1	1
79.	Расширенный алгоритм Евклида	1		1
80.	Алгоритм Евклида		1	1
81.	Алгоритм Евклида. Модификации		1	1
82.	Алгоритм Евклида. Модификации	1		1
83.	Разложение на множители. Решето Эратосфена		1	1
84.	Разложение на множители		1	1
85.	Решето Эратосфена	1		1
86.	Решето Эратосфена. Модификация		1	1
87.	Квадратичные алгоритмы сортировки		1	1
88.	Сортировка «пузырьком»	1		1
89.	Сортировка прямыми вставками		1	1
90.	Сортировка. Модификация		1	1

	алгоритмов			
91.	Сортировка подсчетом	1		1
92.	Большая сортировка		1	1
93.	Сортировка. Модификация алгоритмов		1	1
94.	Сортировка. Эффективные алгоритмы	1		1
95.	Линейные структуры данных		1	1
96.	Стек		1	1
97.	Правильные скобочные последовательности	1		1
98.	Обратная польская нотация		1	1
99.	Очередь, дек		1	1
100	Очередь. Решение задач	1		1
101	Дек. Решение задач		1	1
102	Сортировка слиянием		1	1
103	Быстрая сортировка Хоара	1		1
104	Стандартная сортировка		1	1
105	Решение задач		1	1
106	Два указателя	1		1
107	Метод двух указателей		1	1
108	Решение задач		1	1
		36	72	108

Календарно-тематическое планирование: 3 год обучения

№	*Тема*	*Теория*	*Практика*	*Всего*
1.	Правила ТБ и поведения в компьютерном классе. Системное и прикладное программное обеспечение	1		1
2.	Обзор системного программного обеспечения		1	1
3.	Служебные программы		1	1
4.	Работа с программой Scandisk	1		1
5.	Программа очистки диска		1	1
6.	Обзор прикладного программного обеспечения		1	1
7.	Назначение и описание PowerPoint	1		1
8.	Создание презентаций		1	1
9.	Разработка структуры презентации		1	1
10.	Размещение текста в презентациях	1		1
11.	Сопровождение публичных		1	1

	выступлений средствами компьютера			
12.	Анализ функциональности PowerPoint		1	1
13.	Работа с графическими объектами в PowerPoint	1		1
14.	Работа с таблицами в PowerPoint		1	1
15.	Создание диаграмм в PowerPoint		1	1
16.	Обзор издательских систем	1		1
17.	Возможности MS Word для создания печатных материалов		1	1
18.	Создание презентации на зимнюю тему		1	1
19.	Создание презентации ко Дню Защитника Отечества. Разработка структуры	1		1
20.	Создание презентации ко Дню Защитника Отечества. Выбор дизайна		1	1
21.	Технология создания буклетов		1	1
22.	Технология создания объявлений	1		1
23.	Создание презентации к Международному женскому дню. Разработка структуры		1	1
24.	Создание презентации к Международному женскому дню. Выбор дизайна		1	1
25.	Создание презентации к Международному женскому дню. Выбор звуковых и анимационных эффектов	1		1
26.	Элементы алгоритмизации. Алгоритмы в повседневной жизни		1	1
27.	Способы задания алгоритмов		1	1
28.	Знакомство со средой Лого	1		1
29.	Команды для графического исполнителя		1	1
30.	Вычисления на компьютере		1	1
31.	Математические пакеты	1		1
32.	Построение графиков		1	1
33.	Решение уравнений		1	1
34.	Решение систем уравнений	1		1
35.	Вычисление суммы членов арифметической прогрессии		1	1

36.	Вычисление суммы членов геометрической прогрессии		1	1
37.	Средства разработки программ на C++	1		1
38.	Запуск среды MinGW. Первая программа. Алгоритм, программа, блок-схема алгоритма.		1	1
39.	Операторы ввода-вывода. Квадратное уравнение. Теорема Виета. Программа решения квадратного уравнения.		1	1
40.	Переменные и типы	1		1
41.	Понятие переменной. Целые и вещественные типы данных.		1	1
42.	Структура программы.		1	1
43.	Стандартные библиотеки.	1		1
44.	Программа «Генератор сказок».		1	1
45.	Стандартные функции. Примеры арифметических выражений.		1	1
46.	Теорема Пифагора. Вычисление параметров прямоугольного треугольника.	1		1
47.	Длина окружности и площадь круга.		1	1
48.	Отладка программ в MinGW.		1	1
49.	Целочисленное деление и остаток от деления. Извлечение цифр из целого числа.	1		1
50.	Алгоритм ветвления. Условный оператор «if». Логические выражения.		1	1
51.	Условный оператор «?».		1	1
52.	Программы «Знак числа» и «Сортировка трех чисел».	1		1
53.	Задача «Шахматы». Введение в олимпиадное программирование.		1	1
54.	Сокращенные операции присваивания в C++.		1	1
55.	Циклы. Циклический алгоритм. Типы циклов. Оператор цикла с параметром. Операторы циклов с предусловием и постусловием.	1		1
56.	Поиск счастливых билетов. Чтение данных до конца потока.		1	1

	Программа «Ввод пароля». НОД и НОК. Реализация алгоритма Евклида. Программа «Сокращение дроби». Генератор случайных чисел.			
57.	Программы «Угадай число» и «Угадай число-2». Программы «Таблица умножения» и «Шахматная доска».		1	1
58.	Форматированный ввод-вывод. Использование манипуляторов форматирования. Функции printf и scanf.	1		1
59.	Программа «Таблица умножения в 16-й системе счисления». Программа «Калькулятор». Работа с файлами		1	1
60.	Символьные типы данных		1	1
61.	Тип данных char. Таблица ASCII-кодов. Тип данных string.	1		1
62.	Функции библиотеки string.		1	1
63.	Проверка строки на палиндром.		1	1
64.	Подсчет количества слов в предложении.	1		1
65.	Бинарный поиск		1	1
66.	Понятие бинарного поиска.		1	1
67.	Алгоритмы вычисления квадратного корня.	1		1
68.	Решение задач		1	1
69.	Решение задач		1	1
70.	Системы счисления	1		1
71.	Позиционная система счисления.		1	1
72.	Формы записи числа.		1	1
73.	Примеры и алгоритмы перевода чисел.	1		1
74.	Перевод из десятичной системы счисления в двоичную и обратно.		1	1
75.	Массивы. Понятие массива.		1	1
76.	Программы «Магазин №1» и «Магазин №2».	1		1
77.	Заполнение и вывод массива в обратном порядке.		1	1
78.	Поиск максимального и минимального элемента в массиве.		1	1

79.	Среднее арифметическое, квадратическое и геометрическое.	1		1
80.	Структуры. Понятие структуры.		1	1
81.	Битовые поля		1	1
82.	Объединения и перечисления.	1		1
83.	Программы «Список учащихся»		1	1
84.	Программа «Точки на плоскости».		1	1
85.	Функции. Понятие функции. Примеры описания функций.	1		1
86.	Программы «Автомобильный номер» и «Системы счисления».		1	1
87.	Задачи «Простые операции над массивом» и «Треугольник».		1	1
88.	Механизм параметров. Библиотека ctime. Клавиатурный тренажер.	1		1
89.	Реализация функций Delay, Randomize и Random.		1	1
90.	Сортировка		1	1
91.	Сортировка выбором	1		1
92.	Сортировка пузырьком		1	1
93.	Сортировка подсчетом		1	1
94.	Сортировка в С++	1		1
95.	Реализация и анализ алгоритмов сортировки		1	1
96.	Сортировка структур в С++		1	1
97.	Двумерные массивы	1		1
98.	Понятие двумерного массива		1	1
99.	Поиск столбца с минимальной суммой элементов		1	1
100	Поиск строки с максимальной суммой элементов	1		1
101	Транспонирование матрицы		1	1
102	Сумма и произведение матриц		1	1
103	Задачи «Миша и негатив» и «Спираль»	1		1
104	Рекурсия. Понятие рекурсии. Рекурсивные алгоритмы сортировки		1	1
105	Задача о Ханойской башне		1	1
106	Перебор перестановок	1		1
107	Задача о восьми ферзях		1	1
108	Задача о рюкзаке		1	1
Итого		36	72	108

Содержание программы

Тема 1. Знакомство с программной средой Scratch

Свободное программное обеспечение. Авторы программной среды Scratch. Параметры для скачивания и установки программной среды на домашний компьютер. Основные элементы пользовательского интерфейса программной среды Scratch. Внешний вид рабочего окна. Блочная структура систематизации информации. Функциональные блоки. Блоки команд, состояний, программ, запуска, действий и исполнителей. Установка русского языка для Scratch.

Создание и сохранение документа. Понятия «спрайт», «сцена», «скрипт». Очистка экрана. Основной персонаж как исполнитель программ. Система команд исполнителя (СКИ). Блочная структура программы. Непосредственное управление исполнителем.

Библиотеки исполнителей. Сцена и разнообразие сцен, исходя из библиотеки сцен. Систематизация данных библиотек исполнителей. Иерархия в организации хранения костюмов персонажа и фонов для сцен. Импорт костюма, импорт фона.

Аналитическая деятельность:

- определять технические устройства для ввода и вывода информации;
- понимать иерархическую организацию библиотек исполнителей программной среды;
- выделять путь к элементам библиотеки;

Практическая деятельность:

- выбирать и запускать программную среду Scratch;
- работать с основными элементами пользовательского интерфейса программной среды;
- изменять размер и перемещать окно программы, выбирать необходимый режим окна;
- вводить имя файла с помощью клавиатуры;
- выбирать необходимый файл из нужной папки;
- создавать, копировать, переименовывать, перемещать, копировать и удалять файлы;
- соблюдать требования техники безопасности при работе в компьютерном классе.

Тема 2. Компьютерная графика

Компьютерная графика. Векторные и растровые графические редакторы. Встроенный растровый графический редактор. Основные инструменты графического редактора – кисточка, ластик, заливка (цветом или градиентом), рисование линий, прямоугольников, квадратов, эллипсов и окружностей, выбор фрагмента изображения и отражение его по горизонтали или вертикали, использование инструмента «Печать» для копирования выделенной области изображения, работа с текстом. Масштаб фрагмента изображения. Палитра цветов, установка цвета переднего плана и фона, выбор

цвета изображения с помощью инструмента «Пипетка». Изменение центра костюма. Изменение размера костюма.

Основные способы изменения внешнего вида исполнителя:

1) использование встроенной библиотеки исполнителей путём импорта её элемента;

2) редактирование выбранного элемента с помощью инструментов встроенного растрового графического редактора;

3) создание собственных изображений в других программах (например, LibreOffice Draw) и импортирование их в программную среду Scratch.

Знакомство с основными графическими примитивами векторного редактора LibreOffice Draw. Создание геометрических фигур без внутренней заливки, но с текстовым блоком внутри. Стрелки, их направление.

Аналитическая деятельность:

- выделять фрагменты изображения для дальнейшей работы с ними;
- планировать работу по созданию сложных изображений путём копирования и масштабирования простых;
- различать верхний и нижний цвета изображения;
- придумывать и создавать различные градиенты для заливки замкнутой области;
- планировать создание симметричных объектов с использованием возможности отражения фрагмента изображения.

Практическая деятельность:

- использовать простейшие растровые и векторные редакторы для создания и редактирования изображений;
- изменять центр изображения костюма исполнителя;
- редактировать изображения встроенной библиотеки исполнителей;
- создавать сложные графические объекты путем копирования и модификации простых объектов и их фрагментов;
- использовать возможности работы с цветом.

Тема 3. Алгоритмы и исполнители

Алгоритм. Понятие алгоритма как формального описания последовательности действий исполнителя, приводящих от исходных данных к конечному результату. Схематическая запись алгоритма. Использование геометрических фигур для схематической записи алгоритма. Создание блок-схем в свободном векторном редакторе LibreOffice Draw.

Линейные алгоритмы

Основные признаки линейного алгоритма. Схематическое описание линейного алгоритма. Геометрические примитивы, используемые для описания линейного алгоритма.

Программное управление исполнителем. Создание программ для перемещения исполнителя по экранному полю. Понятие «поворот исполнителя в определённое направление». Угол поворота 90°. Поворот исполнителя на 90° по часовой или против часовой стрелки.

Создание программ для рисования линий. Изменение цвета и толщины рисуемой линии. Особенности пунктирной линии. Программа для исполнителя, чтобы он оставлял пунктирную линию при перемещении по экранному полю.

Прямоугольник, квадрат – основные черты. Написание программ для движения исполнителя вдоль сторон квадрата, прямоугольника. Редактирование программы рисования квадрата для получения квадрата другой с другим размером стороны.

Прерывание программы.

Циклические алгоритмы

Организация циклического алгоритма при необходимости многократного повторения команд. Особенности использования цикла в программе. Оптимизация программы путём сокращения количества команд при переходе от линейных алгоритмов к циклическим.

Схематическая запись циклического алгоритма.

Типы циклических алгоритмов. Основные конструкции программной среды, используемые для написания программ исполнителям с применением циклов.

Конечный цикл. Оптимизация программы для исполнителя, рисующего линии, квадраты, прямоугольники при использовании цикла. Программа исполнителя для рисования нескольких однотипных геометрических фигур, например, нескольких квадратов из одной вершины, но с различным значением стороны.

Конструкции программной среды «спрятаться/показаться». Выполнение программы исполнителем, не показанным в поле выполнения программы.

Написание и отладка программ с применением конструкции «цикл в цикле».

Бесконечный цикл. Повторяющаяся смена внешности исполнителя для имитации движения персонажа. Использование бесконечного цикла для создания анимации.

Получение различных эффектов воспроизведения программы за счёт изменения костюмов исполнителей Scratch.

Параллелизм в программной среде

Использование нескольких исполнителей. Копирование программ исполнителей. Выполнение одинаковых программ разными исполнителями при использовании различных начальных условий. Параллельное выполнение одинотипных действий. Принцип суперкомпьютерных технологий. Таймер для вычисления времени выполнения программы.

Интерактивность программ. Возможность организации диалога между исполнителями. Операторы для слияния текстовых выражений.

Взаимодействие исполнителей. Сенсоры касания объектов или цвета. Использование сенсоров при взаимодействии исполнителей. Задержка выполнения программы.

Слои изображения. Выполнение команд исполнителями в разных слоях изображения.

Ветвление в алгоритмах

Использование ветвления при написании программ. Короткая форма. Полная форма условного оператора. Конструкции ветвления для моделирования ситуации.

Цикл «пока». Повторение команд исполнителя при выполнении определенного условия.

Последовательное выполнение фрагментов программы разными исполнителями

Типы исполнителей программной среды Scratch. Системы команд исполнителей. Различные системы команд для разных типов исполнителей.

Управление событиями. Передача сообщений исполнителям для выполнения определенной последовательности команд.

Передача управления между различными типами исполнителей.

Аналитическая деятельность:

- придумывать задачи для исполнителей программной среды;
- выделять ситуации, для описания которых можно использовать линейный алгоритм, алгоритм с ветвлениями, повторениями;
- определять эффективный способ решения поставленной задачи;
- находить параллельности в выполняемых действиях и программировать их с помощью нескольких исполнителей;
- планировать последовательность событий для заданного проекта.

Практическая деятельность:

- составлять и отлаживать программный код;
- использовать конструкции программной среды для создания линейных, разветвлённых и циклических алгоритмов;
- организовывать параллельные вычисления;
- моделировать последовательность событий программы, передачу управления от одних исполнителей другим.

Тема 4. Проектная деятельность и моделирование процессов и систем

Мультимедийный проект. Описание сюжетных событий. Анимация. Создание эффекта анимации с помощью последовательной смены изображений. Имитационные модели. Интерактивные проекты. Игры.

Аналитическая деятельность:

- создавать план появления событий для отражения определенной темы;
- выбирать иллюстративный материал из встроенной библиотеки;
- выбирать метод анимации для конкретной задачи;
- планировать последовательность событий для создания эффекта анимации по выбранному сценарию.

Практическая деятельность:

- использовать возможности программной среды Scratch для создания мультимедийных проектов;

- создавать имитационные модели, интерактивные проекты и игры средствами программной среды.

Курс является практикоориентированным, поэтому каждое занятие содержит как теоретический, так и практический материал.

Тема 5. Переменные и списки в программной среде Scratch.

Создание геометрических фигур на координатной плоскости. Командный блок Переменные. Имя переменной. Значение переменной. Создание и использование переменных в программной среде Scratch. Использование переменных для создания модели «Координатная плоскость». Списки в программной среде Scratch. Создание и удаление списка. Длина списка. Элемент списка. Действия над элементами списка. Использование списков для решения математических задач. Создание списков для моделирования диалога исполнителей. Использование переменных для создания интерактивной модели вычисления периметра прямоугольника по заданным сторонам. Использование переменных для создания интерактивной модели вычисления площади прямоугольника по заданным сторонам.

Аналитическая деятельность:

- определять необходимость использования переменных и списков при создании творческих проектов;
- понимать назначение инициализации переменных;
- выделять номера и значения элементов списка;

Практическая деятельность:

– создавать списки и переменные в программной среде Scratch;
– присваивать имя переменной в соответствии с назначением;
– работать с основными командами блока Переменные;
– добавлять элементы в созданный список;
– удалять из списка элемент по его номеру;
– заменять элемент списка по номеру;
– использовать значение длины списка;
– использовать условие — содержит ли список указанное значение.

Тема 6. Черепашья графика в Pyturtle.

Интегрированная среда разработки. Импорт библиотеки Pyturtle. Графический исполнитель библиотеки Pyturtle. Формы. Цвет и размер формы графического исполнителя. Сохранение и запуск программ. Создание простых программ на основе линейных алгоритмов. Команды перемещения и поворота в Pyturtle. Создание линейных программ для рисования геометрических фигур исполнителем. Цвет и размер пера. Заполнение цветом. Циклическая конструкция. Комментирование в программе. Перемещение в точку с указанными координатами. Решение задач с использованием циклической конструкции. Рисование исполнителем симметричных геометрических фигур. Использование конструкций цикл в цикле. Рисование спиралей в Pyturtle. Условная конструкция. Определение и использование шестнадцатеричного значения цвета.

Аналитическая деятельность:

– планировать действия исполнителя для изображения геометрических фигур;

– выделять необходимые для создания проектов алгоритмические конструкции;

– придумывать и создавать мини-проекты с использованием библиотеки Pyturtle;

– планировать и оптимизировать деятельность исполнителей для достижения поставленной цели.

Практическая деятельность:

– использовать библиотеку Pyturtle;

– создавать, сохранять и запускать программы в Pyturtle;

– устанавливать параметры формы графического исполнителя;

– использовать основные команды управления черепашкой;

– изменять цвет и размер пера при рисовании геометрических фигур;

– использовать возможности заполнения цветом;

– создавать и удалять комментарии к командам программы;

– использовать линейные, циклические и условные конструкции для решения задач

Тема 7. Начала программирования.

Языки программирования. Среда разработки. Интерактивный режим. Типы данных. Числовые и строковые типы данных. Функции преобразования типов данных. Структура программы в языке Python. Создание и запуск простых программ. Арифметические операции. Основные операции для чисел. Операции над строками. Функция вывода данных. Параметры функции вывода данных. Возможности комбинированного вывода данных. Параметр sep. Построчный вывод данных. Понятие переменной. Тип переменной. Операция присваивания. Таблица значений переменной. Ввод данных с клавиатуры. Функция ввода данных. Ввод данных целочисленного типа. Целочисленная арифметика. Целая часть частного. Остаток от деления. Ветвление, синтаксис оператора. Операции сравнения. Полная форма условной конструкции. Сокращенная форма условной конструкции. Многообразие способов записи ветвления. Программирование циклов с заданных числом повторений. Цикл for. Функция range.

Аналитическая деятельность:

– анализировать условие поставленной задачи для составления алгоритма её решения;

– выделять в условии задачи необходимые для решения данные;

– определять необходимые типы данных для дальнейшей обработки;

– анализировать сообщения, полученные при выполнении программы;

– планировать использование различных алгоритмических конструкций для решения задачи;

– критически оценивать полученную программу и выделять пути её оптимизации. Практическая деятельность:

– использовать команды языка программирования для работы в интерактивном режиме;

– составлять простые программы, используя интегрированную среду разработки;

– выявлять синтаксические ошибки на основе полученных сообщений интерпретатора;

– производить отладку программного кода;

– использовать конструкции программной среды для программирования линейных, разветвлённых и циклических алгоритмов;

– организовывать ввод данных различных типов в соответствии с поставленной задачей;

– использовать необходимые средства языка программирования для осуществления вывода данных в соответствии с заданием.

Курс является практикоориентированным, поэтому каждое занятие содержит как теоретический, так и практический материал.

Тема 8. Использование периферийных устройств и прикладного программного обеспечения в проектной деятельности

Свободное программное обеспечение. Знакомство с текстовыми и графическими редакторами. Сканирование изображений. Анализ интерфейса программ из пакета MS Office. Обработка изображений. Импорт и экспорт объектов. Текстовый процессор Word. Текст и графика в таблицах. Диаграммы в Word. MS Paint. Конвертирование файлов из одного формата в другой. Графические возможности текстового процессора Word. Перенос данных из Word в Excel. Перенос данных из Excel в Word. Импортирование изображений из приложений Windows в MS Paint.

Понятие информационной модели. Виды информационных моделей. Представление данных в виде таблиц. Представление данных в виде схем. Представление данных в виде диаграмм. Табличные процессоры. Сравнительный анализ табличных и текстовых процессоров. Выполнение вычислений средствами Excel. Решение задачи анализа расходов средствами Excel.

Назначение и описание PowerPoint. Анализ функциональности PowerPoint. Создание презентационных материалов. Разработка структуры презентации. Размещение текста в презентациях. Сопровождение публичных выступлений средствами компьютера. Работа с графическими объектами в PowerPoint. Работа с таблицами в PowerPoint. Создание диаграмм в PowerPoint.

Системное и прикладное программное обеспечение. Обзор системного программного обеспечения. Служебные программы. Работа с программой Scandisk. Программа очистки диска. Обзор прикладного программного обеспечения.

Обзор издательских систем. Возможности MS Word для создания печатных материалов.

Аналитическая деятельность:

- определять технические устройства для ввода и вывода информации;

- анализ проблемной ситуации с целью выбора способов сбора и обработки информации.

Практическая деятельность:

- выбирать программное обеспечение для каждой определенной проблемной ситуации;
- сканирование и обработка изображений;
- печать текстов и изображений;
- работать с основными формами информации;
- создавать презентации на выбранные темы;
- импортировать объекты из файлов в текстовый редактор;
- создавать, копировать, переименовывать, перемещать, копировать и удалять фрагменты текста, изображения.

Тема 9. Программирование в Python

9.1. Введение в язык программирования Python

Установка Python и сред разработки, синтаксис языка, переменные, типы данных и приоритет операций, преобразование типов, ввод-вывод данных. Операторы //, %.

9.2. Условная инструкция в языке Python

Оператор if. Составление алгоритмов с ветвлением. Решение задач.

9.3. Циклы в языке Python

Оператор for. Оператор while, инструкции управления циклом.

9.4. Структуры данных в Python

Строки, срезы строк, методы строк, коды символов. Списки, срезы списков, методы split и join для списка строк, генераторы списков. Множества в Python, словари (ассоциативные массивы), работа с текстовыми файлами, многомерные списки, генераторы таблиц. Линейные структуры данных. Стек, правильные скобочные последовательности, обратная польская нотация. Очередь, дек.

9.5. Подпрограммы в Python

Функции, локальные и глобальные переменные, рекурсия.

9.6. Стандартные алгоритмы

Проверка чисел на простоту. Алгоритм Евклида. Расширенный алгоритм Евклида. Разложение на множители. Решето Эратосфена. Квадратичные алгоритмы сортировки. Сортировка подсчетом. Сортировка слиянием, быстрая сортировка Хоара, стандартная сортировка. Метод двух указателей

Аналитическая деятельность:

– анализировать условие поставленной задачи для составления алгоритма её решения;

– выделять в условии задачи необходимые для решения данные;

– определять необходимые типы данных для дальнейшей обработки;

– анализировать сообщения, полученные при выполнении программы;

– планировать использование различных алгоритмических конструкций для решения задачи;

– критически оценивать полученную программу и выделять пути её оптимизации.

Практическая деятельность:

– использовать команды языка программирования для работы в интерактивном режиме;

– выявлять синтаксические ошибки на основе полученных сообщений интерпретатора;

– производить отладку программного кода;

– использовать конструкции программной среды для программирования линейных, разветвлённых и циклических алгоритмов;

– организовывать ввод данных различных типов в соответствии с поставленной задачей;

– использовать необходимые средства языка программирования для осуществления вывода данных в соответствии с заданием.

Курс является практикоориентированным, поэтому каждое занятие содержит как теоретический, так и практический материал.

Тема 10. Язык программирование C++

10.1. Введение в программирование

Средства разработки программ на C++. Запуск среды MinGW. Первая программа. Алгоритм, программа, блок-схема алгоритма. Операторы ввода-вывода. Квадратное уравнение. Теорема Виета. Программа решения квадратного уравнения.

10.2. Переменные и типы

Понятие переменной. Целые и вещественные типы данных. Структура программы. Стандартные библиотеки. Программа «Генератор сказок».

10.3. Стандартные функции

Стандартные функции. Примеры арифметических выражений. Теорема Пифагора. Вычисление параметров прямоугольного треугольника. Длина окружности и площадь круга. Отладка программ в MinGW. Целочисленное деление и остаток от деления. Извлечение цифр из целого числа.

10.4. Условный оператор

Алгоритм ветвления. Условный оператор «if». Логические выражения. Условный оператор «?». Программы «Знак числа» и «Сортировка трех чисел». Задача «Шахматы». Введение в олимпиадное программирование. Сокращенные операции присваивания в C++.

10.5. Циклы

Циклический алгоритм. Типы циклов. Оператор цикла с параметром. Программа, возводящая в квадрат числа от 1 до 10. Программа, переводящая дюймы в сантиметры. Программа, вычисляющая AB без использования функции pow. Поиск счастливых билетов. Операторы циклов с предусловием и постусловием. Чтение данных до конца потока. Программа «Ввод пароля». НОД и НОК. Реализация алгоритма Евклида. Программа «Сокращение дроби». Генератор случайных чисел. Программы «Угадай число» и «Угадай число-2». Программы «Таблица умножения» и «Шахматная доска». Форматированный

ввод-вывод. Использование манипуляторов форматирования. Функции printf и scanf. Программа «Таблица умножения в 16-й системе счисления». Программа «Калькулятор». Работа с файлами

10.6. Символьные типы данных

Тип данных char. Таблица ASCII-кодов. Тип данных string. Функции библиотеки string. Проверка строки на палиндром. Подсчет количества слов в предложении.

10.7. Бинарный поиск

Понятие бинарного поиска. Алгоритмы вычисления квадратного корня.

10.8. Системы счисления

Позиционная система счисления. Формы записи числа. Примеры и алгоритмы перевода чисел. Перевод из десятичной системы счисления в двоичную и обратно.

10.9. Массивы

Понятие массива. Программы «Магазин №1» и «Магазин №2». Заполнение и вывод массива в обратном порядке. Поиск максимального и минимального элемента в массиве. Среднее арифметическое, квадратическое и геометрическое.

10.10. Структуры

Понятие структуры. Битовые поля, объединения и перечисления. Программы «Список учащихся» и «Точки на плоскости».

10.11. Функции

Понятие функции. Примеры описания функций. Программы «Автомобильный номер» и «Системы счисления». Задачи «Простые операции над массивом» и «Треугольник». Механизм параметров. Библиотека ctime. Клавиатурный тренажер. Реализация функций Delay, Randomize и Random.

10.12. Сортировка

Сортировка выбором. Сортировка пузырьком. Сортировка подсчетом. Сортировка в C++. Реализация и анализ алгоритмов сортировки. Сортировка структур в C++.

10.13. Двумерные массивы

Понятие двумерного массива. Поиск строки с максимальной суммой элементов. Транспонирование матрицы. Сумма и произведение матриц. Задачи «Миша и негатив» и «Спираль».

10.14. Рекурсия

Понятие рекурсии. Рекурсивные алгоритмы сортировки. Задача о Ханойской башне. Перебор перестановок. Задача о восьми ферзях. Задача о рюкзаке.

Аналитическая деятельность:

– анализировать условие поставленной задачи для составления алгоритма её решения;

– выделять в условии задачи необходимые для решения данные;

– определять необходимые типы данных для дальнейшей обработки;

– анализировать сообщения, полученные при выполнении программы;

– планировать использование различных алгоритмических конструкций для решения задачи;

– критически оценивать полученную программу и выделять пути её оптимизации. Практическая деятельность:

– использовать команды языка программирования для работы в интерактивном режиме;

– составлять простые программы, используя интегрированную среду разработки;

– выявлять синтаксические ошибки на основе полученных сообщений интерпретатора;

– производить отладку программного кода;

– использовать конструкции программной среды для программирования линейных, разветвлённых и циклических алгоритмов;

– организовывать ввод данных различных типов в соответствии с поставленной задачей;

– использовать необходимые средства языка программирования для осуществления вывода данных в соответствии с заданием.

Курс является практикоориентированным, поэтому каждое занятие содержит как теоретический, так и практический материал.

Требования к кандидату в педагоги

Согласно профессиональному стандарту «Педагог дополнительного образования детей и взрослых», утвержденному приказом Министерства труда РФ от 08.09.2015 г., № 613н, требования к образованию и обучению педагога допобразования: среднее профессиональное образование - программы подготовки специалистов среднего звена или высшее образование, направленность (профиль) которого, как правило, соответствует направленности дополнительной общеобразовательной программы, осваиваемой учащимися, или преподаваемому учебному курсу, дисциплине (модулю). Дополнительное профессиональное образование - профессиональная переподготовка, направленность (профиль) которой соответствует направленности дополнительной общеобразовательной программы, осваиваемой учащимися, или преподаваемому учебному курсу, дисциплине (модулю). При отсутствии педагогического образования - дополнительное профессиональное педагогическое образование; дополнительная профессиональная программа может быть освоена после трудоустройства. Рекомендуется обучение по дополнительным профессиональным программам по профилю педагогической деятельности не реже чем один раз в три года. Педагог дополнительного образования, претендующий на работу с учащимися, должен обладать ярко выраженными педагогическими способностями, в частности, иметь:

- организаторские, проявляющиеся в умении учителя сплотить учащихся, занять их, разделить обязанности, спланировать работу, подвести итоги сделанному;

- дидактические - конкретные умения подобрать и подготовить учебный материал, наглядность, оборудование, доступно, ясно, выразительно, убедительно и последовательно излагать учебный материал, стимулировать развитие познавательных интересов и духовных потребностей, повышать учебно-познавательную активность;

- перцептивные, проявляющиеся в умении проникать в духовный мир воспитуемых, объективно оценивать их эмоциональное состояние, выявлять особенности психики;

- коммуникативные способности проявляются в умении учителя устанавливать педагогически целесообразные отношения с учащимися и их родителями, коллегами, руководителями учебного заведения;

- суггестивные способности заключаются в эмоционально-волевом влиянии на обучаемых;

- исследовательские способности, проявляющиеся в умении познать и объективно оценить педагогические ситуации и процессы;

- научно-познавательные, сводящиеся к способности усвоения научных знаний в избранной отрасли.

Требования к индивидуальным способностям и особенностям педагога:

- четкая, понятная устная речь;
- ответственность;
- нервно-психическая устойчивость;
- стрессоустойчивость, эмоциональная уравновешенность;
- умение управлять собой, личная организованность;
- социальный интеллект (умение понимать поведение других людей);
- коммуникативные и организаторские способности;
- высокая общая культура;
- широкая эрудиция;
- стремление к профессиональному самосовершенствованию.

-приветствуется наличие портфолио с достижениями, опыт участия на выставках и конкурсах профессионального мастерства.

Список используемой литературы

1. Босова Л.Л. Информатика : учебник для 6 класса / Л.Л. Босова, А.Ю. Босова. – М. : БИНОМ. Лаборатория знаний, 2013.– 213 с. : ил.
2. Босова Л.Л. Информатика : учебник для 8 класса / Л.Л. Босова, А.Ю. Босова. – 3-е изд. –М. : БИНОМ. Лаборатория знаний, 2015.– 160 с. : ил.
3. Босова Л.Л., Методика применения интерактивных сред для обучения младших школьников программированию / Л.Л. Босова, Т.Е. Сорокина //Информатика и образование. – №7 (256). – 2014.
4. Сорокина Т.Е. Визуальная среда Scratch как средство мотивации учащихся основной школы к изучению программирования //Информатика и образование. – №5 (264). – 2015.

5. Сорокина Т.Е. Методика раннего общедоступного программирования в основной образовательной программе. Сборник научных трудов XI Международной научно- практической конференции «Современные информационные технологии и ИТ- образование». 2016. Т. 12. № 3-1. С. 228-232.

6. Мэтиз Эрик. Python Crash Course: A Hands-On, Project-Based Introduction to Programming. - Питер, 2017, ISBN: 978-5-496-02305-4

7. Бьерн Страуструп. Язык программирования C++. Специальное издание.