

Рассмотрено
на заседании ШМО
29 августа 2022г
Протокол № 7
Рук.МО А.Р. Хамидуллина А.Р.

Согласовано
Заместитель директора по ВР
А.Р. Хамидуллина А.Р.
29 августа 2022 г

Утверждено:
Директор школы
И.Ф. Муллануров И.Ф.
Приказ № 95
от 29 августа 2022 г



**ДОПОЛНИТЕЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON И C++»**

на 2022-2025 учебный год
возраст детей 15-17 лет

Педагог дополнительного образования
Газизов Рамис Рафисович

с. Исенбаево, 2022 год.

Пояснительная записка

На базе центра «Точка роста» обеспечивается реализация образовательных программ технологической направленности, разработанных в соответствии с требованиями законодательства в сфере образования и с учётом рекомендаций Федерального оператора учебных предметов «Информатика» и «Технология».

Программа «Основы языка программирования» составлена на основе:

- курса Д.П. Кириенко «Основы языка программирования *Python*»;
- учебного пособия для общеобразовательных организаций К.Ю. Полякова «Программирование. *Python. C++*» в 4-х частях.

Программа «Основы программирования» направлена на подготовку творческой, технически грамотной, гармонично развитой личности, обладающей логическим мышлением, способной анализировать и решать задачи в команде в области информационных технологий, решать ситуационные кейсовые задания, основанные на групповых проектах. Занятия по данному курсу рассчитаны на общенаучную подготовку обучающихся, развитие их мышления, логики, математических способностей, исследовательских навыков. Курс направлен на изучение основ программирования на языке *Python* и на языке *C++*. В рамках курса обучающиеся смогут познакомиться с физическими, техническими и математическими понятиями. Приобретённые знания будут применимы в творческих проектах.

Актуальность курса состоит в том, что он направлен:

- на формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики за счет развития представлений об информации как важнейшем стратегическом ресурсе развития личности, государства, общества;
- на понимание роли информационных процессов в современном мире;

- на совершенствование общеучебных и общекультурных навыков работы с информацией в процессе систематизации и обобщения имеющихся и получения новых знаний, умений и способов деятельности в области информатики и ИКТ;
- на развитие навыков самостоятельной учебной деятельности школьников (учебного проектирования, моделирования, исследовательской деятельности и т.д.);
- на воспитание ответственного и избирательного отношения к информации с учётом правовых и этических аспектов ее распространения, воспитанию стремления к продолжению образования и созидательной деятельности с применением средств ИКТ.

Использование оборудования «Точка роста» при реализации данной образовательной программы позволяет создать условия:

- для расширения содержания школьного образования по информатике и технологии;
- для повышения познавательной активности обучающихся в области программирования;
- для развития личности ребёнка в процессе обучения информатики и технологии, его способностей, формирования и удовлетворения социально значимых интересов и потребностей;
- для работы с одарёнными школьниками, организации их развития в различных областях образовательной, творческой деятельности.

Общая характеристика программы

Умение программировать на разных языках – это обязательное требование к разработчику программного обеспечения.

Программа по предмету «Основы языка программирования» предназначена для изучения всех основных разделов курса программирования на базовом уровне.

В рамках предлагаемого курса изучение основ программирования на языках *Python* и *C++* — это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков. Роль программирования велика для формирования мышления, приёмов умственных действий, умения строить модели, самостоятельного нахождения и составления алгоритмов решения задач, умения чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования общеинтеллектуальных и общеучебных умений школьников активизирует процесс индивидуально-личностного становления учащихся. Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции — одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных школьников.

Язык программирования *C++* – основной язык для создания игр и операционных систем. Программы на *C++* работают значительно быстрее, чем на других языках. Язык программирования *Python* предназначен для разработки сайтов, решения задач по биоинформатике и обработки больших данных.

Цель изучения курса:

- формирование интереса обучающихся к изучению профессий, связанных с основами программирования через освоение языков *Python* и *C++*;

- развитие алгоритмических способностей учащихся по программированию;
- формирование знаний, умений и навыков в области программирования на языках *Python* и *C++*.

Задачи:

Обучающие:

- освоение основных алгоритмических конструкций;
- обучение основам алгоритмизации и программирования;
- освоение первоначальных навыков программирования на языках программирования *Python* и *C++*;
- приобщение к проектно-творческой деятельности.

Воспитывающие:

- воспитание интереса к информационной и коммуникационной сфере человеческой деятельности;
- воспитание потребности соблюдать этические и правовые нормы работы с информацией;
- воспитание бережного отношения к техническим устройствам.

Развивающие:

- развитие творческого воображения, алгоритмического мышления учащихся;
- развитие навыков планирования проекта, умения работать в группе;
- развитие навыков ориентации в информационных потоках окружающего мира и применения точной и понятной инструкции для решения учебных задач и в повседневной жизни.

Планируемый результат реализации программы

Основным результатом обучения является формирование вектора развития обучающихся с упором на формирование у них алгоритмического мышления.

Личностные результаты:

- сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов;
- сформированность представлений о мире профессий, связанных с программированием, и требованиях, предъявляемых различными востребованными профессиями, такими как программист, системный администратор;
- навыки сотрудничества в образовательной, учебно-исследовательской, проектной и других видах деятельности;
- навыки самооценки, навыки рефлексии.

Метапредметные результаты:

- владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем;
- способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания.

Предметные результаты:

- навыки алгоритмического мышления и понимание необходимости формального описания алгоритмов;
- владение стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ на языках *Python* и *C++*;

- знание особенностей структуры программ, представленных на языках *Python* и *C++*;
- представление о модулях, входящих в состав сред *Python* и *C++*;
- возможности и ограничения использования готовых модулей;
- представление о величине, ее характеристиках;
- знание что такое операция, операнд и их характеристики;
- принципиальные отличия величин, структурированных и не структурированных;
- представление о таких структурах данных, как число, текст, кортеж, список, словарь;
- представление о составе арифметического выражения;
- знание математических функций, входящих в *Python* и *C++*, представление о логических выражениях и входящих в них операндах, операциях и функциях;
- умение записывать примеры арифметических и логических выражений всех атрибутов, которые могут в них входить;
- знание основных операторов языков *Python* и *C++*, их синтаксис;
- представление о процессе исполнения каждого из операторов;
- умение разрабатывать программы обработки числовой и символьной информации;
- умение разрабатывать программы (линейные, разветвляющиеся и с циклами);
- представление о значении полноценных процедур и функций для структурно-ориентированного языка высокого уровня;
- правила описания функций в *Python* и *C++* и построение вызова;
- принципиальные отличия между формальными, локальными и глобальными переменными.

Сроки реализации программы, форма и режим занятий

Программа рассчитана на 210 учебных занятий в течение 3 лет. Режим занятий – 2 часа в неделю.

Основными видами учебной деятельности учащихся является компьютерный практикум и компьютерный эксперимент по предложенным учебным материалам.

Основная форма обучения: практические работы на компьютере.

Формы подведения итогов реализации программы

Предметом диагностики и контроля в курсе «Основы программирования» являются образовательные продукты учащихся (созданные блок-схемы, программы), а также личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам курса.

Качество образовательной продукции оценивается по следующим параметрам:

- алгоритм должен быть оптимальным по скорости выполнения и максимально простым в реализации на языке программирования;
- программа должна выполнять поставленные задачи;
- по степени «читаемости кода» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.).

Проверка достигаемых учащимися результатов производится в следующих формах:

- текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
- текущая диагностика и оценка учителем деятельности школьников;
- итоговая оценка деятельности и образовательной продукции ученика в соответствии с его индивидуальной образовательной программой освоения курса.

Содержание учебного материала первого года
Основы программирования на языке Python. Часть 1

1. Линейные алгоритмы (20 ч.)

Общие сведения о языке Python. Установка программы Python. Режимы работы с Python. Функция вывода print(). Математические операции на Python. Задачи на элементарные действия с числами. Переменные. Выражения с переменными. Функция ввода input(). Числовые типы данных int() и float(). Стандартные математические функции. Параметры sep и end. Перенос строки с помощью \n. Форматирование методом %.f%.

2. Условия (15 ч.)

Операторы сравнения. Условный оператор if-else. Условный оператор if-else. Каскадные условия. Вложенные условия. Генератор случайных чисел. Сложные условия and, or и not.

3. Циклы for и while (16 ч.)

Цикл for. Функция range() в цикле for. Операторы break и continue в цикле for. Вложенные циклы for. Цикл while. Операторы break и continue в цикле while. Вложенные циклы while.

4. Строки (14 ч.)

Создание строк. Базовые операции над строками. Срезы строк. Форматирование строк методом f-string. Методы строк. Сравнение и сортировка строк.

5. Итоговое повторение + резерв (5 ч.)

Содержание учебного материала второго года
Основы программирования на языке Python. Часть 2

1. Функции (13 ч.)

Функции. Глобальные и локальные переменные. Lambda – функция. Рекурсия.

2. Списки и кортежи (12 ч.)

Понятие списка и кортежа. Создание списков и кортежей. Срезы списков. Операции со списками. Методы списков.

3. Словари (8 ч.)

Понятие словаря. Создание словарей. Операции над словарями. Методы словарей.

4. Итоговое повторение основ программирования на языке Python (3 ч.)

Основы программирования на языке C++. Часть 1

5. Общие сведения о языке программирования C++ (12 ч.)

История и особенности языка программирования C++. Структурное программирование. Объектно-ориентированное программирование. Установка интегрированной среды разработки. Компиляция программ. Режимы конфигурации. Решение распространенных проблем. Алфавит языка. Идентификаторы. Ключевые слова. Константы. Комментарии. Основные операции языка. Знаки операций. Порядок выполнения операций. Модульное программирование. Функции. Объявление и определение функций. Параметры функций. Функция main(). Функции стандартной библиотеки. Функции ввода вывода. Математические функции. Функции работы со строками и символами.

6. Базовые типы данных C++ (7 ч.)

Концепция типов данных C++. Базовые типы данных. Целые типы. Символьные типы. Вещественный тип данных. Операции и стандартные функции. Перечисляемый тип enum. Тип void. Преобразование типов. Неявное преобразование типов.

7. Базовые конструкции структурного программирования. Часть 1 (11 ч.)

Структура программы. Подключение других файлов в код. Комментарии. Базовые конструкции структурного программирования. Линейное программирование. Переменные. Инициализация переменных. Формы оператора присваивания. Основные операции языка C++. Выражения. Присваивание переменной значения выражения. Базовое форматирование кода. Ввод и вывод данных. Создание простейшей программы для демонстрации ввода и вывода данных. Компиляция и запуск.

8. Итоговое повторение + резерв (4 ч.)

Содержание учебного материала третьего года

Основы программирования на языке C++. Часть 2

1. Базовые конструкции структурного программирования. Часть 2 (43 ч.)

Выполнение расчетов по формулам. Выражения смешанного типа. Преобразование типов. Генерация случайных чисел. Управление ходом программы с помощью команд ветвления. Условный оператор if. Логические операции. Оператор выбора switch. Разработка программ с использованием ветвления. Организация циклических процессов. Оператор цикла с параметром for. Разработка программ с использованием цикла с параметром. Цикл с предусловием while. Разработка программ с использованием цикла с предусловием. Цикл с постусловием do while. Разработка программ с использованием цикла с постусловием. Операторы передачи управления: goto, break, continue, return. Разработка программ с использованием операторов передачи управления. Исключения в языке C++.

2. Производные типы данных (22 ч.)

Пользовательские типы данных. Переименование типов. Структуры. Объединения. Перечисления. Обработка перечислений. Вывод перечислений. Массивы. Описание одномерных массивов. Обработка одномерных массивов. Многомерные массивы. Описание двумерных массивов. Обработка двумерных массивов. Строки. Функции для работы со строками. Указатели. Сходства и различия между массивами и указателями. Объявление, присваивание значений указателю. Динамическое выделение памяти. Динамические массивы. Ссылки и их назначение.

3. Итоговое повторение основ программирования на языке C++ (5 ч.)

Тематическое планирование

<i>Основы программирования на языке Python</i>			
№	Тема	Кол-во часов	Кол-во контрольных точек
1.	Линейные алгоритмы	20	2
2.	Условия	15	2
3.	Циклы for и while	16	2
4.	Строки	14	2
5.	Итоговое повторение + резерв	5	1
Итого		70	9
1.	Функции	13	2
2.	Списки и кортежи	12	2
3.	Словари	8	2
4.	Итоговое повторение основ программирования на языке Python	3	1
<i>Основы программирования на языке C++</i>			
5.	Общие сведения о языке программирования C++	12	1
6.	Базовые типы данных C++	7	1
7.	Базовые конструкции структурного программирования. Часть 1	11	1
8.	Итоговое повторение + резерв	4	1
Итого		140	20
1.	Базовые конструкции структурного программирования. Часть 1	43	4
2.	Производные типы данных	22	2
3.	Итоговое повторение основ программирования на языке C++	5	1
Итого		210	27

Календарно-тематическое планирование

ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON ЧАСТЬ 1			
№	Тема	Дата	Примечания
<i>Линейные алгоритмы</i>			
1.	Общие сведения о языке Python. Установка программы Python. Режимы работы с Python		
2.	Функция вывода print()		
3.	Функция вывода print()		
4.	Математические операции на Python. Задачи на элементарные действия с числами		
5.	Математические операции на Python. Задачи на элементарные действия с числами		
6.	Переменные. Выражения с переменными		
7.	Переменные. Выражения с переменными		
8.	Переменные. Выражения с переменными		
9.	Функция ввода input()		
10.	Функция ввода input()		
11.	Функция ввода input()		
12.	Числовые типы данных int() и float()		
13.	Числовые типы данных int() и float()		
14.	Стандартные математические функции		
15.	Стандартные математические функции		
16.	Стандартные математические функции		
17.	Параметры sep и and. Перенос строки с помощью \n		
18.	Форматирование методом %.f%		

19.	Проектная работа		
20.	Защита проектных работ		
<i>Условия</i>			
21.	Операторы сравнения. Условный оператор if-else		
22.	Условный оператор if-else		
23.	Условный оператор if-else		
24.	Каскадные условия		
25.	Каскадные условия		
26.	Каскадные условия		
27.	Вложенные условия		
28.	Вложенные условия		
29.	Вложенные условия		
30.	Генератор случайных чисел		
31.	Сложные условия and, or и not		
32.	Сложные условия and, or и not		
33.	Сложные условия and, or и not		
34.	Проектная работа		
35.	Защита проектных работ		
<i>Циклы for и while</i>			
36.	Цикл for		
37.	Цикл for		
38.	Функция range() в цикле for		
39.	Функция range() в цикле for		

40.	Операторы break и continue в цикле for		
41.	Операторы break и continue в цикле for		
42.	Вложенные циклы for		
43.	Вложенные циклы for		
44.	Цикл while		
45.	Цикл while		
46.	Операторы break и continue в цикле while		
47.	Операторы break и continue в цикле while		
48.	Вложенные циклы while		
49.	Вложенные циклы while		
50.	Проектная работа		
51.	Защита проектных работ		
<i>Строки</i>			
52.	Создание строк. Базовые операции над строками		
53.	Создание строк. Базовые операции над строками		
54.	Срезы строк		
55.	Срезы строк		
56.	Срезы строк		
57.	Форматирование строк методом f-string		
58.	Методы строк		
59.	Методы строк		
60.	Методы строк		
61.	Методы строк		

62.	Сравнение и сортировка строк		
63.	Сравнение и сортировка строк		
64.	Проектная работа		
65.	Защита проектных работ		
<i>Итоговое повторение</i>			
66.	Итоговое повторение пройденного материала		
67.	Итоговое повторение пройденного материала		
68.	Контрольное тестирование		
69.	Резерв		
70.	Резерв		
Итого – 70 часов			
ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON ЧАСТЬ 2			
<i>Функции</i>			
71.	Функции		
72.	Функции		
73.	Функции		
74.	Функции		
75.	Глобальные и локальные переменные		
76.	Глобальные и локальные переменные		
77.	Lambda - функция		
78.	Lambda - функция		

79.	Рекурсия		
80.	Рекурсия		
81.	Рекурсия		
82.	Проектная работа		
83.	Защита проектных работ		
<i>Списки и кортежи</i>			
84.	Понятия списка и кортежа. Создание списков и кортежей		
85.	Создание списков и кортежей		
86.	Срезы списков		
87.	Срезы списков		
88.	Срезы списков		
89.	Операции со списками		
90.	Операции со списками		
91.	Методы списков		
92.	Методы списков		
93.	Методы списков		
94.	Проектная работа		
95.	Защита проектных работ		
<i>Словари</i>			
96.	Понятие словаря. Создание словарей.		
97.	Операции над словарями.		
98.	Операции над словарями.		
99.	Методы словарей		

100.	Методы словарей		
101.	Методы словарей		
102.	Проектная работа		
103.	Защита проектных работ		
<i>Итоговое повторение</i>			
104.	Итоговое повторение основ программирования на языке Python		
105.	Итоговое повторение основ программирования на языке Python		
106.	Итоговая контрольная работа по программированию на языке Python		
ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ C++. ЧАСТЬ 1			
<i>Общие сведения о языке программирования C++</i>			
107.	История и особенности языка программирования C++. Структурное программирование. Объектно-ориентированное программирование		
108.	Установка интегрированной среды разработки		
109.	Компиляция программ. Режимы конфигурации		
110.	Решение распространенных проблем		
111.	Алфавит языка. Идентификаторы. Ключевые слова		
112.	Алфавит языка. Идентификаторы. Ключевые слова		
113.	Константы. Комментарии		
114.	Основные операции языка. Знаки операций. Порядок выполнения операций		
115.	Модульное программирование. Функции. Объявление и определение функций. Параметры функций. Функция main().		
116.	Функции стандартной библиотеки. Функции ввода вывода		
117.	Математические функции. Функции работы со строками и символами		
118.	Тестирование по теме «Общие сведения о языке программирования C++»		

<i>Базовые типы данных C++</i>		
119.	Концепция типов данных C++. Базовые типы данных. Целые типы. Символьные типы	
120.	Концепция типов данных C++. Базовые типы данных. Целые типы. Символьные типы	
121.	Вещественный тип данных. Операции и стандартные функции	
122.	Перечисляемый тип enum. Тип void	
123.	Преобразование типов. Неявное преобразование типов	
124.	Преобразование типов. Неявное преобразование типов	
125.	Тестирование по теме «Базовые типы данных C++»	
<i>Базовые конструкции структурного программирования. Часть 1</i>		
126.	Структура программы. Подключение других файлов в код. Комментарии	
127.	Базовые конструкции структурного программирования. Линейное программирование	
128.	Переменные. Инициализация переменных	
129.	Формы оператора присваивания	
130.	Основные операции языка C++. Выражения. Присваивание переменной значения выражения	
131.	Базовое форматирование кода	
132.	Ввод и вывод данных	
133.	Ввод и вывод данных	
134.	Создание простейшей программы для демонстрации ввода и вывода данных. Компиляция и запуск	
135.	Создание простейшей программы для демонстрации ввода и вывода данных. Компиляция и запуск	
136.	Тестирование по теме «Базовые конструкции структурного программирования»	
<i>Итоговое повторение</i>		
137.	Итоговое повторение пройденного материала	

138.	Контрольное тестирование		
139.	Резерв		
140.	Резерв		
Итого – 140 часов			
ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ C++. ЧАСТЬ 2			
<i>Базовые конструкции структурного программирования. Часть 2</i>			
141.	Выполнение расчетов по формулам		
142.	Выполнение расчетов по формулам		
143.	Выражения смешанного типа. Преобразование типов		
144.	Преобразование типов. Составление программ		
145.	Преобразование типов. Составление программ		
146.	Генерация случайных чисел		
147.	Управление ходом программы с помощью команд ветвления. Условный оператор if. Логические операции		
148.	Оператор выбора switch		
149.	Разработка программ с использованием ветвления		
150.	Разработка программ с использованием ветвления		
151.	Разработка программ с использованием ветвления		
152.	Разработка программ с использованием ветвления		
153.	Разработка программ с использованием ветвления		
154.	Разработка программ с использованием ветвления		

155.	Разработка программ с использованием ветвления		
156.	Разработка программ с использованием ветвления		
157.	Разработка программ с использованием ветвления		
158.	Разработка программ с использованием ветвления		
159.	Проектная работа		
160.	Защита проектных работ		
161.	Организация циклических процессов. Оператор цикла с параметром for		
162.	Разработка программ с использованием цикла с параметром		
163.	Разработка программ с использованием цикла с параметром		
164.	Разработка программ с использованием цикла с параметром		
165.	Разработка программ с использованием цикла с параметром		
166.	Цикл с предусловием while		
167.	Разработка программ с использованием цикла с предусловием		
168.	Разработка программ с использованием цикла с предусловием		
169.	Разработка программ с использованием цикла с предусловием		
170.	Разработка программ с использованием цикла с предусловием		
171.	Цикл с постусловием do while		
172.	Разработка программ с использованием цикла с постусловием		
173.	Разработка программ с использованием цикла с постусловием		
174.	Разработка программ с использованием цикла с постусловием		
175.	Разработка программ с использованием цикла с постусловием		
176.	Операторы передачи управления: goto, break, continue, return		
177.	Разработка программ с использованием операторов передачи управления		

178.	Разработка программ с использованием операторов передачи управления		
179.	Разработка программ с использованием операторов передачи управления		
180.	Разработка программ с использованием операторов передачи управления		
181.	Исключения в языке C++		
182.	Проектная работа		
183.	Защита проектных работ		
<i>Производные типы данных</i>			
184.	Пользовательские типы данных. Переименование типов. Структуры. Объединения		
185.	Перечисления. Обработка перечислений. Вывод перечислений		
186.	Перечисления. Обработка перечислений. Вывод перечислений		
187.	Массивы. Описание и обработка одномерных массивов		
188.	Обработка одномерных массивов. Составление программ		
189.	Обработка одномерных массивов. Составление программ		
190.	Обработка одномерных массивов. Составление программ		
191.	Обработка одномерных массивов. Составление программ		
192.	Многомерные массивы. Описание двумерных массивов		
193.	Обработка двумерных массивов. Составление программ		
194.	Обработка двумерных массивов. Составление программ		
195.	Обработка двумерных массивов. Составление программ		
196.	Обработка двумерных массивов. Составление программ		
197.	Строки. Функции для работы со строками		
198.	Строки. Функции для работы со строками		
199.	Строки. Функции для работы со строками		

200.	Указатели. Сходства и различия между массивами и указателями. Объявление, присваивание значений указателю		
201.	Динамическое выделение памяти. Динамические массивы		
202.	Динамическое выделение памяти. Динамические массивы		
203.	Ссылки и их назначение		
204.	Проектная работа		
205.	Защита проектных работ		
<i>Итоговое повторение</i>			
206.	Итоговое повторение пройденного материала		
207.	Итоговое повторение пройденного материала		
208.	Контрольное тестирование		
209.	Резерв		
210.	Резерв		
Итого – 210 часов			