

Управление образования администрации
Богородского муниципального округа
Муниципальное бюджетное образовательное учреждение
«Школа №1»

Принята на заседании
Педагогического совета
от 31 августа 2021г
Протокол № 1



Утверждаю
Директор МБОУ «Школа №1»
Багаева И.А.
от 31 августа 2021г

Дополнительная общеобразовательная
(общеразвивающая) программа
технической направленности
«Основы программирования на языке Python»

Возраст обучающихся: 15–18 лет
Срок реализации: 1 год

Автор-составитель:
Санкин Алексей Алексеевич,
педагог дополнительного образования

г. Богородск
2021

Пояснительная записка

Актуальность программы

В современном обществе информационные технологии имеют огромное значение – почти все отрасли нашей жизни в той или иной мере компьютеризированы, поэтому одним из факторов, предсказывающих успешность индивида в обществе является способность к работе с компьютером на всех доступных уровнях. Целью данного курса является развитие у учащихся компетенций в области разработки программного обеспечения, а также алгоритмического мышления, навыков проектной деятельности и применения компьютерных средств в учебной и трудовой деятельности. Обучение в рамках курса способно направить индивидуальную образовательную траекторию учащихся к профессиональному будущему в сфере информационных технологий.

Направленность программы

Данный курс имеет техническую направленность и нацелен на развитие исследовательских, прикладных и творческих способностей учащихся в области программирования.

Отличительные особенности

Программа предназначена для вхождения в программирование на языке Python и рассчитана на учащихся без опыта программирования на этом языке, или же без опыта программирования вообще.

Адресат программы

Курс нацелен для учащихся 8-11 классов, прием осуществляется по желанию. Рекомендованный размер учебной группы – 12-15 человек.

Цель и задачи программы

Целью программы является повышение интереса и развитие способностей учащихся к программированию через ознакомление с основами языка Python.

В рамках достижения данной цели ставятся следующие задачи:

Обучающие:

- Ознакомление учащихся с особенностями синтаксиса и семантики языка программирования Python.
- Ознакомление учащихся с наиболее часто используемыми операторами языка Python.
- Ознакомление учащихся с линейными, условными и циклическими алгоритмами и их реализацией на языке Python.

Развивающие:

- Развитие способностей к самостоятельному целеполаганию в учебной и трудовой деятельности.
- Развитие способностей к самостоятельному планированию путей к достижению поставленных целей.
- Развитие творческих способностей учащихся в области программирования.
- Развитие у учащихся алгоритмического мышления.

Воспитывающие:

- Воспитание стремления к творческому решению задач.
- Воспитание представлений о культуре программирования.

Объем и срок освоения программы

Срок реализации программы – 1 учебный год, 72 часа в год. Она состоит из тринадцати разделов.

Формы обучения

Занятия проходят в компьютерном классе и включают в себя теоретические занятия и выполнение практических работ в среде программирования.

Режим занятий

Занятия проводятся два раза в неделю по 45 минут, всего на них отводится 72 учебных часа.

Планируемые результаты

Основные *личностные результаты* освоения курса:

- формирование мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование творческого сознания через решение задач программирования.

Основные *метапредметные результаты* освоения курса:

- развитие навыков самостоятельного целеполагания и постановки задач;
- развитие способности к планированию путей решения поставленных проблем;
- развитие навыков контроля и самокоррекции;
- развитие способностей к алгоритмизации своей учебной и повседневной деятельности.

Основные *предметные результаты* освоения курса:

- овладение основами составления и чтения программ на языке Python;
- формирование и расширение представлений об основных предметных понятиях – «алгоритм», «программа», «оператор», «функция»;
- развитие логических способностей и алгоритмического мышления;
- развитие навыков составления линейных, условных и циклических алгоритмов на языке Python;
- расширение представления о культуре программирования.

Учебный план

№ п/п	Название раздела, темы	Количество часов			Формы аттестации / контроля
		Всего	Теория	Практика	
1	Введение в программирование. Знакомство с программной средой Python	2	2		Наблюдение, тестирование
2	Алгоритмы. Виды и представления алгоритмов	3	2	1	Самостоятельная работа
3	Линейные алгоритмы	5	2	3	Контрольная работа

4	Вычислительные задачи	4	1	3	Самостоятельная работа
5	Функции в Python	4	1	3	Контрольная работа
6	Условные алгоритмы	5	2	3	Контрольная работа
7	Задачи математической логики	5	2	3	Самостоятельная работа
8	Циклические алгоритмы	6	3	3	Контрольная работа
9	Массивы	13	4	9	Контрольная работа
11	Матричные задачи	5	2	3	Самостоятельная работа
12	Работа с файлами	5	2	3	Контрольная работа
13	Графика в Python	9	3	6	Контрольная работа
14	Обобщение изученного	6		6	Зачет
	Всего часов	72	26	46	

Содержание учебного плана

№ п/п	Перечень разделов	Теория	Практика
1	Введение в программирование. Знакомство с программной средой Python	Понятие программирования. Структура программы. Типы данных. Переменные. Идентификаторы. Особенности синтаксиса и семантики. Основные элементы пользовательского интерфейса программной среды. Примеры программ.	Запуск программной среды. Написание программного кода. Ввод и вывод данных в среде Python.
2	Алгоритмы. Виды и представления алгоритмов	Понятие алгоритма. Свойства алгоритма. Виды алгоритмов. Способы представления алгоритма.	Выявление и составление алгоритмов. Различия алгоритмов по видам и по формам записи. Запись алгоритмов в словесно-формульной форме. Запись алгоритмов в форме блок-схем.

			Написание и отладка программного кода.
3	Линейные алгоритмы	Линейный алгоритм. Признаки линейного алгоритма. Арифметические операторы. Операторы присваивания. Понятие множества.	Решение задач на линейное преобразование данных. Запись алгоритмов в форме блок-схем. Написание и отладка программного кода.
4	Вычислительные задачи	Понятие вычислительных задач. Особенности решения вычислительных задач в среде программирования.	Решение вычислительных задач. Запись алгоритмов в форме блок-схем. Написание и отладка программного кода.
5	Функции в Python	Понятие функции. Аргументы функции. Рекурсия.	Решение задач с использованием функций. Запись алгоритмов в форме блок-схем. Написание и отладка программного кода.
6	Условные алгоритмы	Условный алгоритм. Признаки условного алгоритма. Условные операторы. Операторы сравнения. Логические операторы. Операторы членства и тождественности.	Решение задач с условием. Запись алгоритмов в форме блок-схем. Написание и отладка программного кода.
7	Задачи математической логики	Математическая логика. Истинные и ложные высказывания. Логические операторы. Таблицы истинности. Булевы функции.	Составление и решение задач математической логики. Алгоритмизация задач математической логики. Написание и отладка программного кода.
8	Циклические алгоритмы	Циклический алгоритм. Признаки циклического алгоритма. Виды циклов. Операторы цикла. Итерация.	Решение задач с циклами. Запись алгоритмов в форме блок-схем. Написание и отладка программного кода.
9	Массивы	Массив. Список. Индекс. Перебор. Работа с массивами. Строки.	Решение задач с массивами. Запись алгоритмов в форме блок-

		Операции со строками.	схем. Написание и отладка программного кода.
10	Матричные задачи	Понятие матрицы. Алгебраические операции над матрицами. Виды матриц. Транспонирование матрицы. Представление матриц в среде программирования.	Решение задач с матрицами. Запись алгоритмов в форме блок-схем. Написание и отладка программного кода.
11	Работа с файлами	Файл. Открытие и закрытие файлов. Чтение и запись файлов.	Решение задач с файлами. Запись алгоритмов в форме блок-схем. Написание и отладка программного кода.
12	Графика в Python	Компьютерная графика. Растровая и векторная графика. Графические операторы.	Решение графических задач. Построение и чтение графиков. Написание и отладка программного кода.
13	Обобщение изученного	Повторение пройденного материала.	Запись алгоритмов в форме блок-схем. Написание и отладка программного кода.

Календарный учебный график

№ п/п	Дата		Тема занятия	Кол-во часов	Форма занятия	Форма контроля
	По плану	Фактически				
1			Введение в программирование	1	Фронтальное	Наблюдение
2			Знакомство с программной средой Python	1	Фронтальное	Наблюдение
3			Алгоритм и его свойства	1	Фронтальное	Наблюдение
4			Виды алгоритмов. Способы представления алгоритма	1	Фронтальное	Наблюдение
5			Составление алгоритмов и	1	Фронтальное	Самостоятельная работа

			блок-схем			
6			Линейный алгоритм и его признаки	1	Фронтальное	Наблюдение
7			Арифметические операторы и операторы присваивания. Множества	1	Фронтальное	Наблюдение
8-9			Составление линейных алгоритмов на языке Python	2	Индивидуальное	Анализ программ, наблюдение
10			Контрольная работа по разделу: Линейный алгоритм	1	Индивидуальное	Анализ программ
11			Вычислительные задачи и особенности их решения в среде программирования	1	Фронтальное	Наблюдение
12-13			Решение вычислительных задач в среде программирования	2	Индивидуальное	Анализ программ
14			Функции и их свойства	1	Фронтальное	Наблюдение
15-17			Решение задач с использованием методов функционального программирования	3	Индивидуальное	Анализ программ
18			Условный алгоритм и его признаки	1	Фронтальное	Наблюдение
19			Условные операторы. Операторы сравнения. Логические операторы. Операторы	1	Фронтальное	Наблюдение

			членства и тождественности			
20 - 21			Решение задач с условием	2	Индивидуальное	Анализ программ
22			Контрольная работа по разделу: Условный алгоритм	1	Индивидуальное	Анализ программ
23			Математическая логика	1	Фронтальное	Наблюдение
24			Логические операторы. Таблицы истинности. Булевы функции.	1	Фронтальное	Наблюдение
25 - 27			Алгоритмизация задач математической логики на языке Python	3	Индивидуальное	Анализ программ
28			Циклический алгоритм и его признаки	1	Фронтальное	Наблюдение
29			Виды циклов	1	Фронтальное	Наблюдение
30			Циклы на языке программирования Python	1	Фронтальное	Наблюдение
31 - 32			Решение задач с циклами	2	Индивидуальное	Анализ программ
33			Контрольная работа по разделу: Циклический алгоритм	1	Индивидуальное	Анализ программ
34			Массивы. Алгоритм обработки массивов	1	Фронтальное	Наблюдение
35 - 36			Создание и обработка массивов	2	Индивидуальное	Анализ программ
37			Сортировка массива	1	Фронтальное	Наблюдение
38 -			Решение задач на сортировку	2	Индивидуальное	Анализ программ

39						
40			Поиск в массиве. Двоичный поиск	1	Фронтальное	Наблюдение
41 - 42			Решение задачи на поиск в массиве	2	Индивидуальное	Анализ программ
43			Строки	1	Фронтальное	Наблюдение
44 - 46			Решение задач со строками	3	Индивидуальное	Анализ программ
47			Контрольная работа по разделу: Массивы	1	Индивидуальное	Анализ программ
48			Матрицы. Виды и свойства	1	Фронтальное	Наблюдение
49			Представление матриц в среде программирования	1	Фронтальное	Наблюдение
50 - 52			Решение задач с матрицами в среде программирования	3	Индивидуальное	Анализ программ
53			Файлы в среде программирования	1	Фронтальное	Наблюдение
54			Чтение и запись файлов	1	Фронтальное	Наблюдение
55 - 56			Решение задач с файлами	2	Индивидуальное	Анализ программ
57			Контрольная работа по разделу: Работа с файлами	1	Индивидуальное	Анализ программ
58			Компьютерная графика	1	Фронтальное	Наблюдение
59 - 60			Графические операторы в Python	2	Фронтальное	Наблюдение
61 - 65			Решение графических задач в среде програм-	5	Индивидуальное	Анализ программ

			мирования			
66			Контрольная работа по разделу: Графика в Python	1	Индивидуальное	Анализ программ
67 - 71			Обобщение изученного	5	Индивидуальное	Анализ программ
72			Итоговое зачетное занятие	1	Индивидуальное	Анализ программ

Формы аттестации

Курс предусматривает следующие формы аттестации:

- промежуточная – по завершении изучения каждого раздела: контрольная работа, самостоятельная работа, тестирование;
- итоговая – по завершении изучения курса: зачет.

№	Раздел программы	Форма контроля	Критерий оценки	Система оценки
1	Введение в программирование. Знакомство с программной средой Python	Тестирование из 10 вопросов	1 балл – 0-3 правильных ответов; 2 балла – 4-7 правильных ответов; 3 балла – 8-10 правильных ответов;	1 балл – низкий уровень освоения; 2 балла – средний уровень освоения; 3 балла – высокий уровень освоения.
2	Алгоритмы. Виды и представления алгоритмов	Тестирование 10 вопросов	1 балл – 0-3 правильных ответов; 2 балла – 4-7 правильных ответов; 3 балла – 8-10 правильных ответов;	1 балл – низкий уровень освоения; 2 балла – средний уровень освоения; 3 балла – высокий

				уровень освоения.
3	Линейные алгоритмы	Контрольная работа из 3 заданий: написание программ	1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания.	1 балл – низкий уровень освоения; 2 балла – средний уровень освоения; 3 балла – высокий уровень освоения.
4	Вычислительные задачи	Самостоятельная работа из 3 заданий: написание программ	1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания.	1 балл – низкий уровень освоения; 2 балла – средний уровень освоения; 3 балла – высокий уровень освоения.
5	Условные алгоритмы	Контрольная работа из 5 заданий: написание программ	1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания; 4 балла – 4 правильно выполненных задания; 5 баллов – 5 правильно выполненных заданий.	1-2 балла – низкий уровень освоения; 3-4 балла – средний уровень освоения; 5 баллов – высокий уровень освоения.
6	Задачи математической логики	Самостоятельная работа из 5 заданий: написание программ	1 балл – 1 правильно выполненное задание; 2 балла – 2 пра-	1-2 балла – низкий уровень освоения;

			<p>вильно выполненных заданий; 3 балла – 3 правильно выполненных задания; 4 балла – 4 правильно выполненных задания; 5 баллов – 5 правильно выполненных заданий.</p>	<p>3-4 балла – средний уровень освоения; 5 баллов – высокий уровень освоения.</p>
7	Функции в Python	Контрольная работа из 3 заданий: написание программ	<p>1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания.</p>	<p>1 балл – низкий уровень освоения; 2 балла – средний уровень освоения; 3 балла – высокий уровень освоения.</p>
8	Циклические алгоритмы	Контрольная работа из 4 заданий: написание программ	<p>1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания; 4 балла – 4 правильно выполненных задания.</p>	<p>1 балл – низкий уровень освоения; 2-3 балла – средний уровень освоения; 4 баллов – высокий уровень освоения.</p>
9	Массивы	Контрольная работа из 4 заданий: написание программ	<p>1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания; 4 балла – 4 пра-</p>	<p>1 балл – низкий уровень освоения; 2-3 балла – средний уровень освоения; 4 баллов – высокий</p>

			вильно выполнен- ных задания.	уровень освоения.
10	Матричные задачи	Самостоятельная работа из 3 заданий: написание программ	1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания.	1 балл – низкий уровень освоения; 2 балла – средний уровень освоения; 3 балла – высокий уровень освоения.
11	Работа с файлами	Контрольная работа из 3 заданий: написание программ	1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания.	1 балл – низкий уровень освоения; 2 балла – средний уровень освоения; 3 балла – высокий уровень освоения.
12	Графика в Python	Контрольная работа из 5 заданий: написание программ	1 балл – 1 правильно выполненное задание; 2 балла – 2 правильно выполненных заданий; 3 балла – 3 правильно выполненных задания; 4 балла – 4 правильно выполненных задания; 5 баллов – 5 правильно выполненных заданий.	1-2 балла – низкий уровень освоения; 3-4 балла – средний уровень освоения; 5 баллов – высокий уровень освоения.
13	Обобщение изученного	Зачетная работа из 4 заданий: написание программ	1 балл – 1 правильно выполненное задание; 2 балла – 2 пра-	1-2 балла – незачет; 3-4 балла – зачет.

			вильно выполненных заданий; 3 балла – 3 правильно выполненных задания; 4 балла – 4 правильно выполненных задания.	
--	--	--	-------------------------------------------------------------------------------------------------------------------------	--

Методические материалы

Теоретические занятия проводятся фронтально в форме лекции, в качестве дидактических материалов используются презентации. Во время практических занятий, контрольных и практических работ, а также сдачи зачета, учащиеся работают индивидуально, за персональными компьютерами, в качестве дидактических материалов используются карточки с заданиями.

Разделы или тема программы	Формы занятий	Приемы и методы организации образовательной деятельности (в рамках занятия)	Дидактический материал	Техническое оснащение занятий	Формы подведения итогов
Введение в программирование. Знакомство с программной средой Python	Лекция	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Тестирование
Алгоритмы. Виды и представления алгоритмов	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Тестирование

Линейные алгоритмы	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Контрольная работа
Вычислительные задачи	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Самостоятельная работа
Условные алгоритмы	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Контрольная работа
Задачи математической логики	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Самостоятельная работа
Функции в Python	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Контрольная работа

Циклические алгоритмы	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Контрольная работа
Массивы	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Контрольная работа
Матричные задачи	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Самостоятельная работа
Работа с файлами	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Контрольная работа
Графика в Python	Лекция, практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями, презентации	Персональные компьютеры, оборудованные средой разработки на языке Python	Контрольная работа

Обобщение изученного	Практическое занятие	Словесные, репродуктивные, наглядные, конспектирование, аудиовизуальные, практические	Карточки с заданиями	Персональные компьютеры, оборудованные средой разработки на языке Python	Зачет
----------------------	----------------------	---------------------------------------------------------------------------------------	----------------------	--------------------------------------------------------------------------	-------

Условия реализации программы

Для проведения занятий требуется:

1. Кабинет, оборудованный согласно требованиям СанПиН 2.4.4.3172-14;
2. Рабочее место преподавателя:
 - a. Рабочий стол;
 - b. Стул или кресло;
 - c. Персональный компьютер, укомплектованный монитором, клавиатурой, компьютерной мышью, принтером, а также мультимедийным проектором или интерактивной доской, оборудованный средой разработки на языке Python и выходом в сеть Интернет;
3. Рабочее место учащегося (на каждого обучающегося):
 - a. Рабочий стол(парта) (индивидуальные или один на двоих);
 - b. Стул или кресло;
 - c. Ноутбук, либо персональный компьютер, укомплектованный монитором, клавиатурой и компьютерной мышью, оборудованный средой разработки на языке Python;

Список литературы

Используемая литература

1. Поляков, К. Ю. П54 Информатика. 10 класс (базовый и углубленный уровни) (в 2 частях) : учебник. Ч. 2 / К. Ю. Поляков, Е. А. Еремин. – 2-е изд., стереотип. – М. : БИНОМ. Лаборатория знаний, 2020. – 351, [1] с. : ил.

2. Поляков, К. Ю. П54 Информатика (в двух частях). 11 класс. Ч. 2: учебник базового и углубленного уровней / К. Ю. Поляков, Е. А. Еремин. – 3-е изд., стереотип. – М. : Просвещение, 2021. – 302, [2] с. : ил.
3. Доусон, М., Програмируем на Python. – СПб.: Питер, 2014. – 416 с.: ил.;
4. Рейтц К., Шлюссер Т., Автостопом по Python. – СПб.: Питер, 2017. — 336 с.: ил. – (Серия «Бестселлеры O’Reilly»);
5. Сысоева М. В., Сысоев И. В., Программирование для «нормальных» с нуля на языке Python: Учебник. В двух частях. Часть 1 / Ответственный редактор: В. Л. Черный: – М.: Базальт СПО; МАКС Пресс, 2018. – 176 с. [+4 с. вкл]: ил. – (Библиотека ALT).

Интернет-ресурсы

1. The official home of the Python Programming Language [Электронный ресурс]. – URL: <https://www.python.org>
2. Pythonicway. [Электронный ресурс]. – URL: <http://pythonicway.com>
3. Справочная документация по языку Python3. [Электронный ресурс]. – URL: <https://docs-python.ru>
4. Python 3 для начинающих – уроки программирования. [Электронный ресурс]. – URL: <https://pythonworld.ru>
5. Python на русском – скрипты, библиотеки, модули. [Электронный ресурс]. – URL: <https://pythonru.com>

Официально-документальные материалы

1. Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
2. Стратегия развития воспитания в РФ на период до 2025 года. Распоряжение правительства Российской Федерации № 996-р от 29 мая 2015 года;
3. Концепция развития дополнительного образования детей. Распоряжение Правительства Российской Федерации от 04.09.2014 года № 1726-р;
4. Приказ МинПросвещения РФ «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (от 9 ноября 2018 г. N 196);

5. Методические рекомендации по проектированию дополнительных общеобразовательных программ (включая разноуровневые программы). Письмо Министерства образования и науки РФ от 18 ноября 2015 г. № 09-3242;
6. Письмо Министерства образования Нижегородской области от 30.05.2014 г. № 316-01-100-1674/14 «Методические рекомендации по разработке образовательной программы образовательной организации дополнительного образования»;
7. СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей».