

**МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**Муниципальное бюджетное общеобразовательное учреждение**  
**Петрозаводского городского округа**  
**МОУ «Средняя школа № 20»**

**ПРИНЯТА**

Педагогическим Советом  
МОУ «Средняя школа № 20»

Протокол № 1  
от 29.08.2025 г.



**РАБОЧАЯ ПРОГРАММА**  
**учебного предмета «Элективный курс Информатика»**  
для обучающихся 10–11 классов

## Пояснительная записка

Предлагаемый элективный курс адресован учащимся 10 – 11 классов. Главная цель курса - это реализация идеи предпрофильной подготовки учащихся, организация систематического и системного повторения, углубления и расширения знаний по теме «Алгоритмы и элементы программирования» курса информатики за период изучения в основной школе. Вопросы, рассматриваемые в курсе, выходят за рамки обязательного содержания. Вместе с тем, они тесно примыкают к основному курсу, поэтому данный элективный курс будет способствовать совершенствованию и развитию важнейших знаний и умений в области информатики, предусмотренных школьной программой. Данный курс является практикоориентированным, дает учащимся возможность познакомиться с основами программирования на языке Python и применить знания на практике при работе над проектами.

Язык программирования Python обладает рядом преимуществ перед другими языками: ясность кода, быстрота реализации. Это развивающийся язык, используемый в реальных проектах. Средства для работы с Python относятся к категории свободно распространяемого программного обеспечения. Python имеет обширную область применения: создание расширений к графическому редактору GIMP, программирование в офисном пакете OpenOffice.org, написание сценариев для пакета 3D-моделирования Blender, создание компьютерных игр и web-приложений.

Программа элективного курса составлена на основе основной образовательной программы полного среднего образования по информатике и предполагает совершенствование подготовки школьников по освоению раздела «алгоритмы и элементы программирования»

За основу элективного курса взят материал учебных изданий:

- К.Ю. Поляков. Программирование. Python. C++ : учебное пособие: в 4 частях: Просвещение, 2023. (варианты глав по программированию для изучающих python <http://kpolyakov.spb.ru/school/probook/python.htm>)

- Электронный учебник: Питонтьютор ([pythontutor.ru](http://pythontutor.ru)).

Курс ориентирован на предпрофильную подготовку учащихся по информатике, расширяет базовый курс по информатике, является практикоориентированным и дает учащимся возможность познакомиться с основами программирования на языке Python.

Обучение программированию является важным этапом в общеобразовательном развитии обучающегося, поскольку позволяет в наиболее общей и в то же время наглядной форме выработать навык применения

формальных операций к широкому кругу объектов.

**Цель элективного курса:** формирование базовых понятий программирования, знакомство с различными стилями программирования, развитие алгоритмического и логического мышления обучающихся.

**Задачи курса:**

- формирование интереса к изучению профессии, связанной с программированием;
- формирование у обучающихся представления о принципах построения языков программирования;
- углубление знаний об алгоритмических конструкциях и структурах данных;
- развитие алгоритмического и логического мышления;
- формирование навыков грамотной разработки программ;
- углубление знаний, умений и навыков решения задач по программированию и алгоритмизации.

**Курс включает:**

- знакомство с языком программирования Python, с концепцией языка,
- изучение синтаксиса языка, различных стилей программирования, методов разработки, кодирования и отладки программ,
- углубление знаний об алгоритмических конструкциях и структурах данных.

Благодаря тому, что рекомендуемые источники содержат большое количество заданий разного уровня сложности, можно составлять для каждого учащегося индивидуальное задание по каждой изучаемой теме, которое будет учитывать индивидуальные интересы ученика, уровень освоения учебного материала, особенности освоения учебного материала.

Текущий контроль уровня усвоения материала осуществляется по результатам выполнения практических работ, зачетных практических работ по основным темам данного элективного курса, а также по результатам выполнения мини-проектов.

Итоговый контроль реализуется в форме защиты итоговых проектов. Защита итогового проекта предполагает публичное выступление учащегося с демонстрацией результатов работы программы и презентации, содержащей аргументы в пользу практической ценности работы, анализ полученных результатов, оценку достигнутого прогресса в знаниях. Также обязательной частью отчёта является ответы на вопросы одноклассников.

Преобладающий тип занятий - практикум. Все задания курса выполняются с

помощью персонального компьютера. Форма занятий направлена на активизацию познавательной деятельности, на развитие алгоритмического, операционного мышления учащихся.

Уроки строятся в соответствии с требованием санитарных норм, теоретические и практические части занятий чередуются, во время работы за компьютером используются упражнения для глаз.

### **Требования к планируемым результатам**

#### *Личностные результаты:*

- сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов.

#### *Метапредметные результаты:*

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

#### *Предметные результаты:*

После изучения курса учащиеся должны:

- владеть навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов;
- владеть стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ на языке Python;
- знать место языка Python среди языков программирования высокого уровня;
- знать особенности структуры программы, представленной на языке Python;
- иметь представление о модулях, входящих в состав среды Python;
- знать возможности и ограничения использования готовых модулей;
- иметь представление о величине, ее характеристиках;
- знать что такое операция, операнд и их характеристики;
- знать принципиальные отличия величин структурированных и не структурированных;
- иметь представление о таких структурах данных, как число, текст, кортеж,

список, словарь;

- иметь представление о составе арифметического выражения;
- знать математические функции, входящие в Python;
- иметь представление о логических выражениях и входящих в них операндах, операциях и функциях;
- уметь записывать примеры арифметических и логических выражений всех атрибутов, которые могут в них входить;
- знать основные операторы языка Python, их синтаксис;
- иметь представление о процессе исполнения каждого из операторов;
- уметь разрабатывать программы обработки числовой и символьной информации;
- уметь разрабатывать программы (линейные, разветвляющиеся, циклические);
- иметь представление о значении полноценных процедур и функций для структурно-ориентированного языка высокого уровня;
- знать правила описания функций в Python и построение вызова;
- знать принципиальные отличия между формальными, локальными и глобальными переменными;
- знать область действия описаний в функциях;
- иметь представление о рекурсии, знать ее реализацию на Python;
- владеть основными приемами формирования процедуры и функции;
- знать свойства данных типа «массив», «матрица»;
- уметь воспроизводить алгоритмы сортировки одномерных и двумерных массивов, поиска в упорядоченном массиве, распространять эти алгоритмы на сортировку и поиск в нечисловых массивах;
- уметь читать и записывать текстовые файлы в заданном формате.

## **Планируемые результаты**

### *Первый уровень*

- научится составлять и отлаживать простые диалоговые программы;
- узнает особенностей машинных вычислений с целыми и вещественными числами;
- научится использовать основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной;
- овладеет методами построения графических изображений

программными средствами;

- овладеет простыми методами программирования компьютерной анимации.

### *Второй уровень*

- познакомится с методами проектирования программ «сверху вниз» и «снизу вверх»;
- научится использовать вспомогательные алгоритмы (процедуры и функции) для структуризации программ;
- научится применять рефакторинг для улучшения читаемости программ;
- научится использовать символьные строки;
- овладеет основными алгоритмами обработки одномерных и двумерных массивов;
- познакомится с понятием сложности алгоритма;

### *Третий уровень*

- научится применять различные алгоритмы сортировки массивов;
- научится использовать двоичный поиск;
- научится обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы;
- научится использовать структуры для объединения данных;
- научится применять словари, стеки, очереди, деки для решения задач обработки данных;
- научится использовать деревья для организации данных;
- познакомится с методами описания графов и некоторыми популярными алгоритмами на графах;
- научится использовать динамическое программирование для решения комбинаторных и оптимизационных задач;
- познакомится с понятием выигрышных и проигрышных позиций в играх с полной информацией;

### *Четвёртый уровень*

- познакомится с объектно-ориентированным подходом к разработке программ;
- научится выполнять объектно-ориентированный анализ задачи,

- выделять свойства и методы объектов;
- научиться использовать инкапсуляцию для защиты данных объектов;
  - познакомится с понятиями «класс» и «абстрактный класс»;
  - познакомится с понятиями «инкапсуляция», «наследование», «полиморфизм»;
  - научиться проектировать несложные иерархии классов для прикладных задач;
  - познакомится с принципами разработки событийно-ориентированных программ;
  - научиться создавать программы с графическим интерфейсом на языке Python;
  - научиться использовать готовые и создавать новые компоненты (виджеты) для сред быстрой разработки программ.

## **Содержание курса**

*Первый уровень (17 часов).*

1. Простейшие программы. Диалоговые программы. Переменные. Консольный ввод и вывод данных.
2. Компьютерная графика. Система координат. Управление пикселями. Графические примитивы: линии, прямоугольники, окружности. Изменение координат. Анимация.
3. Процедуры. Процедуры с параметрами. Рефакторинг.
4. Обработка целых чисел. Арифметические выражения. Деление нацело. Остаток от деления.
5. Обработка вещественных чисел. Особенности представления вещественных чисел в памяти компьютера. Операции с вещественными числами.
6. Случайные и псевдослучайные числа. Генераторы случайных чисел.
7. Ветвления. Условный оператор. Полная и неполная формы условного оператора. Вложенные условные операторы. Логические переменные. Экспертные системы. Сложные условия. Логические операции И, ИЛИ, НЕ. Порядок выполнения операций.
8. Циклы с условием. Алгоритм Евклида. Обработка потока данных. Бесконечные циклы. Циклы по переменной. Шаг изменения переменной цикла.

9. Циклы в компьютерной графике. Узоры. Вложенные циклы. Штриховка.

*Второй уровень (17 часов).*

1. Этапы создания программ. Методы проектирования программ «сверху вниз» и «снизу вверх». Интерфейс и реализация. Документирование программы.
2. Подпрограммы: процедуры и функции. Процедуры. Процедуры с параметрами. Локальные и глобальные переменные.
3. Функции. Логические функции.
4. Рекурсия. Рекурсивные процедуры и функции. Фракталы.
5. Символьные строки. Сравнение строк. Операции со строками. Обращение к символам. Перебор всех символов. Срезы. Удаление и вставка. Встроенные методы. Поиск в символьных строках. Замена символов. Преобразования «строка — число». Символьные строки в функциях. Рекурсивный перебор.
6. Массивы (списки). Массивы в языке Python. Создание массива. Обращение к элементу массива. Перебор элементов массива. Генераторы. Вывод массива. Ввод массива с клавиатуры. Заполнение массива случайными числами.
7. Алгоритмы обработки массивов. Сумма элементов массива. Подсчёт элементов массива, удовлетворяющих условию. Особенности копирования списков в Python.
8. Поиск в массивах. Линейный поиск. Поиск максимального элемента в массиве. Максимальный элемент, удовлетворяющий условию. Использование массивов в прикладных задачах.
9. Матриц. Создание и заполнение матриц. Вывод матрицы на экран. Перебор элементов матрицы. Квадратные матрицы.
10. Сложность алгоритмов. Асимптотическая сложность.

*Третий уровень (18 часов).*

1. Сортировка массивов. Метод пузырька (сортировка обментами). Метод выбора. Сортировка слиянием. Быстрая сортировка. Сортировка в языке Python.
2. Двоичный поиск в массиве данных. Двоичный поиск по ответу.
3. Обработка файлов. Типы файлов. Чтение данных. Запись данных. Обработка данных из файла.
4. Целочисленные алгоритмы. Решето Эратосфена. Целочисленный квадратный корень.
5. Словари. Алфавитно-частотный словарь. Перебор элементов словаря.

6. Структуры. Классы. Создание структур. Работа с полями структур. Хранение структур в файлах. Сортировка структур.
7. Стек. Использование списка. Вычисление арифметических выражений. Скобочные выражения. Системный стек. Очередь. Дек.
8. Деревья. Деревья поиска. Обход дерева. Использование связанных структур. Вычисление арифметических выражений.
9. Графы. Описание графа. Жадные алгоритмы. Минимальное остовное дерево. Алгоритм Дейкстры. Алгоритм Флойда–Уоршелла. Использование списков смежности.
10. Динамическое программирование. Числа Фибоначчи. Количество программ для исполнителя. Двумерные задачи. Поиск оптимального решения.
11. Игровые модели. Выигрышные и проигрышные позиции.

*Четвёртый уровень (18 часов).*

1. Проблема сложности программ. Процедурный и объектно-ориентированный подходы к написанию программ.
2. Классы и объекты. Объектно-ориентированный анализ. Взаимодействие объектов. Свойства и методы.
3. Классы и объекты в программе. Объявление класса. Поля класса. Конструктор класса. Данные и методы класса.
4. Скрытие внутреннего устройства. Доступ к полям через методы. Свойства (property). Свойство «только для чтения»
5. Иерархия классов. Наследование. Базовый класс. Доступ к полям. Классы-наследники. Полиморфизм. Разработка модулей.
6. Событийно-ориентированное программирование. Программы с графическим интерфейсом. Форма. Свойства формы. Обработчики событий.
7. Использование компонентов (виджетов). Ввод и вывод данных. Обработка ошибок с помощью исключений.
8. Создание компонентов. Добавление свойств и методов. Составные компоненты.
9. Модель и представление.

## **Основная литература для учащихся**

1. К.Ю. Поляков. Программирование. Python. C++ : учебное пособие: в 4 частях: Просвещение, 2023. (варианты глав по программированию для изучающих python <http://kpolyakov.spb.ru/school/probook/python.htm>)
2. задачник: <http://informatics.mccme.ru/course/view.php?id=666> .
3. Авторская программа Д. П. Кириенко. Программирование на python (школа 179 г.Москвы) <http://informatics.mccme.ru/course/view.php?id=15>

## **Дополнительная литература**

1. Марк Лутц. Изучаем Python. 4-е издание учебник. – М.:Символ-Плюс, 2011
2. Марк Саммерфилд. Программирование на Python 3. – М.: ДМК Пресс, 2014.
3. «Программирование: типовые задачи, алгоритмы, методы» / Д.М. Златопольский. – М.: БИНОМ. Лаборатория знаний, 2012.
4. Сборник задач по программированию» / Д. М. Златопольский. – СПб.: БХВ-Петербург, 2011.
5. «Задачи по программированию» / под ред. С. М. Окулова. – М.: БИНОМ. Лаборатория знаний, 2014.

## **Электронные образовательные ресурсы**

1. Сайт разработчика. <https://www.python.org/>(дата обращения 06.09.16)
2. Самоучитель для начинающих. python <https://pythonworld.ru>
3. Интерактивный учебник. <https://pythontutor.ru>

**Поурочное планирование курса  
10 класс. 34 часа**

Номер урока	Тема занятия	Кол-во часов	
		теория	практика
	<b>1 уровень</b>		
1.	Первые программы	0,5	0,5
2.	Диалоговые программы	0,5	0,5
3.	Компьютерная графика	0,5	0,5
4.	Процедуры	0,5	0,5
5.	Обработка целых чисел	0,5	0,5
6.	Обработка вещественных чисел	0,5	0,5
7.	Случайные и псевдослучайные числа	0,5	0,5
8.	Ветвления	0,5	0,5
9.	Сложные условия	0,5	0,5
10.	Циклы с условием	0,5	0,5
11.	Циклы с условием: практикум		1
12.	Анимация	0,5	0,5
13.	Циклы по переменной	0,5	0,5
14.	Циклы в компьютерной графике	0,5	0,5
15.	Выполнение проекта		1
16.	Выполнение проекта		1
17.	Выполнение проекта		1
	<b>2 уровень</b>		
18.	Проектирование программ	1	
19.	Процедуры	0,5	0,5
20.	Рекурсия	0,5	0,5
21.	Функции	0,5	0,5
22.	Символьные строки	0,5	0,5
23.	Обработка символьных строк	0,5	0,5
24.	Строки в функциях	0,5	0,5
25.	Массивы	0,5	0,5
26.	Ввод и вывод массивов	0,5	0,5
27.	Суммирование элементов массива	0,5	0,5
28.	Подсчёт элементов массива, удовлетворяющих условию	0,5	0,5
29.	Поиск значения в массиве	0,5	0,5
30.	Поиск максимального элемента в массиве	0,5	0,5
31.	Игра «Стрельба по тарелкам»	0,5	0,5
32.	Игра «Стрельба по тарелкам»	0,5	0,5
33.	Матрицы	0,5	0,5
34.	Сложность алгоритмов	1	
	<b>ИТОГО:</b>	<b>11,5</b>	<b>22,5</b>

### 11 класс. 34 часа

Номер урока	Тема занятия	Кол-во часов	
		теория	практика
	<b>3 уровень</b>		
1.	Простые алгоритмы сортировки	0,5	0,5
2.	Сортировка слиянием	0,5	0,5
3.	Быстрая сортировка	0,5	0,5
4.	Двоичный поиск	0,5	0,5
5.	Обработка файлов	0,5	0,5
6.	Обработка файлов: практикум		1
7.	Целочисленные алгоритмы	0,5	0,5
8.	Словари	0,5	0,5
9.	Структуры	0,5	0,5
10.	Структуры: практикум		1
11.	Стек, очередь, дек	0,5	0,5
12.	Деревья	0,5	0,5
13.	Графы	0,5	0,5
14.	Графы: практикум		1
15.	Динамическое программирование	0,5	0,5
16.	Динамическое программирование: практикум	0,5	0,5
17.	Игровые модели	0,5	0,5
18.	Игровые модели: практикум		1
	<b>4 уровень</b>		
19.	Что такое ООП?	1	
20.	Модель задачи: классы и объекты	0,5	0,5
21.	Классы и объекты в программе	0,5	0,5
22.	Классы и объекты в программе: практикум		1
23.	Скрытие внутреннего устройства	0,5	0,5
24.	Иерархия классов	0,5	0,5
25.	Классы-наследники (I)	0,5	0,5
26.	Классы-наследники (II)	0,5	0,5
27.	Доработка игры		1
28.	Событийно-ориентированное программирование	0,5	0,5
29.	Использование компонентов (виджетов)	0,5	0,5
30.	Использование компонентов (виджетов)	0,5	0,5
31.	Создание компонентов	0,5	0,5
32.	Модель и представление	0,5	0,5
33.	Выполнение проекта		1
34.	Выполнение проекта		1
	<b>ИТОГО:</b>	11,5	22,5