

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПРАВИТЕЛЬСТВО САНКТ-ПЕТЕРБУРГА

КОМИТЕТ ПО ОБРАЗОВАНИЮ

Государственное бюджетное общеобразовательное учреждение  
лицей № 369 имени Героя Российской Федерации А.Н.Жихарева  
Красносельского района Санкт-Петербурга  
(Лицей № 369)

**АННОТАЦИЯ**  
**К РАБОЧЕЙ ПРОГРАММЕ**  
**учебного курса «Практикум по программированию»**  
для обучающихся 10-11 классов

Санкт-Петербург  
2023

## **АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ УЧЕБНОГО КУРСА «ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ»**

Рабочая программа учебного курса «Практикум по программированию» (далее — рабочая программа) разработана в соответствии со следующей официальной правовой информацией:

- Федеральным законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказом Минобрнауки России от 17.05.2012 № 413 «Об утверждении федерального государственного образовательного стандарта среднего общего образования»;
- Приказа Минпросвещения России от 18.05.2023 № 371 «Об утверждении федеральной образовательной программы среднего общего образования»;
- Санитарными правилами и нормами СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания», утвержденными постановлением Главного государственного санитарного врача Российской Федерации от 28.01.2021 № 2;
- Санитарными правилами СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи», утвержденными постановлением Главного государственного санитарного врача Российской Федерации от 28.09.2020 № 28;
- образовательной программы среднего общего образования Лицея № 369.

Рабочая программа обеспечивает достижение планируемых результатов освоения образовательной программы среднего общего образования Лицея № 369.

Рабочая программа содержит:

- Пояснительную записку;
- Планируемые результаты освоения учебного курса;
- Содержание учебного курса;
- Тематическое планирование.

Рабочая программа является частью основной образовательной программы среднего общего образования Лицея № 369, принятой решением Педагогического совета Лицея № 369 (протокол от 29.08.2023 № 1), утвержденной приказом директора Лицея № 369 от 01.09.2023 № 75/21-од.

Выписка-извлечение из основной образовательной программы среднего общего образования Лицея № 369, принятой решением Педагогического совета Лицея № 369 (протокол от 29.08.2023 № 1), утвержденной приказом от 01.09.2023 № 75/21-од

**МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ПРАВИТЕЛЬСТВО САНКТ-ПЕТЕРБУРГА**

**КОМИТЕТ ПО ОБРАЗОВАНИЮ**

Государственное бюджетное общеобразовательное учреждение лицей № 369 имени Героя Российской Федерации А.Н.Жихарева Красносельского района Санкт-Петербурга  
(Лицей № 369)

**РАБОЧАЯ ПРОГРАММА**  
**учебного курса «Практикум по программированию»**  
для обучающихся 10-11 классов

Санкт-Петербург  
2023

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа учебного курса «Практикум по программированию» составлена на основании следующих нормативных правовых актов:

— Федеральный Закон «Об образовании в Российской Федерации» (от 29.12.2012 № 273-ФЗ).

— Приказ Минпросвещения России от 18.05.2023 № 371 «Об утверждении федеральной образовательной программы среднего общего образования».

### Общая характеристика изучаемого курса

Рабочая программа составлена по авторской программе Полякова К.Ю. и использует учебно-методический комплект по информатике для старшей школы (авторы К.Ю. Поляков, Е.А. Еремин; издательство «БИНОМ. Лаборатория знаний») и учебное пособие «Программирование. Python, C++» для 8–11 классов (автор К.Ю. Поляков). Учебное пособие «Программирование. Python, C++» содержит практические задания по всем темам курса.

В рамках этого курса учащиеся познакомятся с языком программирования Python в качестве второго языка в дополнение к языку программирования, изучаемому на уроках информатики. В современном мире активно используются десятки языков программирования, и их количество только растёт. Python — один из самых популярных языков программирования в последнее время. Эта популярность объясняется простым синтаксисом и компактностью программного кода. Python поддерживает различные подходы к программированию (императивный, объектно-ориентированный, функциональный). Он широко применяется в профессиональных разработках, то есть не является чисто учебным языком без перспектив применения в реальной жизни. Свободно распространяемые реализации языка Python существуют для всех популярных операционных систем. Знакомство с языком программирования Python осуществляется в той же методической последовательности, что и с основным языком в УМК.

К исходным требованиям, необходимым для изучения учебного курса «Практикум по программированию», относятся знания, умения и виды деятельности, сформированные в процессе изучения информатики на уровне основного общего образования. В системе школьного образования «Практикум по программированию» занимает место в качестве предмета по выбору и является средством предвузовской подготовки выпускников школы, мотивированных на дальнейшее обучение в системе высшего образования на IT-ориентированных специальностях (направлениях подготовки).

**Цель программы:** освоение системы базовых знаний, применяемых в большинстве языков программирования; формирование и углубленное развитие знаний, умений и навыков по языку программирования Python; применение полученных знаний на практике.

#### **Задачи:**

##### Обучающие:

— овладеть умениями строить математические объекты информатики, в том числе логические формулы и программы на формальном языке, удовлетворяющие заданному описанию; создавать программы на языке программирования по их описанию;

— развивать алгоритмическое мышление, способности к формализации, элементы системного мышления;

— систематизировать и углубить знания в области программирования;

— заложить основу для дальнейшего профессионального обучения.

##### Развивающие:

— развивать познавательный интерес, интеллектуальные и творческие способности путем освоения и использования методов информатики и средств ИКТ при изучении различных учебных предметов;

— сформировать навык использования информационных технологий в индивидуальной и коллективной учебной и познавательной, в том числе проектной деятельности;

— сформировать умение самостоятельно находить, изучать и анализировать информационные продукты;

Воспитательные:

— воспитать культуру работы с техническим оборудованием и программным обеспечением;

— воспитать ответственное отношение к соблюдению этических и правовых норм информационной деятельности;

— сформировать представление об информационном обществе и уровне развития науки;

— воспитать культуру информационной безопасности.

**Место изучаемого курса в учебном плане**

Учебный курс «Практикум по программированию» изучается в 10-11 классе, как предмет по выбору. На изучение отведено по 68 часов (2 часа в неделю) в каждом классе.

**ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОГО КУРСА  
«ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ»**

**ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ:**

1. Сформированность мировоззрения, соответствующего современному уровню развития науки и общественной практики.

2. Сформированность навыков сотрудничества со сверстниками, детьми младшего возраста, взрослыми в образовательной, общественно полезной, учебно-исследовательской, проектной и других видах деятельности.

3. Готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности; осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов.

4. Осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

**МЕТАПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ:**

1. Умение самостоятельно определять цели и составлять планы; самостоятельно осуществлять, контролировать и корректировать учебную и внеучебную (включая внешкольную) деятельность; использовать все возможные ресурсы для достижения целей; выбирать успешные стратегии в различных ситуациях.

2. Умение эффективно общаться в процессе совместной деятельности со всеми её участниками, не допускать конфликтов;

3. 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.

4. Готовность и способность к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации,

критически оценивать и интерпретировать информацию, получаемую из различных источников.

5. Владение навыками познавательной рефлексии как осознания совершаемых действий и мыслительных процессов, их результатов и оснований, границ своего знания и незнания, новых познавательных задач и средств их достижения.

## **ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ**

1. Овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки.

2. Владение универсальным языком программирования высокого уровня Python, представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции.

3. Владение навыками и опытом разработки программ в выбранной среде программирования, включая тестирование и отладку программ; владение элементарными навыками формализации прикладной задачи и документирования программ.

В результате изучения курса учащийся должен знать:

- особенности машинных вычислений с целыми и вещественными числами;
- основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной.

В результате изучения курса учащийся должен уметь:

- строить графические изображения программными средствами;
- составлять и отлаживать простые диалоговые программы;
- пользоваться простыми методами программирования компьютерной анимации.
- владеть методами проектирования программ «сверху вниз» и «снизу вверх»;
- использовать вспомогательные алгоритмы (процедуры и функции) для структуризации программ;
- применять рефакторинг для улучшения читаемости программ;
- использовать символьные строки;
- владеть основными алгоритмами обработки одномерных и двумерных массивов;
- применять различные алгоритмы сортировки массивов;
- использовать двоичный поиск;
- обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы;
- использовать структуры для объединения данных;
- применять словари, стеки, очереди, деки для решения задач обработки данных;
- использовать деревья для организации данных.

В результате изучения курса учащийся должен владеть:

- методами описания графов и некоторыми популярными алгоритмами на графах;
- методами использования динамического программирования для решения комбинаторных и оптимизационных задач;
- понятием выигрышных и проигрышных позиций в играх с полной информацией;
- объектно-ориентированным подходом к разработке программ;
- выполнением объектно-ориентированного анализа задачи, выделять свойства и методы объектов;
- методами использования инкапсуляции для защиты данных объектов;

- понятиями «класс» и «абстрактный класс»;
- понятиями «инкапсуляция», «наследование», «полиморфизм»;
- проектированием несложных иерархии классов для прикладных задач;
- принципами разработки событийно-ориентированных программ;
- созданием программы с графическим интерфейсом на языках Python;
- навыками использования готовых и создания новых компонентов (виджетов)

для сред быстрой разработки программ.

## **СОДЕРЖАНИЕ УЧЕБНОГО КУРСА «ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ»**

### **10 КЛАСС**

#### **Введение**

Алгоритм и программа. Этапы решения задач на компьютере. Этапы создания программ. Интерпретаторы и компиляторы. Загрузка среды программирования Python с сайта разработчика. Установка Python 3.4 в ОС Windows. Знакомство с интерфейсом среды программирования Python.

#### **Линейные алгоритмы**

Элементы языка программирования Python. Простая программа. Переменные. Типы данных. Размещение переменных в памяти. Консольный ввод и вывод данных. Диалоговые программы. Арифметические выражения и операции.

Вычисления. Деление нацело и остаток. Вещественные значения. Особенности представления вещественных чисел в памяти компьютера. Стандартные функции. Случайные и псевдослучайные числа. Генераторы случайных чисел.

#### **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.

#### **Циклические алгоритмы**

Цикл с условием. Алгоритм Евклида. Поиск максимальной цифры числа. Обработка потока данных. Бесконечные циклы. Циклы с постусловием. Циклы по переменной. Вложенные циклы.

#### **Компьютерная графика**

Система координат. Управление пикселями. Графические примитивы: линии, прямоугольники, окружности. Изменение координат. Циклы в компьютерной графике. Узоры. Вложенные циклы. Рефакторинг. Штриховка. Анимация. Анимация движения.

#### **Вспомогательные алгоритмы**

Методы проектирования программ «сверху вниз» и «снизу вверх». Интерфейс и реализация. Документирование программы. Подпрограммы: процедуры и функции. Процедуры. Процедуры с параметрами. Локальные и глобальные переменные. Вызов подпрограммы. Возврат нескольких значений. Функции. Логические функции.

Рекурсия. Ханойские башни. Анализ рекурсивных функций. Фракталы.

#### **Символьные строки**

Ввод и вывод строк. Операции со строками. Обращение к символам. Перебор всех символов. Срезы. Удаление и вставка. Встроенные методы. Поиск в символьных строках. Замена символов. Примеры обработки строк. Преобразование число-строка. Строки в процедурах и функциях. Рекурсивный перебор. Сравнение и сортировка строк.

#### **Массивы (списки)**

Массивы (списки). Создание массива. Обращение к элементу массива. Перебор элементов массива. Генераторы. Вывод массива. Ввод массива с клавиатуры. Заполнение массива случайными числами.

Алгоритмы обработки массивов. Сумма и произведение элементов массива. Подсчёт элементов массива, удовлетворяющих условию. Особенности копирования списков в Python.

Поиск в массиве. Линейный поиск. Поиск максимального элемента в массиве. Максимальный элемент, удовлетворяющий условию.

Реверс массива. Сдвиг элементов массива. Срезы массива. Отбор нужных элементов. Использование массивов в прикладных задачах.

Сложность алгоритмов. Асимптотическая сложность.

Сортировка массивов. Метод пузырька (сортировка обменами). Метод выбора. Сортировка слиянием. «Быстрая сортировка». Двоичный поиск. Сортировка в языке Python. Двоичный поиск в массиве данных. Двоичный поиск по ответу.

### **Матрицы**

Матрицы. Создание и заполнение матриц. Вывод матрицы на экран. Перебор элементов матрицы. Квадратные матрицы. Обработка элементов матрицы.

## **11 КЛАСС**

### **Работа с файлами**

Типы файлов. Чтение данных. Запись данных. Неизвестное количество данных. Обработка данных из файла. Обработка массивов. Обработка строк.

### **Целочисленные алгоритмы**

Целочисленные алгоритмы. Решето Эратосфена. «Длинные» числа. Целочисленный квадратный корень.

### **Абстрактные структуры данных**

Словари. Алфавитно-частотный словарь. Перебор элементов словаря.

Структуры. Классы. Создание структур. Работа с полями структур. Хранение структур в файлах. Сортировка структур.

Стек. Использование списка. Вычисление арифметических выражений. Скобочные выражения. Системный стек. Очередь. Дек.

Деревья. Деревья поиска. Обход дерева. Использование связанных структур. Вычисление арифметических выражений.

Графы. Описание графа. Жадные алгоритмы. Минимальное остовное дерево. Алгоритм Дейкстры. Алгоритм Флойда–Уоршелла. Использование списков смежности.

### **Динамическое программирование**

Числа Фибоначчи. Количество программ для исполнителя. Двумерные задачи. Поиск оптимального решения.

Игровые модели. Выигрышные и проигрышные позиции.

### **Объектно-ориентированное программирование**

Проблема сложности программ. Процедурный и объектно-ориентированный подходы к написанию программ. Объектный подход.

Классы и объекты. Объектно-ориентированный анализ. Взаимодействие объектов. Свойства и методы.

Классы и объекты в программе. Объявление класса. Поля класса. Конструктор класса. Конструктор с параметрами. Данные и методы класса.



Скрытие полей. Доступ к полям через методы. Свойства (*property*). Свойство «только для чтения».

Иерархия классов. Наследование. Базовый класс. Доступ к полям. Классы-наследники. Полиморфизм. Разработка модулей.

### **Событийно-ориентированное программирование**

Особенности современных прикладных программ. Программы с графическим интерфейсом. Форма. Свойства формы. Обработчики событий.

Использование компонентов (виджетов). Ввод и вывод данных. Обработка ошибок с помощью исключений.

Создание компонентов. Добавление свойств и методов. Составные компоненты.

Модель и представление. Вычисление арифметических выражений с помощью моделей.

Создание многооконных приложений.

## ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

### 10 КЛАСС

№ п/п	Наименование разделов и тем	Количество часов			Электронные (цифровые) образовательные ресурсы	Воспитательный потенциал
		Всего	Контрольные работы	Практические работы		
1.	Введение	1			<a href="https://resh.edu.ru/">https://resh.edu.ru/</a>	формирование стремления к получению знаний
2.	Линейные алгоритмы	6		3		формирование уважения к результатам труда (своего и других людей)
3.	Ветвления	8		3		
4.	Циклические алгоритмы	8		4		
5.	Компьютерная графика	5		4		
6.	Вспомогательные алгоритмы	8		3		
7.	Строки	10		5		
8.	Массивы (списки)	10		5		
9.	Матрицы	8		4		
Резервное время		4				
Общее количество часов		<b>68</b>		<b>31</b>		

### 11 КЛАСС

№ п/п	Тема	Количество часов			Электронные (цифровые) образовательные ресурсы	Воспитательный потенциал
		Всего	Контрольные работы	Практические работы		
1.	Повторение	2			<a href="https://resh.edu.ru/">https://resh.edu.ru/</a>	формирование стремления к получению знаний
2.	Работа с файлами	4		2		формирование уважения к результатам труда (своего и других людей)
3.	Целочисленные алгоритмы	3		1		
4.	Абстрактные структуры данных	18		8		
5.	Динамическое программирование	7		2		
6.	Объектно-ориентированное программирование	12		5		
7.	Событийно-ориентированное программирование	18		6		

Резервное время	4				
Общее количество часов	<b>68</b>		<b>24</b>		