

Государственное бюджетное общеобразовательное учреждение гимназия № 49  
Приморского района Санкт-Петербурга  
197082, Санкт-Петербург, Богатырский пр., д.55, корп.3, лит.А,  
Тел./Факс +7 (812) 246-1898, E-mail: [info.gim49@obr.gov.spb.ru](mailto:info.gim49@obr.gov.spb.ru)

**«ПРИНЯТА»**  
Педагогическим советом  
ГБОУ гимназии №49  
Протокол №1  
от «31» августа 2023 года

**«УТВЕРЖДЕНА»**  
Директор гимназии №49  
\_\_\_\_\_/Ф.Ф. Сёмочкина/  
Приказ №115 от «31» августа 2023 года

**«СОГЛАСОВАНА»**  
Руководитель отделения  
дополнительного образования  
\_\_\_\_\_/В.А Шпинок/  
«31» августа 2023 года

**Дополнительная общеобразовательная общеразвивающая программа  
Технической направленности  
Программирование Python  
на 2023-2024 учебный год  
год обучения: первый**

Автор-составитель: Чернышева Елена Евгеньевна,  
педагог дополнительного образования

Санкт-Петербург  
2023

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Зарница» разработана в соответствии с нормативно- правовыми документами:

- Федеральным законом «Об образовании в Российской Федерации» от 29.12.2012 года №273-ФЗ (далее – Федеральный закон № 273);

- Приказом Министерства образования и науки Российской Федерации от 29.08. 2013 года № 1008 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

- Концепцией развития дополнительного образования детей (Распоряжение Правительства РФ от 04.09.2014 года №1726-р);

- Постановлением Главного государственного санитарного врача РФ от 04.07.2014 года №41 «Об утверждении СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей»;

- Письмом Минобрнауки России № 09-3242 от 18.11.2015 года «Методические рекомендации по проектированию дополнительных общеобразовательных общеразвивающих программ»;

- Распоряжением Комитета по образованию Санкт-Петербурга от 01.03.2017 г. № 617-р «Об утверждении Методических рекомендаций по проектированию дополнительных общеразвивающих программ в государственных образовательных организациях Санкт-Петербурга, находящихся в ведении Комитета по образованию»;

- Уставом ГБОУ гимназия № 49 Приморского района Санкт-Петербурга.20. Устава ГБОУ гимназии № 49 Санкт-Петербурга;

**Направленность программы:** техническая.

Содержание программы предполагает освоение учащимися основ алгоритмизации и программирования, знание национальных приоритетных направлений развития информационных технологий, культурных, исторических особенностей области ИТ, воспитание гражданственности, чувства патриотизма и ответственности за развитие России.

Программа определяет основные пути развития системного подхода к процессам алгоритмизации и программирования.

**Уровень освоения программы:** углублённый.

**Актуальность.**

Достижение цифровой независимости, т. е. возможности сохранения безопасности в сфере ИТ-решений в рамках всей страны и опоры на собственные разработки, стало одним из важнейших векторов развития России с начала 2022 г. Решение практических задач связано с обучением подрастающего поколения современным, востребованным языкам программирования.

**Отличительные особенности программы:**

Занятия по алгоритмизации и программированию, направленные решение практических задач способствуют развитию у учащихся профессионального и творческого подхода к работе в ИТ сфере, готовность к выполнению сложных и ответственных задач в реальных жизненных ситуациях.

**Адресат программы:**

Программа ориентирована на учащихся 14 -17 лет без специальной подготовки.

**Сроки реализации программы:**

– Программа рассчитана на 288 часов за 4 года обучения (72 часа в год).

**Цель программы** – развитие познавательной деятельности учащихся в области новых информационных технологий; совершенствование навыков работы на компьютере и повышение интереса к программированию на языке Python: социальное становление подростков в процессе интеллектуального развития.

## Задачи

### **Обучающие:**

- анализировать поставленную задачу: определять входные данные и результаты, устанавливать связь между ними,
- выделять этапы решения задачи и т.д.;
- разрабатывать алгоритм решения;
- писать текст программы на языке Python;
- отлаживать и тестировать программу;
- готовить описание программы и инструкции для предполагаемых пользователей(документацию);
- дорабатывать и сопровождать программу после отладки и тестирования.

### **Развивающие:**

- формирование активной жизненной позиции,
- формирование морально-волевых качеств, в том числе и в преодолении трудностей,
- содействие удовлетворению интересов подростка в IT области;

### **Воспитательные:**

- воспитание подрастающего поколения на примерах творческого и профессионального подхода к решению актуальных задач в IT области;
- воспитание желания развитию информационных технологий в России,
- развитие инициативы, самостоятельности, навыков работы в команде.

### **Условия реализации программы**

Набор в тренировочные группы воспитанников проходит на добровольной основе, с согласия родителей (представителей), с обязательным допуском врача.

### **Наполняемость групп:**

- 1 год обучения – не менее 15 человек;
- 2 год обучения – не менее 12 человек;
- 3 год обучения – не менее 10 человек;
- 4 год обучения – не менее 10 человек;

### **Режим занятий:**

Занятия групповые. Занятия проводятся 1 раз в неделю по 2 часа.

### **Формы занятий:**

- практические занятия;
- лекции;
- групповая работа над проектами.

### **Формы организации деятельности учащихся на занятиях:**

- индивидуальная;
- групповая;
- индивидуально-групповая.

### **Материально-техническое обеспечение программы**

- Компьютерный класс.
- Интерпретатор Python 3 ([www.python.org](http://www.python.org))
- Среда разработки Wing IDE 101 ([wingware.com](http://wingware.com))
- Среда разработки PyScripter ([sourceforge.net/projects/pyscripter/](http://sourceforge.net/projects/pyscripter/))
- Среда разработки PyCharm ([www.jetbrains.com](http://www.jetbrains.com))

### **Планируемые результаты:**

#### **Личностные результаты**

У учащихся будут сформированы:

- Интерес к работе в сфере информационных технологий;
- потребность сотрудничества со сверстниками, доброжелательное отношение к сверстникам, бесконфликтное поведение, стремление прислушиваться к мнению одноклассников;

**Метапредметными результатами** изучения курса является формирование следующих универсальных учебных действий.

Регулятивные УУД:

Учащиеся научатся:

- понимать и принимать учебную задачу, сформулированную учителем;
- осуществлять контроль, коррекцию и оценку результатов своей деятельности.

Познавательная УУД:

Учащиеся научатся:

- проводить сравнение и классификацию объектов;
- понимать и применять полученную информацию при выполнении заданий;
- проявлять индивидуальные творческие способности.

Коммуникативные УУД:

Учащиеся научатся:

- работать в группе, учитывать мнение партнеров, отличные от собственных;
- обращаться за помощью;
- предлагать помощь и сотрудничество;
- слушать собеседника;
- формулировать свои затруднения;
- формулировать собственное мнение и позицию;
- договариваться и приходить к общему решению;
- осуществлять взаимный контроль.

## **Предметные результаты**

### **1 год обучения**

*В результате изучения курса на первом уровне учащийся*

- 1) научится составлять и отлаживать простые диалоговые программы;
- 2) узнает особенностей машинных вычислений с целыми и вещественными числами;
- 3) научится использовать основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной;
- 4) овладеет методами построения графических изображений программными средствами;
- 5) овладеет простыми методами программирования компьютерной анимации.

### **2 год обучения**

*В результате изучения курса на втором уровне учащийся*

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

### **3 год обучения**

- 1) научится применять различные алгоритмы сортировки массивов;
- 2) научится использовать двоичный поиск;
- 3) научится обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы;
- 4) научится использовать структуры для объединения данных;

- 5) научиться применять словари, стеки, очереди, деки для решения задач обработки данных;
- 6) научиться использовать деревья для организации данных;
- 7) познакомиться с методами описания графов и некоторыми популярными алгоритмами на графах;
- 8) научиться использовать динамическое программирование для решения комбинаторных и оптимизационных задач;
- 9) познакомиться с понятием выигрышных и проигрышных позиций в играх с полной информацией;

#### 4 год обучения

- 1) познакомиться с объектно-ориентированным подходом к разработке программ;
- 2) научиться выполнять объектно-ориентированный анализ задачи, выделять свойства и методы объектов;
- 3) научиться использовать инкапсуляцию для защиты данных объектов;
- 4) познакомиться с понятиями «класс» и «абстрактный класс»;
- 5) познакомиться с понятиями «инкапсуляция», «наследование», «полиморфизм»;
- 6) научиться проектировать несложные иерархии классов для прикладных задач;
- 7) познакомиться с принципами разработки событийно-ориентированных программ;
- 8) научиться создавать программы с графическим интерфейсом на языках Python и C#;
- 9) научиться использовать готовые и создавать новые компоненты (виджеты) для сред быстрой разработки программ.

#### Учебно-тематическое планирование 1 год обучения

№	Тема занятия	Кол-во часов		
		Теория	Практика	Всего
1.	Первые программы	2	2	<b>4</b>
2.	Диалоговые программы	2	2	<b>4</b>
3.	Компьютерная графика	2	4	<b>6</b>
4.	Процедуры	2	2	<b>4</b>
5.	Обработка целых чисел	2	2	<b>4</b>
6.	Обработка вещественных чисел	2	2	<b>4</b>
7.	Случайные и псевдослучайные числа	2	2	<b>4</b>
8.	Ветвления	2	4	<b>6</b>
9.	Сложные условия	2	4	<b>6</b>
10.	Циклы с условием	2	4	<b>6</b>
11.	Циклы с условием: практикум	2	4	<b>6</b>
12.	Анимация	2	4	<b>6</b>

13.	Циклы по переменной	2	4	6
14.	Циклы в компьютерной графике	2	4	6
<b>ВСЕГО</b>		<b>28</b>	<b>44</b>	<b>72</b>

### Содержание программы

Простейшие программы. Диалоговые программы. Переменные. Консольный ввод и вывод данных.

Компьютерная графика. Система координат. Управление пикселями. Графические примитивы: линии, прямоугольники, окружности. Изменение координат. Анимация.

Процедуры. Процедуры с параметрами. Рефакторинг.

Обработка целых чисел. Арифметические выражения. Деление нацело. Остаток от деления.

Обработка вещественных чисел. Особенности представления вещественных чисел в памяти компьютера. Операции с вещественными числами.

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.

Циклы с условием. Алгоритм Евклида. Обработка потока данных. Бесконечные циклы. Циклы по переменной. Шаг изменения переменной цикла.

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

### Учебно-тематическое планирование 2 год обучения

№	Тема урока	Кол-во часов		
		Теория	Практика	Всего
1	Проектирование программ	1	1	2
2.	Процедуры	2	2	4
3.	Рекурсия	2	4	6
4.	Функции	2	4	6
5.	Символьные строки	2	2	4
6.	Обработка символьных строк	1	2	3
7.	Строки в функциях	2	2	4
8.	Массивы	2	4	6
9.	Ввод и вывод массивов	2	4	6
10.	Суммирование элементов массива	2	4	6
11.	Подсчёт элементов массива, удовлетворяющих условию	2	4	6
12.	Поиск значения в массиве	2	2	4
13.	Поиск максимального элемента в массиве	2	2	4
14.	Игра «Стрельба по тарелкам»	1	2	3
15.	Матрицы	2	4	6
16.	Сложность алгоритмов	1	1	2
<b>ВСЕГО</b>		<b>28</b>	<b>44</b>	<b>72</b>

## Содержание программы

Этапы создания программ. 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.

Функции. Логические функции.

Рекурсия. Рекурсивные процедуры и функции. Фракталы.

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.

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.

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

Поиск в массивах. Линейный поиск. Поиск максимального элемента в массиве. Максимальный элемент, удовлетворяющий условию. Использование массивов в прикладных задачах.

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

Сложность алгоритмов. Асимптотическая сложность.

## Учебно-тематическое планирование 3 год обучения

№	Тема урока	Кол-во часов		
		Теория	Практика	Всего
1.	Простые алгоритмы сортировки	1	1	2
2.	Сортировка слиянием	2	2	4
3.	Быстрая сортировка	2	4	6
4.	Двоичный поиск	2	4	6
5.	Обработка файлов	2	2	4
6.	Обработка файлов: практикум	1	2	3
7.	Целочисленные алгоритмы	2	2	4
8.	Словари	2	4	6
9.	Структуры	2	4	6
10.	Структуры: практикум	2	4	6
11.	Стек, очередь, дек	2	4	6
12.	Деревья	2	2	4
13.	Графы	2	2	4
14.	Графы: практикум	1	2	3
15.	Динамическое программирование	2	4	6
16.	Игровые модели	1	1	2
<b>ВСЕГО</b>		<b>28</b>	<b>44</b>	<b>72</b>

### Содержание программы

Сортировка массивов. Метод пузырька (сортировка обменами). Метод выбора. Сортировка слиянием. Быстрая сортировка. Сортировка в языке Python.

Двоичный поиск в массиве данных. Двоичный поиск по ответу.

Обработка файлов. Типы файлов. Чтение и запись данных. Обработка данных из файла.

Целочисленные алгоритмы. Решето Эратосфена. Целочисленный квадратный корень.

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. 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.

### Учебно-тематическое планирование

#### 4 год обучения

№	Тема урока	Кол-во часов		
		Теория	Практика	Всего
1.	Что такое ООП?	1	1	2
2.	Модель задачи: классы и объекты	2	2	4
3.	Классы и объекты в программе	2	4	6
4.	Классы и объекты в программе: практикум	2	4	6
5.	Скрытие внутреннего устройства	2	2	4
6.	Иерархия классов	1	2	3
7.	Классы-наследники (I)	2	2	4
8.	Классы-наследники (II)	2	4	6
9.	Доработка игры	2	4	6
10.	Событийно-ориентированное программирование	2	4	6
11.	Использование компонентов (виджетов)	2	6	8
12.	Создание компонентов	2	2	4
13.	Модель и представление	1	2	3
14.	Выполнение проекта	2	8	10
<b>ВСЕГО</b>		<b>25</b>	<b>47</b>	<b>72</b>

## Содержание программы

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

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

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

Скрытие внутреннего устройства. 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. 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.

## Календарный учебный график

Год обучения	Дата начала обучения	Дата окончания обучения	Всего учебных недель	Количество учебных часов	Режим занятий
1 год	01.09	25.05	36	72	1 занятие 2 часа в неделю
1 год	01.09	25.05	36	72	1 занятие 2 часа в неделю
1 год	01.09	25.05	36	72	1 занятие 2 часа в неделю
1 год	01.09	25.05	36	72	1 занятие 2 часа в неделю

### Оценочные и методические материалы

- словесные (лекции, беседы, работа по устным рекомендациям, анализ проделанной работы);
- наглядные (демонстрация, показ приемов работы, работа по заданиям, просмотр видеоматериалов);
- игровые;
- практические (упражнения);
- решение проблемных задач;
- использования ИКТ (просмотр видеоматериалов, поиск информации в Интернет)

#### Способы проверки результатов:

- опрос ;
- тестирование;

Тесты и опрос проводятся в течение учебного года. Итоги подводятся в заключительном занятии.

#### Педагогическая диагностика предполагает:

- личные беседы с детьми и их родителями;
- наблюдение за учащимися во время тренировочных и соревновательных мероприятий;

#### Формы подведения итогов реализации программы:

- проведение итогового занятия – защита проекта;
- участие в олимпиадах.

Занятия по данной программе состоят из теоретической и практической частей, причем большее количество времени занимает практическая часть, в процессе которой, в основном происходит освоение программного материала. Каждое учебное занятие является звеном системы занятий, связанных в логическую последовательность, построенных друг за другом. Важнейшим требованием современного учебного занятия является обеспечение дифференцированного и индивидуального подхода к обучающимся, с учетом их состояния здоровья, физического развития, особенностей развития психических свойств и качеств.

Содержание программы основывается на следующих основных педагогических принципах образования: демократизации, гуманизации, детоцентризма, природосообразности, культуросообразности, педагогики сотрудничества, дифференциации и индивидуализации.

В зависимости от поставленных педагогических задач, занятия могут быть:

- вводное (введение в предмет, постановка учебных цели и задач, определение средств и методов достижения цели, инструктаж по ТБ)
- занятия – изучение нового материала;
- занятия – повторение, направленные на закрепление, совершенствование ранее освоенных навыков;

- комбинированное (сочетающее изучение, повторение, совершенствование ранее освоенных упражнений) – имеют наибольшее распространение в процессе обучения, - контрольное занятие - проводится после прохождения части программного материала;
- соревнование, поход, экскурсия и т.д.

Основные принципы построения учебного занятия:

- постепенность в развитии природных данных обучающихся;
- строгая последовательность в изучении и овладении технологических приемов;
- систематичность, регулярность занятий;
- целенаправленность учебного процесса;
- проявление педагогической гибкости по отношению к обучающимся;
- принцип эмоционально-психологической комфортности (создание образовательной среды, обеспечивающей снятие, по возможности, всех стрессообразующих факторов учебного процесса);
- принцип деятельности (новое знание вводится не в готовом виде, а через самостоятельное открытие);
- принцип интеграции разных видов деятельности;
- принцип взаимосвязи и взаимопроникновения программных разделов;
- принцип совместной деятельности педагога, обучающегося и родителей;
- принцип учета индивидуальных особенностей обучающихся.

Основные методы, используемые в учебно-воспитательном процессе.

1. Демонстрационные:

- показ;
- пример;
- видеоиллюстрация.

2. Вербальные:

- объяснение;
- беседа;
- рассказ;
- анализ;
- инструктаж.

3. Практические:

- упражнение;
- игра;
- взаимоконтроль;
- самоконтроль;

4. Стимулирующие:

- соревнование;
- поощрение,

### **Информационные источники**

**Для учителя и ученика**

1. Поляков К. Учебное пособие «Программирование. Python, C++» 8-11 классы. Профильная школа. 2019г.
2. Добряк П.В. «Python. 12 уроков для начинающих». БХВ, 2023 г.
3. Материалы сайта: <https://pythonworld.ru/>.

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ

**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ГИМНАЗИЯ № 49 ПРИМОРСКОГО РАЙОНА  
САНКТ-ПЕТЕРБУРГА**, Семочкина Фарида Фаридовна, Директор

17.10.23 17:04 (MSK)

Сертификат 3A57D79CE6AE4E1B1876D75D6657705C