

**Государственное бюджетное общеобразовательное учреждение  
лицей № 373 Московского района Санкт-Петербурга «Экономический лицей»**

**ПРИНЯТА**  
решением Педагогического совета  
Государственного бюджетного  
общеобразовательного учреждения  
лицей № 373 Московского района  
Санкт-Петербурга «Экономический  
лицей», протокол от 30.08.2023 г. № 1

**УТВЕРЖДЕНА**  
приказом по Государственному бюджетному  
общеобразовательному учреждению лицей  
№ 373 Московского района Санкт-Петербурга  
«Экономический лицей» от 30.08.2023 № 120-од



**РАБОЧАЯ ПРОГРАММА**  
**курса внеурочной деятельности**  
**«Программирование на языке Python»**  
**10 класс**

**Учитель составитель:**  
Ляпустина Ольга Юрьевна,  
учитель информатики ГБОУ лицей №373  
Московского района Санкт-Петербурга

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа по курсу внеурочной деятельности «Программирование на языке Python» разработана на основе:

- Федерального государственного образовательного стандарта среднего общего образования, утвержденного приказом Министерства образования и науки Российской Федерации от 17.05.12 № 413 (ред. 12.08.2022);
- Федеральной образовательной программы среднего общего образования, утвержденной приказом Министерства просвещения Российской Федерации от 18.05.2023 № 371
- Положения «О рабочей программе учебного предмета, курса государственного бюджетного общеобразовательного учреждения лицей №373 Московского района Санкт-Петербурга «Экономический лицей».

Настоящая рабочая программа является составной частью основной образовательной программы среднего общего образования ГБОУ лицей №373 Московского района Санкт-Петербурга (содержательный раздел).

Программа курса внеурочной деятельности «Программирование на языке Python» предназначена для обучающихся 10 класса и предназначен для развития навыков алгоритмического мышления.

Язык Python – один из самых востребованных на рынке труда. Изучение Python в школе откроет ученикам возможности дальнейшего развития в области IT и поможет профориентации в старших классах, пригодится в олимпиадах по программированию и решению заданий ЕГЭ.

Курс предполагает смешанный формат обучения. Сочетание групповой работы с учителем в классе и индивидуальной работы в личном кабинете на онлайн-платформе Uchi Дома позволяет ученикам выработать не только технические навыки программирования, но и навыки социального взаимодействия при работе над проектами курса, а главное – научиться самостоятельно выстраивать свое профессиональное развитие.

Курс последовательно и целенаправленно вовлекает учащегося в процесс самостоятельного и осмысленного составления законченных алгоритмов и программ, вырабатывает необходимые составляющие алгоритмической и программистской грамотности: ясный и понятный стиль, надежность и эффективность решений, умение организовать переборы и ветвления.

Курс внеурочной деятельности рассчитан на 136 часов (4 ч в неделю) является дополнением основных уроков информатики в школе, он позволяет систематизировать и углубить знания по программированию.

**Цель курса:** формирование свободного и творческого подхода к программированию на языке Python

**Направленность программы:** техническая.

**Задачи курса:**

- познакомить учеников с основами программирования на языке Python
- научить принципам программирования и применение их при решении алгоритмических задач и создании графических проектов на языке программирования Python с использованием вспомогательных библиотек;
- приобщить обучающихся к новым технологиям, способным помочь им в реализации собственного творческого потенциала;
- совершенствовать у учащихся навыки работы на компьютере и повысить интерес к программированию;

- обучить принципам создания творческих проектов с помощью графической библиотеки в среде Python;
- научиться применять полученные знания для решения практических задач.

## **ЛИЧНОСТНЫЕ, МЕТАПРЕДМЕТНЫЕ И ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ КУРСА**

### **Результаты обучения:**

#### **Личностные:**

- Понимание значения информатики как науки в жизни современного общества .
- Самостоятельно реализовывать проекты, связанные с разработкой игр.
- Наличие базовых навыков самостоятельной работы с учебными текстами, справочной литературой, разнообразными средствами информационных технологий
- Самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности.
- Планировать свою самостоятельную учебно-познавательную деятельность; выбирать индивидуальную траекторию достижения учебной цели.

#### **Метапредметные:**

- Ставить учебные цели.
- Формулировать достигнутый результат.
- Умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать.
- Самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логические рассуждения, делать умозаключения (индуктивные, дедуктивные и по аналогии) и выводы.
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- самостоятельно выбирать способ решения учебной задачи (сравнивать несколько вариантов решения, выбирать наиболее подходящий с учётом самостоятельно выделенных критериев).
- Определять подходы и методы для достижения поставленной цели.
- Отбирать необходимые средства для достижения поставленной цели.
- Осуществлять самооценку промежуточных и итоговых результатов своей самостоятельной учебно-познавательной деятельности.
- Проводить рефлексию своей учебно-познавательной деятельности.

#### **Предметные:**

- Программировать на языке Python
- Практическими результатами каждого ученика служат самостоятельно разработанные мини-проекты в консольном режиме языка Python.

### **К концу обучения в 10 классе обучающийся будет знать/понимать:**

- учебную среду IDLE для составления программ
- основные термины программирования «алгоритм», «программа», «язык программирования», «функция», «рекурсия»;
- отличия основных типов данных;
- вещественные числа. Основы работы с вещественными числами.

- функции ввода и вывода данных, функции перехода от одного типа данных к другому;
- операторы continue и др.;
- 3 типа алгоритмов: линейный, ветвление, цикл с условием;
- условный оператор. Вложенный условный оператор
- цикл While
- функция RANGE, цикл FOR.
- понимать различия локальных и глобальных переменных;
- решать задачи с использованием глобальных переменных на Python;
- методы. Словари. Кортежи. Списки. Строки;
- методы работы со списками;
- что такое срезы строк, использование срезов
- итераторы и генераторы;
- метод def
- чтение из файла. Сортировка подсчетом.
- лямбда-функции.
- создание и работа с множествами
- генерация комбинаторных объектов ITERTOOLS
- функции PARTIAL, REDUCE, ACCUMULATE
- классы объектов. Обработка ошибок. Наследование и полиморфизм
- работу с модулем Turtle.

#### **Уметь:**

- записывать алгоритмические конструкции на языке программирования;
- решать задачи с условным оператором и циклами for, while.
- составлять алгоритмы для решения учебных задач различных типов;
- выражать алгоритм решения задачи различными способами (словесным, графическим, в том числе и в виде блок-схемы, с помощью формальных языков и др.);
- определять результат выполнения заданного алгоритма (программы) или его фрагмента;
- использовать термины «алгоритм», «программа», а также понимать разницу между употреблением этих терминов в обыденной речи и в информатике;
- выполнять без использования компьютера («вручную») несложные алгоритмы и анализа числовых и текстовых данных, записанные на язык программирования с использованием основных управляющих конструкций последовательного программирования (линейная программа, ветвление, повторение, вспомогательные алгоритмы);
- составлять алгоритмы управления исполнителями и анализа числовых и текстовых данных с использованием основных управляющих конструкций последовательного программирования и записывать их в виде программ на языке программирования; выполнять эти программы на компьютере;
- использовать величины (переменные) различных типов, табличные величины (массивы), а также выражения, составленные из этих величин; использовать оператор присваивания;
- анализировать предложенный алгоритм, например, определять какие результаты возможны при заданном множестве исходных значений;
- использовать логические значения, операции и выражения с ними;

- записывать на выбранном языке программирования арифметические и логические выражения и вычислять их значения.
- использовать в программах строковые величины и уметь выполнять операции со строковыми величинами;
- использовать события при написании программ на Python; искать ошибки в программном коде на Python и исправлять их;
- дописывать программный код на Python;
- писать программный код на Python;
- писать свои функции на Python;
- разбивать задачи на подзадачи;
- анализировать блок-схемы и программы на Python .

## **СОДЕРЖАНИЕ КУРСА**

### **Модуль 1. Основы программирования на языке Python**

Вводное занятие. Алгоритмизация. История языков программирования. Введение в язык программирования Python. Среда программирования Python. Типы данных и функции вывода. Определение переменной. Переменные и арифметические выражения. Чтение данных. Операции над строками. Примеры решения задач. Отработка навыков решения простейших задач. Логический тип данных и операции. Примеры использования логических выражений. Условный оператор. Вложенный условный оператор. Примеры решения задач. Цикл While. Подсчет суммы и оператор CONTINUE. Арифметические задачи с циклом While. Вещественные числа. Основы работы с вещественными числами. Вещественные числа и цикл While. Мини-проект «Калькулятор». Тестирование по итогам 1 Модуля

### **Модуль 2. Методы программирования на языке Python: find. Rfind. Replace и count. Split и join.**

Срезы строк. Использование срезов. Метод FIND. Примеры решения задач. Отработка навыков решения простейших задач со строками. Отработка навыков решения простейших задач с методом FIND. Продолжение. Методы RFIND, REPLACE и COUNT. Решение несложных задач. RFIND. REPLACE. Решение несложных задач. COUNT. Функции. Использование функций. Локальные и глобальные переменные. Примеры решения задач. Функции. Рекурсия. Использование рекурсии. Примеры решения задач. Отработка навыков решения задач. Свой мини-проект. Кортежи. Функция RANGE, цикл FOR. Кортежи. Функция RANGE, цикл FOR. Примеры решения задач. Свой мини-проект с циклом FOR. Списки. Метод SPLIT и JOIN. Тестирование по итогам 2 Модуля

### **Модуль 3. Методы программирования на языке Python. Словари. Кортежи. Списки. Функции**

Отработка навыков решения задач. Метод JOIN. Основные конструкции языка программирования Python. Типы данных. Арифметические выражения. Условный оператор. Циклы. Решение задач. Срезы строк. Кортежи. Методы. Функции и рекурсия в Python. Списки. Методы работы со списками. Собственный мини-проект «Библиотека». Сортировка. Сравнение списков и кортежей. Именованный параметр KEY. Примеры решения задач. Отработка навыков решения задач. Списки и кортежи. Именованные параметры. Чтение из файла. Сортировка подсчетом. Сортировка подсчетом. Отработка навыков решения задач. Лямбда-функции. Отработка навыков решения задач. Множества и хеш-функции. Создание и работа с множествами. Отработка навыков решения задач. Проект мини-бот. Словари. Методы строк. Тестирование по итогам 3 Модуля

#### **Модуль 4. Объектно-ориентированное программирование. Решение задач повышенной сложности**

Парадигмы программирования и функциональное программирование. Примеры решения задач в функциональном стиле. Встроенные функции для работы с последовательностями. Примеры решения задач в функциональном стиле. Создание собственных функций. Метод def. Примеры решения сложных задач. Решение олимпиадных задач в функциональном стиле. Итераторы и генераторы. Генерация комбинаторных объектов ITERTOOLS. Функции PARTIAL, REDUCE, ACCUMULATE. Примеры решения задач. Отработка навыков решения сложных задач. Генерация комбинаторных объектов ITERTOOLS. Функции PARTIAL, REDUCE, ACCUMULATE. Объектно-ориентированное программирование. Модуль Turtle. Классы объектов. Обработка ошибок. Наследование и полиморфизм. Примеры решения задач. Работа с модулем Turtle. Циклы. Условия. Графика. Проект «Черепашьи гонки». Тестирование по итогам 4 Модуля

### **ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ**

<b>№ п/п</b>	<b>Название темы</b>	<b>Всего часов</b>
1.	Основы программирования на языке Python	36
2.	Методы программирования на языке Python: find, Rfind, Replace и count. Split и join.	35
3.	Методы программирования на языке Python. Словари. Кортежи. Списки. Функции	35
4.	Объектно-ориентированное программирование. Решение задач повышенной сложности	30
<b>Итого:</b>		<b>136</b>

