Муниципальное общеобразовательное автономное учреждение «Лицей № 3»

|  |  |  |
| --- | --- | --- |
|  |  | УТВЕРЖДЕНО  Директор МОАУ «Лицей № 3»  \_\_\_\_\_\_\_\_\_\_\_\_\_Попуца Е.А.  Приказ № 1  от «31» августа 2023 г. |

РАБОЧАЯ ПРОГРАММА

учебного курса

«Программирование на альтернативных языках»

по информатике и ИКТ

10-11 классы

профильный уровень

на 2023-2024 уч.год

г. Оренбург

[**1. Планируемые результаты освоения обучающимися элективного курса** 3](#_Toc48006390)

[1.1. Планируемые метапредметные результаты освоения элективного курса 3](#_Toc48006391)

[1.2. Планируемые личностные результаты освоения элективного курса 3](#_Toc48006392)

[1.3. Планируемые предметные результаты освоения элективного курса 3](#_Toc48006393)

[**2. Содержание элективного курса** 5](#_Toc48006394)

[**3. Тематическое планирование с указанием количества часов, отведенных на освоение каждой темы** 6](#_Toc48006395)

# **1. Планируемые результаты освоения обучающимися учебного курса**

В результате изучения дисциплины получают дальнейшее развитие личностные регулятивные, коммуникативные и познавательно-универсальные учебные действия, учебная (общая и предметная) и общепользовательская ИКТ–компетентность обучающихся, составляющая психолого-педагогическую, инструментальную основу формирования способности и готовности к освоению систематических знаний, к их самостоятельному пополнению, переносу и интеграции, способности к сотрудничеству и коммуникации, решению личностно и социально значимых проблем и воплощению решений в практику, способности к самоорганизации, саморегуляции и рефлексии.

## 1.1. Планируемые метапредметные результаты освоения учебного курса

В ходе изучения данного курса в основном формируются и получают развитие следующие метапредметные результаты:

* умение самостоятельно планировать пути достижения цели, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения задач;
* умение соотносить свои действия с планируемыми результатами, осуществлять контроль всей деятельности в процессе достижения результата, корректировать свои действия в соответствии с изменяющейся ситуацией;
* умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения;
* владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
* умение организовывать сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально в группе: находить общее решение и разрешать конфликты на основе согласовании позиций и учета интересов; формулировать, аргументировать и отстаивать свое мнение;
* формирование и развитие компетентности в области использования информационно-коммуникационных технологий.

## 1.2. Планируемые личностные результаты освоения учебного курса

Вместе с тем, вноситься существенный вклад в развитие личностных результатов:

* формирование ответственного отношения к обучению, готовности и способности обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию;
* формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, взрослыми в процессе образовательной, общественно-полезной, учебно-исследовательской, творческой и других видах деятельности.

## 1.3. Планируемые предметные результаты освоения учебного курса

В части развития предметных результатов наибольшее влияние изучение курса оказывает на:

* формирование знаний, умений и навыков при решении задач информатики и программирования разных видов;
* приобретение опыта использования информационных технологий в индивидуальной и коллективной учебной и познавательной, в том числе проектной деятельности;
* формирование информационной и алгоритмической культуры; формирование представления о компьютере как универсальном устройстве обработки информации;
* развитие основных навыков и умений использования компьютерных устройстве;
* формирование представления об основных изучаемых понятиях: информация, алгоритм, модель и их свойства;
* развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм для конкретного исполнителя;
* формирование знаний об алгоритмических конструкциях, логических значениях и операциях;
* знакомство с одним из языков программирования и основными алгоритмическими структурами – линейной, условной и циклической;
* владение системой базовых знаний, отражающих вклад информатики в формирование современной научной картины мира;
* овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки;
* владение универсальным языком программирования высокого уровня, представлениями о базовых типах данных и структурах данных;
* умением использовать основные управляющие конструкции;
* владение навыками и опытом разработки программ в среде программирования, включая тестирование и отладку программ; владение элементарными навыками формализации прикладной задачи и документирования программ;
* формирование умения работать с библиотеками программ;
* получение опыта использования компьютерных средств представления и анализа данных.

*В результате изучения дисциплины учащиеся должны знать:*

* основные типы алгоритмов;
* иметь представление о структуре программы, основы программирования на языках высокого уровня;
* базовые алгоритмические конструкции;
* содержание этапов разработки программы: алгоритмизация-кодирование-отладка-тестирование;
* дополнительные возможности языка программирования для выражения различных алгоритмических ситуаций;
* алгоритмы и программы решения простых, сложных и нестандартных задач в математической области;
* основы разработки простых игр в системе программирования.

*Учащиеся должны уметь:*

* записывать алгоритмические структуры на языке программирования;
* использовать языка программирования для решения задач из области математики, физики;
* строить алгоритмы методом последовательного уточнения (сверху вниз), изображать эти алгоритмы в виде блок-схем;
* использовать основные алгоритмические приемы при решении математических задач;
* решать простые, сложные и нестандартные задачи;
* создавать простые игры;
* анализировать текст чужих программ, находить в них неточности, оптимизировать алгоритм, создавать собственные варианты решения.

# **2. Содержание учебного курса**

**Основы программирования на языке Python.**

История языков программирования. Введение в язык программирования Python. Среда программирования Python. Установка программы. Типы данных и функции вывода. Определение переменной. Переменные и арифметические выражения. Чтение данных. Операции над строками. Логический тип данных и операции. Примеры использования логических выражений. Условный оператор. Вложенный условный оператор. Цикл WHILE. Подсчет суммы и оператор CONTINUE. Вещественные числа. Основы работы с вещественными числами. Округление вещественных чисел.

**Методы программирования на языке Python. Введение в олимпиадное программирование.**

Срезы строк. Использование срезов. Метод FIND. Методы RFIND, REPLACE и COUNT. Решение несложных олимпиадных задач. Функции. Использование функций. Возврат значений. Локальные и глобальные переменные. Рекурсия. Использование рекурсии. Кортежи. Функция RANGE, цикл FOR. Списки. Метод SPLIT и JOIN. Решение олимпиадных задач.

**Методы программирования на языке Python. Решение олимпиадных задач.**

Сортировка. Сравнение списков и кортежей. Именованный параметр KEY. Структуры в Python. Лямбда-функции. Именованные параметры. Чтение из файла. Сортировка подсчетом. Множества и хеш-функции. Создание и работа с множествами. Решение олимпиадных задач. Словари. Методы строк.

**Объектно-ориентированное программирование. Решение задач повышенной сложности.**

Парадигмы программирования и функциональное программирование. Встроенные функции для работы с последовательностями. Итераторы и генераторы. Генерация комбинаторных объектов ITERTOOLS. Функции PARTIAL, REDUCE, ACCUMULATE. Объектно-ориентированное программирование. Комплексные числа. Инкапсуляция и конструкторы. Классы объектов. Обработка ошибок. Наследование и полиморфизм. Решение задач повышенной сложности. Решение сложных олимпиадных задач.

# **3.** **Тематическое планирование с указанием количества часов, отведенных на освоение каждой темы**

**10 класс**

|  |  |  |
| --- | --- | --- |
| **№** | **Тема урока** | **Количество часов** |
| Основы программирования на языке Python (14 часов) | | |
| 1 | История языков программирования. Введение в язык программирования Python. Среда программирования Python. Установка программы. | 1 |
| 2 | Типы данных и функции вывода. Определение переменной. Переменные и арифметические выражения. | 1 |
| 3 | Типы данных и функции вывода. Определение переменной. Переменные и арифметические выражения. | 1 |
| 4 | Чтение данных. Операции над строками. | 1 |
| 5 | Чтение данных. Операции над строками. | 1 |
| 6 | Логический тип данных и операции. Примеры использования логических выражений. | 1 |
| 7 | Условный оператор. Вложенный условный оператор. | 1 |
| 8 | Условный оператор. Вложенный условный оператор. | 1 |
| 9 | Цикл WHILE. | 1 |
| 10 | Цикл WHILE. | 1 |
| 11 | Подсчет суммы и оператор CONTINUE. | 1 |
| 12 | Подсчет суммы и оператор CONTINUE. | 1 |
| 13 | Вещественные числа. Основы работы с вещественными числами. Округление вещественных чисел. | 1 |
| 14 | Вещественные числа. Основы работы с вещественными числами. Округление вещественных чисел. | 1 |
| Методы программирования на языке Python. Введение в олимпиадное программирование (20 часов) | | |
| 15 | Срезы строк. Использование срезов. Метод FIND. | 1 |
| 16 | Срезы строк. Использование срезов. Метод FIND. | 1 |
| 17 | Методы RFIND, REPLACE и COUNT. | 1 |
| 18 | Методы RFIND, REPLACE и COUNT. | 1 |
| 19 | Решение несложных олимпиадных задач. | 1 |
| 20 | Решение несложных олимпиадных задач. | 1 |
| 21 | Функции. Использование функций. | 1 |
| 22 | Функции. Использование функций. | 1 |
| 23 | Возврат значений. Локальные и глобальные переменные. | 1 |
| 24 | Возврат значений. Локальные и глобальные переменные. | 1 |
| 25 | Рекурсия. Использование рекурсии. | 1 |
| 26 | Рекурсия. Использование рекурсии. | 1 |
| 27 | Кортежи. Функция RANGE, цикл FOR. | 1 |
| 28 | Кортежи. Функция RANGE, цикл FOR. | 1 |
| 29 | Списки. Метод SPLIT и JOIN. | 1 |
| 30 | Списки. Метод SPLIT и JOIN. | 1 |
| 31 | Решение олимпиадных задач. | 1 |
| 32 | Решение олимпиадных задач. | 1 |
| 33 | Решение олимпиадных задач. | 1 |
| 34 | Решение олимпиадных задач. | 1 |

**11 класс**

|  |  |  |
| --- | --- | --- |
| **№** | **Тема урока** | **Количество часов** |
| Методы программирования на языке Python. Решение олимпиадных задач (15 часов) | | |
| 1 | Сортировка. Сравнение списков и кортежей. Именованный параметр KEY. | 1 |
| 2 | Сортировка. Сравнение списков и кортежей. Именованный параметр KEY. | 1 |
| 3 | Структуры в Python. Лямбдафункции. Именованные параметры. Чтение из файла. | 1 |
| 4 | Структуры в Python. Лямбдафункции. Именованные параметры. Чтение из файла. | 1 |
| 5 | Сортировка подсчетом. | 1 |
| 6 | Сортировка подсчетом. | 1 |
| 7 | Множества и хеш-функции. Создание и работа с множествами. | 1 |
| 8 | Множества и хеш-функции. Создание и работа с множествами. | 1 |
| 9 | Решение олимпиадных задач. | 1 |
| 10 | Решение олимпиадных задач. | 1 |
| 11 | Решение олимпиадных задач. | 1 |
| 12 | Решение олимпиадных задач. | 1 |
| 13 | Словари. Методы строк. | 1 |
| 14 | Словари. Методы строк. | 1 |
| 15 | Словари. Методы строк. | 1 |
| Объектно-ориентированное программирование. Решение задач повышенной сложности (19 часов) | | |
| 16 | Парадигмы программирования и функциональное программирование. Встроенные функции для работы с последовательностями. | 1 |
| 17 | Парадигмы программирования и функциональное программирование. Встроенные функции для работы с последовательностями. | 1 |
| 18 | Итераторы и генераторы. Генерация комбинаторных объектов ITERTOOLS. Функции PARTIAL, REDUCE, ACCUMULATE. | 1 |
| 19 | Итераторы и генераторы. Генерация комбинаторных объектов ITERTOOLS. Функции PARTIAL, REDUCE, ACCUMULATE. | 1 |
| 20 | Объектно-ориентированное программирование. Комплексные числа. Инкапсуляция и конструкторы. | 1 |
| 21 | Объектно-ориентированное программирование. Комплексные числа. Инкапсуляция и конструкторы. | 1 |
| 22 | Классы объектов. Обработка ошибок. Наследование и полиморфизм. | 1 |
| 23 | Классы объектов. Обработка ошибок. Наследование и полиморфизм. | 1 |
| 24 | Решение задач повышенной сложности. | 1 |
| 25 | Решение задач повышенной сложности. | 1 |
| 26 | Решение задач повышенной сложности. | 1 |
| 27 | Решение задач повышенной сложности. | 1 |
| 28 | Решение сложных олимпиадных задач. | 1 |
| 29 | Решение сложных олимпиадных задач. | 1 |
| 30 | Решение сложных олимпиадных задач. | 1 |
| 31 | Решение сложных олимпиадных задач. | 1 |
| 32 | Решение сложных олимпиадных задач. | 1 |
| 33 | Решение сложных олимпиадных задач. | 1 |
| 34 | Решение сложных олимпиадных задач. | 1 |