

**МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
МИНИСТЕРСТВО ОБРАЗОВАНИЯ КАЛИНИНГРАДСКОЙ ОБЛАСТИ  
КОМИТЕТ ПО ОБРАЗОВАНИЮ АДМИНИСТРАЦИИ ГОРОДСКОГО  
ОКРУГА "ГОРОД КАЛИНИНГРАД"**

**МАОУ гимназия № 32**

**СОГЛАСОВАНО**  
Зам. директора по НМР  
Потапенко С.М.

**УТВЕРЖДЕНО**  
Директор  
Белякова В.Н.

Протокол № 1 от  
«30» августа 2023 г.

Приказ 98/1-ос от  
«30» августа 2023 г.

**РАБОЧАЯ ПРОГРАММА**

**«Программирование»**

для обучающихся 7 классов

**Калининград 2023 г.**

## Пояснительная записка

Настоящая рабочая программа для учащихся 7 класса МАОУ гимназии 32 г. Калининграда

Программирование - стержень профильного курса информатики. В современных условиях развитого прикладного программного обеспечения изучение программирования не потеряло свое значение как средство подготовки основной массы школьников к труду, профессиональной деятельности. Изучение основ программирования связано с целым рядом умений и навыков (организация деятельности, планирование ее и т.д.), которые носят общеинтеллектуальный характер. Формирование таких умений и навыков – одна из приоритетных задач современной школы. Очень велика роль изучения программирования для развития мышления школьников, формирования многих приемов умственной деятельности. Здесь роль информатики повторяет роль математики в школьном образовании. Изучая программирование, учащиеся прочнее усваивают основы алгоритмизации, познают азы профессии программиста, приобщаются к алгоритмической культуре, которая необходима для любой будущей профессии. Курс рассчитан на 35 часов, 1 час в неделю. Усвоив материал курса, учащиеся могут применять свои знания в изучении этой темы в базовом курсе программы, в старшей школе при изучении объектно-ориентированного программирования в профильном курсе основной программы. С целью обеспечения индивидуальных потребностей в обучении в программу включены различные формы и виды деятельности, включающие и другие предметы в широкий общеучебный и жизненный контекст (компетентностный подход), дифференцированные задания, используется метод проектов. Изучение курса основано на системно - деятельностном подходе, что позволяет организовать обучение таким образом, чтобы для каждого учащегося выстраивалась траектория самоопределения в учебном процессе.

Цели и задачи курса:

1. Формирование интереса к изучению профессии, связанной с программированием.
2. Формирование алгоритмической культуры.
3. Дать ученику возможность реализовать свой интерес к выбранному курсу.
4. Научить учащихся структурному программированию как методу, предусматривающему создание понятных, локально простых и удобочитаемых программ, характерными особенностями которых является: модульность, использование унифицированных структур следования, выбора и повторения, отказ от неструктурированных передач управления, ограниченное использование глобальных переменных.
5. Приобретение знаний и навыков алгоритмизации учащимися в ее структурном варианте.
6. Освоение всевозможных методов решения задач, реализуемых на языке Python.
7. Развитие алгоритмического мышления учащихся.
8. Формирование навыков грамотной разработки программ.
9. Углубление знаний, умений и навыков решения задач по программированию и алгоритмизации.

Оцениваются результаты освоения курса по 100-балльной шкале с выставлением отметок в электронный журнал.

### *Содержание программы*

Тема «Введение в Python»

Учащиеся должны знать и уметь:

Место языка Python среди языков программирования высокого уровня. Структуру программы на языке Python. Читать синтаксические диаграммы и сопоставлять их с реальными текстами на языке Python. Структуру модулей в Турбо Python. Пользоваться готовыми модулями и разбираться в их структуре, назначении отдельных разделов. Пользоваться стандартным модулем Crt.

## Тема «Данные. Типы данных. Выражения. Операнды. Операции»

Учащиеся должны знать и уметь:

Что такое величина и чем она характеризуется. Что такое операция, операнд и их характеристики; в чем принципиальные отличия величин структурированных и не структурированных; о таких структурах данных, как множество, запись, файл, стек, очередь, строка. Что может входить в состав арифметического выражения; перечень математических функций, входящих в Турбо Python. О логических выражениях и входящих в них операндах, операциях и функциях. Записывать примеры арифметических и логических выражений всех атрибутов, которые могут в них входить.

## Тема «Операторы»

Учащиеся должны знать и уметь:

Перечень основных операторов языка Python. Синтаксис этих операторов. Детали процесса исполнения каждого из операторов. Описывать словесно работу каждого из рассмотренных операторов. Разрабатывать программы обработки числовой и символьной информации, требующие вложения одного и более основных операторов.

## Тема «Перечислимый и интервальный типы данных»

Учащиеся должны знать и уметь:

Назначение перечислимого и интервального типов данных. Какие ограничения связаны с этими типами. Примеры программ, использующих эти типы. Создавать перечислимые типы. Описывать переменные перечислимого типа. Разрабатывать программы, содержащие величины перечислимого типа. Строить интервальный тип на базе произвольного порядкового типа.

## Тема «Процедуры и функции»

Учащиеся должны знать и уметь:

Почему наличие полноценных процедур и функций является принципиально важным для структурно-ориентированного языка высокого уровня. Каковы правила описания процедур в Паскале. Как строится вызов процедур. В чем принципиальные отличия между формальными, локальными и глобальными переменными. В чем отличия между параметрами-переменными и параметрами-значениями, и в каких ситуациях целесообразно использовать те и другие. В чем отличия между процедурами и функциями. Область действия описаний в процедурах. В чем в принципе заключается рекурсия и как она реализована на Паскале. Выделять вспомогательные алгоритмы в сложных задачах. Формировать процедуры и функции. Правильно строить обращения к процедурам и функциям.

## Тема «Структурированные типы данных»

Учащиеся должны знать и уметь:

Какие структурированные типы данных есть в языке Турбо Python. Как формально определять в программе типы «массив», «строка», «множество», «запись», «файл». Какими свойствами обладают данные указанных типов. Какие операции возможны над величинами указанных типов. При решении содержательных задач, в которых целесообразно пользоваться структурированными данными, адекватно выбирать подходящую из перечисленных выше структур. Пользоваться комбинациями структур данных (типа «массив записей» и т.д.). Воспроизвести алгоритмы сортировки линейных числовых массивов и поиска в упорядоченном массиве. Распространить эти алгоритмы на сортировку и поиск в нечисловых массивах, массивах записей.

## Тематическое планирование

№	Тема	теория	практика
1	Введение в Python. Структура программы на языке Python. Рекомендации по стилю записи программы, использование комментариев. Алфавит языка.	1	
2	Типы данных: целый и вещественный, логический и символьный. Константы. Переменные. Организация ввода-вывода. Оператор присваивания.	1	
3	Решение задач		1
4	Арифметические выражения. Стандартные функции. Правила записи арифметических выражений. Операции. Операнды. Следование.	1	
5	Практикум по решению задач		2
6	Обобщающий урок по теме «Алгоритмы линейной структуры»		1
7	Решение задач		1
8	Организация ветвлений в программах. Основные понятия математической логики. Условный оператор. Оператор безусловного перехода.	2	
9	Практикум по решению задач		3
10	Решение задач		1
11	Программирование циклических алгоритмов, виды циклов. Операторы организации циклов. Вложенные циклы.	2	
12	Практикум по решению задач		4
13	Решение задач		1
14	Процедуры. Функции. Рекурсии. Процедуры и функции пользователя.	2	
15	Практикум по решению		5
16	Одномерные массивы: описание и задание элементов, действия над ними. Поиск, замена в одномерном массиве. Сортировка массива. Способы сортировки.	1	
17	Практикум по решению		2
18	Понятие двумерного массива. Действия над элементами массива. Обработка элементов двумерных массивов. Квадратная матрица. Транспонирование матрицы.	2	
19	Практикум по решению		2
20	итого	12	18

### *Используемый УМК*

Учебно-методический комплект по элективному курсу «Программируем на языке Python» включает учебные пособия и практикум:

- Информатика для 10-11 классов: сборник элективных курсов/ авт.-сост. А.А. Чернов, А.А. Чернов. Волгоград: Учитель, 2008 -191с.
  - Попов В.Б. Turbo Pascal для школьников: Учеб. Пособие.- 3-е доп. изд. - М.: Финансы и статистика, 2002.
  - Рапаков Г.Г., Ржеуцкая С.Ю. Turbo Pascal для студентов и школьников. – СПб.: БХВ-Петербург, 2004.
- Информатика. Задачник - практикум в 2т. / Под ред. И.Г. Семакина, Е.К. Хеннера: Том 1. – М.: Бином. Лаборатория Знаний, 2002

### *Планируемые результаты обучения*

*Аналитическая деятельность:*

- определять по блок-схеме, для решения какой задачи предназначен данный алгоритм;
- анализировать изменение значений величин при пошаговом выполнении алгоритма;
- определять по выбранному методу решения задачи, какие алгоритмические конструкции могут войти в алгоритм;
- сравнивать различные алгоритмы решения одной задачи.

*Практическая деятельность:*

- исполнять готовые алгоритмы для конкретных исходных данных;
- преобразовывать запись алгоритма с одной формы в другую;
- строить цепочки команд, дающих нужный результат при конкретных исходных данных для исполнителя арифметических действий;
- строить цепочки команд, дающих нужный результат при конкретных исходных данных для исполнителя, преобразующего строки символов;

**Количество часов****Всего 35 часов; в неделю 1 час**

№ урока	Тема урока	Количество часов	дата	Стандарт/содержания	Универсальные учебные действия	Тип урока	Тип и форма контроля
<b>Введение в Python. Данные. Типы данных – 3 часа</b>							
1.	Введение в Python. Структура программы на языке Python.	1		Рекомендации по стилю записи программы, использование комментариев. Алфавит языка.	<b>Учащиеся должны знать и уметь:</b> Место языка Python среди языков программирования высокого уровня. Структуру программы на языке Python. Читать синтаксические диаграммы и сопоставлять их с реальными текстами на языке Python. Структуру модулей в Турбо Python. Пользоваться готовыми модулями и разбираться в их структуре, назначении отдельных разделов. Пользоваться стандартным модулем Crt.		
2.	Типы данных.	1		Типы данных: целый и вещественный, логический и символьный. Константы. Переменные	<b>Учащиеся должны знать и уметь:</b> Что такое величина и чем она характеризуется. Что такое операция, операнд и их характеристики; в чем принципиальные отличия величин структурированных и не структурированных; о таких структурах данных, как множество, запись, файл, стек, очередь, строка. Что может входить в состав арифметического выражения; перечень математических функций, входящих в Турбо Python. О логических выражениях и входящих в них операндах, операциях и функциях. Записывать примеры арифметических и логических выражений всех атрибутов, которые могут в них входить.		
3.	Организация ввода-вывода.	1		Операторы ввода, вывода, присваивания.	<b>Учащиеся должны знать и уметь:</b> Перечень основных операторов языка Python. Синтаксис этих операторов. Детали процесса исполнения каждого из операторов. Описывать словесно работу каждого из рассмотренных операторов. Разрабатывать программы обработки числовой и символьной информации, требующие вложения одного и более основных операторов.		
<b>Алгоритмы линейной структуры – 5 часов</b>							
4.	Арифметические выражения. Стандартные функции. Правила записи арифметических выражений. Операции. Операнды. Следование.	1		Арифметические выражения. Стандартные функции. Правила записи арифметических выражений. Операции. Операнды. Следование.			
5.	Следование.	1					

6.	Практикум по решению задач	1		Решение задач			
7.		1		Решение задач			
8.	Зачетный класс	1					
<b>Алгоритмы разветвляющейся структуры – 6 часов</b>							
9.	Организация ветвлений в программах. Основные понятия математической логики. Условный оператор. Оператор безусловного перехода.	1		Организация ветвлений в программах. Основные понятия математической логики. Условный оператор. Оператор безусловного перехода.	Учащиеся должны знать: понятие, назначение и синтаксис условного оператора; две формы записи условного оператора. Учащиеся должны уметь: использовать в программах условные операторы; использовать полное и неполное ветвление в алгоритмах; использовать вложенные конструкции; осуществлять выбор по сложному условию; использовать в программах управляющие элементы; использовать управляющий элемент Переключатель; использовать оператор Select Case для множественного выбора; использовать в интерфейсе управляющий элемент флажок; работать с массивами управляющих элементов; создавать фрагменты тестирующих программ.		
10.		1					
11.	Практикум по решению задач	1		Решение задач			
12.		1		Решение задач			
13.		1		Решение задач			
14.	Зачетный класс	1		Решение задач			
<b>Циклы – 7 часов</b>							
15.	Программирование циклических алгоритмов, виды циклов.	1		Программирование циклических алгоритмов, виды циклов. Операторы организации циклов. Вложенные циклы.	Учащиеся должны знать и уметь: виды циклов, формат их записи, правила выполнения и отличия одного циклического оператора от другого		
16.	Операторы организации циклов. Вложенные циклы.	1					
17.	Практикум по решению задач	1		Решение задач			
18.		1		Решение задач			
19.		1		Решение задач			
20.		1		Решение задач			
21.	Зачетный класс	1					

<b>Подпрограммы – 7 часов</b>							
22.	Процедуры. Функции. Рекурсии. Процедуры и функции пользователя.	1		Процедуры. Функции. Рекурсии. Процедуры и функции пользователя.	<b>Учащиеся должны знать и уметь:</b> Почему наличие полноценных процедур и функций является принципиально важным для структурно-ориентированного языка высокого уровня. Каковы правила описания процедур в Паскале. Как строится вызов процедур. В чем принципиальные отличия между формальными, локальными и глобальными переменными. В чем отличия между параметрами-переменными и параметрами-значениями, и в каких ситуациях целесообразно использовать те и другие. В чем отличия между процедурами и функциями. Область действия описаний в процедурах. В чем в принципе заключается рекурсия и как она реализована на Паскале. Выделять вспомогательные алгоритмы в сложных задачах. Формировать процедуры и функции. Правильно строить обращения к процедурам и функциям.		
23.		1					
24.	Практикум по решению задач	1		Решение задач			
25.		1		Решение задач			
26.		1		Решение задач			
27.	Зачетный класс	1					
<b>Массивы – 8 часов</b>							
28.	Одномерные массивы: описание и задание элементов, действия над ними. Поиск, замена в одномерном массиве. Сортировка массива. Способы сортировки.	1			<b>Учащиеся должны знать и уметь:</b> как объявляются и как используются массивы при решении задач; методы сортировок массивов: сортировка подсчетом, вставки, выбором, методом пузырька, с помощью рекурсии и сортировку слиянием;		
29.	Практикум по решению задач №3	1		Решение задач			
30.		1		Решение задач			
31.	Понятие двумерного массива. Действия над элементами массива. Обработка элементов двумерных массивов. Квадратная матрица. Транспонирование матрицы.	1					
32.							
33.	Практикум по решению задач №4	1		Решение задач			
34.		1		Решение задач			
35.	<b>Зачетный класс</b>	1					