

МУНИЦИПАЛЬНОЕ АВТОНОМНОЕ УЧРЕЖДЕНИЕ  
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ  
СТАВРОПОЛЬСКИЙ ДВОРЕЦ ДЕТСКОГО ТВОРЧЕСТВА

Принята на заседании  
научно-методического совета  
от «30» августа 2024 года  
Протокол № 1

УТВЕРЖДАЮ  
Директор *Л.С. Козлова*  
Приказ № 160-ОД  
«30» августа 2024 года



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

Разноуровневая  
технической направленности

«Программирование»

**Уровень программы:** продвинутый

**Возрастная категория:** 11-17 лет

**Состав группы:** 12 человек

**Срок реализации:** 3 года

**ID-номер программы в Навигаторе:** 830

Составитель:  
Тышляр Евгений Игоревич,  
педагог дополнительного образования

г. Ставрополь  
2024 год

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Настоящая программа имеет **техническую направленность**,

**Уровень программы** - продвинутый.

### **Актуальность программы**

В системе дополнительного образования учащихся обучение программированию является востребованным, потому что школьная подготовка по информатике и программированию часто оказывается недостаточной для обеспечения успешного освоения в дальнейшем ряда специальностей.

Информационные (компьютерные) технологии являются одним из наиболее популярных направлений дополнительного образования учащихся из-за стремительного развития электронной техники и средств коммуникаций. Навыки программирования позволят учащимся не только глубже освоить и эффективно использовать компьютерные и сетевые средства, но и активно приспосабливать их для решения новых прикладных задач.

Бурное развитие информационного общества требует от специалиста любой области не только знания основ информатики, но и умения работать с информацией, находить наиболее эффективные способы получения решений конкретных задач. Актуальность данной программы выражается в том, что обучение программированию позволяет развить алгоритмическое мышление, научиться формализовывать и решать различные прикладные задачи.

Программа состоит из двух блоков:

- «Программирование на Паскаль» – 1 год обучения;
- «Объектно-ориентированное программирование» – 2 год обучения.

Каждый блок рассчитан на один год изучения. Занятия проводятся два раза в неделю по два академических часа. Общее количество часов по годам – 144 ч.

### **Задачи:**

- *обучения:* научить учащихся формализовывать различные задачи;
- обучить учащихся практическим навыкам реализации базовых алгоритмов;
- изучить синтаксис языков программирования Pascal и Python;
- дать обучающимся навыки отладки и тестирования программ;

- выработать у учащихся культуру поведения в информационных системах;
- обучить учащихся навыкам решения алгоритмически сложных задач;
- изучить современные средства объектно–ориентированного программирования;
- *воспитательные*: сформировать у учащихся культуру общения и поведения в информационном пространстве;
- прививать детям навыки использования хорошего стиля программирования;
- стимулировать стремление к здоровому образу жизни;
- сформировать навыки и умения работы в коллективе для достижения общей цели;
- *развивающие*:
  - развить навыки алгоритмического мышления;
  - сформировать у учащихся навыки самостоятельного принятия решений;
  - приучить преодолевать трудности;
  - способствовать расширению кругозора.

**Основными принципами организации обучения являются:**

- принцип интерактивности, выражающийся в возможности постоянных контактов всех участников учебного процесса с помощью специализированной информационно-образовательной среды (в том числе форумов, электронной почты, Интернет-тестирования);
- принцип адаптивности, позволяющий использовать учебные материалы нового поколения, содержащие цифровые образовательные ресурсы, в конкретных условиях учебного процесса, что способствует сочетанию разных дидактических моделей проведения занятий с применением дистанционных образовательных технологий и сетевых средств обучения, интерактивных тестов, виртуальных симуляторов, тренажеров, лабораторных практикумов удаленного доступа и др.;

- принцип гибкости, дающий возможность участникам учебного процесса работать в необходимом для них темпе или его корректировке;
- принцип модульности, позволяющий обучающемуся и педагогу использовать необходимые им сетевые учебные курсы (или отдельные составляющие учебного курса) для реализации индивидуальных учебных планов;
- принцип оперативности и объективности оценивания учебных достижений учащихся.

**Особенность и новизна** данной программы заключается в использовании модульного принципа обучения, который предполагает углубленное обучение обучающихся в той области информационных технологий, в которой обучающийся видит перспективу своей дальнейшей практической деятельности или вызывающей наибольший интерес у него. Изучение конкретных модулей программы определяется совместно обучающимся с педагогом.

Реализация программы осуществляется по двум большим модулям, каждый из которых содержит определенные разделы, предусматривающие контроль за их освоением. Описание каждого раздела включает в себя цели и структуру его изучения, аннотацию содержания, перечень знаний и умений, формируемых в ходе обучения.

Возможно изменение содержания обучения в разделах, с учетом реальных требований времени, а также логики его изложения в связи с постоянно изменяющимися технологиями и навыками. Это позволяет перейти от усвоения конкретных знаний, умений, навыков к усвоению их системы и тенденций развития, с использованием разноуровневых подходов.

Все это в целом позволяет выработать универсальный подход, позволяющий рассматривать технологическую и информационную составляющую курса не как независимые компоненты, а как части единого целого.

Контроль знаний, умений, усвоения материала осуществляется по результатам выполнения обучающимися практических заданий на лабораторных практикумах и тестирования.

Для оценивания учебных достижений учащихся, вводится система критериев оценивания по каждому разделу учебного плана. Критерии оценивания по разделам разрабатываются педагогом и являются обязательными при оценивании.

Процесс обучения по данной программе завершается итоговым контролем – выполнением выпускной работы и ее защитой. При этом выпускной работой может являться: создание законченной программы, решающей какую-либо учебную или прикладную задачу, или создание электронного учебника (пособия) по наиболее понравившейся теме данной программы.

Программа предназначена для обучения программированию учащихся в возрасте 14 – 18 лет.

**Региональный компонент.** В Ставрополе процесс компьютеризации проходит интенсивно и уже практически не осталось предприятий и учреждений, которые не использовали бы в своей деятельности информационные технологии. В городе, как и в Ставропольском крае, активно используются компьютеры и Интернет-технологии, как в организациях, так и в домашних условиях.

Проводимая информатизация системы образования Ставропольского края в «пилотном» режиме должна обеспечить высокий уровень образованности в области информационных технологий. Но информатизация не позволяет организовать системный подход к формированию навыков и умений у обучающихся, включая и их дальнейшее самообразование.

Данная программа, системна, направлена на подготовку обучающихся к начальному и дальнейшему применению и освоению быстро меняющихся технологий программирования.

### **Педагогическая целесообразность**

Уровень образования учащихся, прошедших обучение по данной программе, позволит им самостоятельно работать на персональном компьютере при выполнении ими производственных задач в качестве оператора ЭВМ, свободно владеть и использовать информационные технологии (Internet, LAN, ISQ, Web) и различные платформы программирования.

## Формы контроля

Основными формами контроля по разделам (темам), применяющимися в данном направлении, являются: лабораторная работа, практическая работа, зачет, фронтальный опрос, тестирование (в том числе и компьютерное), игра, конкурс творческих работ, реферат, творческий отчет, брейн-ринг, викторина.

Формы проведения начальной, промежуточной и итоговой аттестации следующие:

- устные фронтальные опросы;
- тестирование по темам курса, решение задач;
- защита проектов.

Предусмотрено обязательное проведение занятий по технике безопасности на рабочем месте.

## Учебный (тематический) план первого года обучения

№ п/п	Название темы	Всего часов			Формы контроля
		Всего	Теоретические	Практические	
<b>Раздел № 1 Математические основы программирования.</b>					
1.	<b>Введение в программу</b>	2	2	-	фронтальный опрос
2.	<b>Математические основы программирования:</b> – системы счисления, используемые в цифровой технике; – двоичная система счисления; – двоичная арифметика, сведение арифметических операций к сложению; – логические операции; – алгебра логики.	10	10	-	фронтальный опрос, практическая работа, тестирование
<b>Раздел № 2 Основы программирования на языке pascal.</b>					

3.	<b>Язык программирования pascal:</b> – ABC pascal, общая структура программ; – линейные программы; – написание линейных программ; – условный оператор; – написание программ с ветвлением; – составные условия; – написание программ с составными условиями; – оператор выбора вариантов; – написание программ с выбором вариантов; – логический тип данных; – написание программ с использованием логических переменных.	24	4	20	фронтальный опрос, практическая работа, тестирование
4	<b>Циклические алгоритмические структуры:</b> – цикл с параметром; – написание программ с параметрическим циклом; – вложенные циклы; – написание программ с вложенными циклами; – цикл с предусловием, цикл с постусловием; – написание программ с итерационными циклами.	12	4	8	фронтальный опрос, практическая работа, тестирование
5.	<b>Типы данных:</b> – символьный тип данных; – таблицы символов; – кодировки, перекодирование символов; – обработка символьных данных в программах; – строки; – написание программ обработки строк; – обработка строк; – использование строчных функций в программах.	8	4	4	фронтальный опрос, практическая работа, тестирование

6.	<b>Организация вычислительного процесса:</b> <ul style="list-style-type: none"> <li>– функции;</li> <li>– использование функций в программах;</li> <li>– процедуры;</li> <li>– использование пользовательских процедур;</li> <li>– рекурсивные функции;</li> <li>– написание программ с рекурсивными функциями;</li> <li>– рекурсивные процедуры;</li> <li>– использование рекурсивных процедур.</li> </ul>	16	4	12	фронтальный опрос, практическая работа, тестирование
7.	<b>Сложные типы данных:</b> <ul style="list-style-type: none"> <li>– одномерные массивы;</li> <li>– написание программ с использованием массивов;</li> <li>– сортировка массивов;</li> <li>– написание программ пузырьковой сортировки массивов;</li> <li>– написание программ с использованием сортировки выбором;</li> <li>– быстрая сортировка;</li> <li>– написание программ быстрой сортировки;</li> <li>– двумерные массивы;</li> <li>– написание программ обработки двумерных массивов;</li> <li>– файлы;</li> <li>– написание программ работы с файлами;</li> <li>– интервалы и множества;</li> <li>– написание программ обработки интервалов и множеств;</li> <li>– работа с диапазонами значений.</li> </ul>	28	8	20	фронтальный опрос, практическая работа, тестирование
<b>Раздел № 3 Решение задач программирования.</b>					
8.	<b>Сложные циклические программы:</b> <ul style="list-style-type: none"> <li>– сложные циклы в решении задач численной арифметики;</li> <li>– написание программ со сложными циклами;</li> <li>– числа Армстронга;</li> <li>– обработка числовых множеств.</li> </ul>	8	4	4	фронтальный опрос, практическая работа, тестирование

9	<b>Системы счисления</b> – алгоритмический перевод чисел в двоичную систему счисления, – алгоритмический перевод чисел в шестнадцатиричную систему счисления; – алгоритмический перевод чисел в восьмиричную систему счисления; – перевод чисел в произвольную систему счисления.	8	4	4	фронтальный опрос, практическая работа, тестирование
10	<b>Задачи численной арифметики</b> – алгоритм Эратостена; – числа Мерсенна, – компоновка сложной программы в виде набора функций и процедур – компоновка сложной программы с использованием модулей.	8	4	4	фронтальный опрос, практическая работа, тестирование
11	<b>Решение задач комбинаторики</b> – перестановки из $n$ элементов, – размещения и сочетания из $n$ элементов по $m$ ; – вычисление вероятностей; – решение статистических задач.	8	4	4	фронтальный опрос, практическая работа, тестирование
12	<b>Решение экономических задач методами программирования</b> – имитационные модели систем массового обслуживания; – имитационные модели СМО с отказами; – имитационные модели СМО с очередью; – имитационные модели теории игр, модель воздушного боя.	8	4	4	фронтальный опрос, практическая работа, тестирование
<b>Раздел № 4 Методы программирования.</b>					
13.	<b>Основные подходы к организации программ:</b> – процедурное программирование и его особенности, структурное программирование и сферы его применения, – объектно–ориентированное программирование и его особенности	4	4		фронтальный опрос, практическая работа, тестирование, зачет
	<b>Всего:</b>	72	30	42	

## Содержание программы (1 год обучения)

### *Раздел № 1 Основы программирования на языке pascal.*

#### **Тема 1. Введение в дополнительную общеобразовательную программу**

*Теория.*

Ознакомление со структурой, содержанием, целью, задачами программы, методы преподавания и ожидаемые результаты.

*Форма работы.*

Лекция, рассказ, беседа.

#### **Тема 2. Математические основы программирования**

*Теория.*

Системы счисления, используемые в цифровой технике. Двоичная система счисления. Перевод чисел в двоичную систему счисления. Двоичная арифметика, сведение арифметических операций к сложению, использование обратного и дополнительного кода.

Логические операции. Полные системы логических функций. Алгебра логики. Порядок анализа и минимизации логических выражений, переход к базисам И-НЕ, ИЛИ-НЕ.

*Форма работы.*

Лекция, беседа, составление таблиц истинности сложных логических выражений, тестирование (определение уровня знания ПК).

### *Раздел № 2. Основы программирования на языке pascal*

#### **Тема 3. Язык программирования pascal**

*Теория.*

Изучаются общая структура программ на языке программирования ABC Pascal.Net.

*Практика.*

Составление линейных программ. Организация ветвлений в программах с помощью условного оператора. Способы записи составных условий, использование оператора выбора вариантов. Логический тип данных в программах с ветвлениями.

*Форма работы.*

Лекция, практическая работа.

#### **Тема 4. Циклические алгоритмические структуры**

*Теория.*

Понятие и способы организации циклических вычислений.

*Практика.*

Составление программ использующих цикл с параметром. Вложенные циклы с параметром. Составление программ использующих итерационные циклы с предусловием и с постусловием.

*Форма работы.*

Практическая работа.

#### **Тема 5. Типы данных**

*Теория.*

Знакомство с нечисловыми типами данных. Символьный тип данных. Строки. Функции обработки строк.

*Практика.*

Составление программ работы с символами. Использование функций обработки строк в программах.

*Форма работы.*

Лекция, практическая работа.

## **Тема 6. Организация вычислительного процесса**

### *Теория.*

Процедуры и функции в программах. Понятие рекурсии. Рекурсивные процедуры и рекурсивные функции.

### *Практика.*

Составление программ использующих процедуры и функции пользователя для повторяющихся вычислений. Использование рекурсивных функций в программе. Составление программ использующих рекурсивные процедуры.

### *Форма работы.*

Лекция, практическая работа.

## **Тема 7. Сложные типы данных**

### *Теория.*

Понятие массива. Одномерные и двумерные массивы, порядок поиска элементов в массиве. Методы сортировки массивов. Обработка множеств в программах, работа с интервалами и диапазонами значений. Способы работы с файлами в программах.

### *Практика.*

Составление программ использующих одномерные массивы. Использование циклов для сортировки массивов, поиска нужного элемента массива. Составление программ обработки двумерных массивов. Организация быстрой сортировки данных. Составление программ работы с множествами. Обработка интервалов и диапазонов значений. Составление программ работы с файлами. Считывание данных из файла и запись в файл.

### *Форма работы.*

Лекция- беседа, практическая работа.

### ***Раздел №3 Решение задач программирования.***

#### **Тема 8. Сложные циклические программы**

*Теория.*

Сложные циклы в решении задач численной арифметики. Числа Армстронга.

Порядок обработки числовых множеств.

*Практика.*

Получение чисел Армстронга.

*Форма работы*

Лекция, практическая работа, самостоятельная работа, тестирование.

#### **Тема 9. Системы счисления**

*Теория.*

Алгоритмический перевод чисел в двоичную систему счисления, в шестнадцатиричную систему счисления. Перевод чисел в произвольную систему счисления.

*Практика.*

Программа перевода чисел в произвольную систему счисления.

*Форма работы*

Лекция, практическая работа, самостоятельная работа, тестирование.

#### **Тема 10. Решение задач численной арифметики**

*Теория.*

Алгоритм Эратостена, числа Мерсенна. Компоновка сложной программы в виде набора функций и процедур.

*Практика.*

Алгоритм Эратостена. Числа Мерсенна.

*Форма работы*

Лекция, практическая работа, самостоятельная работа, тестирование.

## **Тема 11. Решение задач комбинаторики**

*Теория.*

Перестановки из  $n$  элементов, размещения и сочетания из  $n$  элементов по  $m$ .  
Вычисление вероятностей, решение статистических задач.

*Практика.*

Расчет вероятности выигрыша в Спортлото.

*Форма работы*

Лекция, практическая работа, самостоятельная работа, тестирование.

## **Тема 12. Решение экономических задач методами программирования**

*Теория.*

Имитационные модели систем массового обслуживания. Имитационные модели теории игр, модель воздушного боя.

*Практика.*

Разработка имитационных моделей СМО, игровых моделей.

*Форма работы*

Лекция, практическая работа, самостоятельная работа, тестирование.

### ***Раздел № 4 Методы программирования.***

## **Тема 13. Основные подходы к организации программ**

*Теория.*

Сравнительная характеристика и области применения различных методов программирования. Процедурное программирование и его особенности. Структурное программирование и сферы его применения. Современные системы объектно–ориентированного программирования. Основные понятия ООП.

*Практика.*

Знакомство с программой в среде программирования Lazarus.

*Форма работы*

Лекция, практическая работа, самостоятельная работа, итоговое тестирование.

### **Прогнозируемые результаты**

#### **К концу 1 года обучения учащиеся должны уметь:**

- реализовывать базовые алгоритмы в среде программирования;
- пользоваться инструментами отладки программ;
- грамотно компоновать программу;
- использовать дополнительные библиотеки для решения задач;
- пользоваться электронной справкой, осуществлять поиск информации в Интернете;
- использовать различные типы данных в том числе нестандартные;
- создавать имитационные программы.

#### **Учащиеся должны знать:**

- синтаксис языка программирования;
- типы данных;
- основные логические операции;
- правила ввода, редактирования и форматирования программного кода;
- виды алгоритмов и их особенности;
- единицы измерения информации и их физический смысл;
- порядок тестирования и отладки программ;
- порядок использования электронных справочников и ресурсов Интернета;
- правила обработки расчетных данных.
- состав и порядок реализации имитационных программ;
- правила оформления текстовых пояснений в программах;
- особенности сетевых программ;
- средства защиты информации.

**Учащиеся должны владеть основными приемами программирования на языке Pascal.**

## Учебно – тематический план второго года обучения

№ п/ п	Название темы	Всего (час.)	Теоре- тиче- ские за- нятия	Практи- ческие занятия	Формы кон- троля
1.	<b>Введение в программу</b>	2	2	-	
<i><b>Раздел № 1 Среда программирования Python.</b></i>					
2.	<b>Особенности Python:</b> – установка, настройка, интерфейс; – работа в интерактивном режиме.	4	-	4	фронтальный опрос, практическая работа, тестирование
3.	<b>Переменные, выражения и инструкции:</b> – значения и типы данных; – имена переменных, ключевые слова; – операторы и операнды, выражения; – ввод-вывод данных, комментарии.	8	4	4	фронтальный опрос, практическая работа, тестирование
4.	<b>Линейные программы:</b> – отличия работы в режимах shell и edit; – написание линейной программы.	4	-	4	фронтальный опрос, практическая работа, тестирование
5.	<b>Ветвления в программах:</b> – логические выражения и операторы; – альтернативное исполнение, последовательность условий, вложенные условия; – перехват исключений с использованием try - except; – написание программы с ветвлениями.	8	4	4	фронтальный опрос, практическая работа, тестирование
6.	<b>Функции:</b> – встроенные функции и преобразование типов данных; – функции пользователя, параметры и аргументы; – написание программы с функциями пользователя.	6	2	4	фронтальный опрос, практическая работа, тестирование

7.	<b>Циклические программы:</b> – итерационные циклы; – циклы с заданным числом повторений; – управление циклом с помощью break и continue; – написание программ циклических вычислений.	8	2	6	фронтальный опрос, практическая работа, тестирование
8.	<b>Строки:</b> – строковые операторы; – циклическая обработка строк; – строковые методы.	6	2	4	фронтальный опрос, практическая работа, тестирование
9.	<b>Файлы:</b> – работа с файлами, текстовые файлы и строки; – чтение и поиск в файле, печать файла.	4	-	4	фронтальный опрос, практическая работа, тестирование
10	<b>Списки:</b> – обработка списков; – методы списков, псевдонимы; – программа поиска строки.	6	2	4	фронтальный опрос, практическая работа, тестирование
11	<b>Словари:</b> – словари; – поиск слов.	4	2	2	фронтальный опрос, практическая работа, тестирование
12	<b>Кортежи:</b> – обработка кортежей; – множественное присваивание при работе со словарями; – использование кортежей в качестве ключей; – последовательности.	8	4	4	фронтальный опрос, практическая работа, тестирование
13	<b>Регулярные выражения:</b> – сопоставление символов в регулярных выражениях; – сочетание поиска и извлечения.	4	2	2	фронтальный опрос, практическая работа, тестирование

14	<b>Сетевые программы</b> – протокол HTTP; – извлечение информации из web-страниц; – чтение бинарных файлов.	6	2	4	фронтальный опрос, практическая работа, тестирование
15	<b>Использование web-служб</b> – расширяемый язык разметки XML; – интерфейсы прикладного программирования.	4	2	2	фронтальный опрос, практическая работа, тестирование
16	<b>Использование баз данных и языка структурированных запросов (SQL)</b> – базы данных и таблицы; – обзор языка SQL; – моделирование данных; – виды ключей и их использование.	8	4	4	фронтальный опрос, практическая работа, тестирование
17	<b>Автоматизация типичных задач на компьютере</b> – имена файлов и пути; – аргументы командной строки; – программные каналы.	6	2	4	фронтальный опрос, практическая работа, тестирование
18	<b>Основные стандартные модули Python</b> – состав модулей, использование стандартных функций; – взаимодействие с ОС.	4	-	4	фронтальный опрос, практическая работа, тестирование
19	<b>Элементы функционального программирования</b> – рекурсия; – рекуррентные функции; – функции и итераторы.	6	4	2	фронтальный опрос, практическая работа, тестирование
20	<b>Объектно-ориентированное программирование:</b> – объекты, классы, наследование; – инкапсуляция, полиморфизм; – агрегация, ассоциация; – методы.	8	4	4	фронтальный опрос, практическая работа, тестирование

21	<b>Численные алгоритмы. Матричные вычисления:</b> – модуль Numeric; – решение вычислительных задач; – модуль LinearAlgebra.	6	2	4	фронтальный опрос, практическая работа, тестирование
22	<b>Обработка текстов. Регулярные выражения. Unicode:</b> – работа со строками, форматирование; – регулярные выражения, Unicode.	4	-	4	фронтальный опрос, практическая работа, тестирование
23	<b>Работа с данными в различных форматах:</b> – формат CSV; – формат XML.	4	2	2	фронтальный опрос, практическая работа, тестирование
24	<b>Разработка Web-приложений:</b> – CGI-сценарии – среды разработки.	4	2	2	фронтальный опрос, практическая работа, тестирование
25	<b>Многопоточные приложения:</b> – потоки управления; – замки, семафоры, события.	4	2	2	фронтальный опрос, практическая работа, тестирование
26	<b>Создание приложений с графическим интерфейсом пользователя:</b> – обзор графических библиотек; – создание и конфигурирование виджета.	4	-	4	фронтальный опрос, практическая работа, тестирование
27	<b>Интеграция Python с другими языками программирования:</b> – модули расширения; – встраивание интерпретатора в другие языки.	4	2	2	фронтальный опрос, практическая работа, тестирование
	<b>Всего:</b>	144	54	90	

## Содержание программы

(2 год обучения)

### **Тема 1. Введение в программу**

*Теория.*

Уточнение цели и задач второго года обучения по данной программе.

*Форма работы.*

Лекция, беседа.

### **Тема 2. Введение в Python**

*Теория.*

Исторический экскурс. Порядок установки. Настройка.

Интерфейс. Порядок установки.

*Практика.*

Создание первой программы

*Форма работы.*

Лекция, беседа, практическая работа.

### **Тема 3. Переменные, выражения и инструкции:**

*Теория.*

Значения и типы данных. Имена переменных, ключевые слова. Операторы и операнды, выражения. Ввод-вывод данных, комментарии.

*Практика.*

Использование основных типов данных в программе

*Форма работы.*

Лекция, беседа, практическая работа.

### **Тема 4. Линейные программы:**

*Теория.*

Интерпретаторы и компиляторы. Отличия работы в режимах shell и edit

*Практика.*

написание линейной программы.

*Форма работы.*

Лекция, практическая работа.

## **Тема 5. Ветвления в программах:**

*Теория.*

Логические выражения и операторы. Альтернативное исполнение, последовательность условий, вложенные условия. П перехват исключений с использованием try – except.

*Практика*

написание программы с ветвлениями

*Форма работы.*

Лекция, беседа.

## **Тема 6. Функции:**

*Теория.*

Встроенные функции и преобразование типов данных. Функции пользователя, параметры и аргументы.

*Практика.*

Написание программы с функциями пользователя.

*Форма работы.*

Лекция, практическая работа.

## **Тема 7. Циклические программы:**

*Теория.*

Циклы с заданным числом повторений. Итерационные циклы. Управление циклом с помощью break и continue.

*Практика.*

Написание программ циклических вычислений.

*Форма работы.*

Практическая работа.

## **Тема 8. Строки:**

*Теория.*

Строковые операторы. Строковые методы. Срезы.

*Практика.*

Написание программ обработки строк.

*Форма работы.*

Лекция, беседа, практическая работа.

## **Тема 9. Файлы:**

*Теория.*

работа с файлами, текстовые файлы и строки.

*Практика.*

Чтение и поиск в файле, печать файла.

*Форма работы.*

Лекция, практическая работа.

## **Тема 10. Списки:**

*Теория.*

Обработка списков. Методы списков, псевдонимы.

*Практика.*

Написание программы поиска строки.

*Форма работы.*

Лекция, практическая работа.

## **Тема 11. Словари**

*Теория.*

Словари. Поиск слов. Обработка индексов и обработка значений.

*Практика.*

Использование словарей в программе поиска слов.

*Форма работы.*

Лекция, практическая работа.

## **Тема 12. Кортежи:**

*Теория.*

Кортежи. Обработка кортежей. множественное присваивание при работе со словарями. Использование кортежей в качестве ключей. Последовательности.

*Практика.*

Написание программы обработки последовательностей.

*Форма работы.*

Лекция, практическая работа.

## **Тема 13. Регулярные выражения:**

*Теория.*

Сопоставление символов в регулярных выражениях.

*Практика.*

Сочетание поиска и извлечения.

*Форма работы.*

Лекция, практическая работа.

## **Тема 14. Сетевые программы**

*Теория.*

Протокол HTTP. Извлечение информации из web-страниц. Чтение бинарных файлов.

*Практика.*

Написание сетевой программы.

*Форма работы.*

Лекция, практическая работа.

## **Тема 15. Использование web-служб**

*Теория.*

Расширяемый язык разметки XML. Интерфейсы прикладного программирования.

*Практика.*

Создание и наполнение сайта с помощью программных сред.

*Форма работы.*

Лекция, практическая работа.

## **Тема 16. Использование баз данных и языка структурированных запросов (SQL)**

*Теория.*

Базы данных и таблицы. Обзор языка SQL. Моделирование данных. Виды ключей и их использование.

*Практика.*

Создание запросов к базе данных.

*Форма работы.*

Лекция, практическая работа.

## **Тема 17. Автоматизация типичных задач на компьютере**

*Теория.*

Имена файлов и пути. Аргументы командной строки. Программные каналы.

*Практика.*

Создание программы взаимодействующей с ОС.

*Форма работы.*

Лекция, практическая работа.

## **Тема 18. Основные стандартные модули Python**

*Теория.*

Состав модулей, использование стандартных функций. Взаимодействие с ОС

*Практика.*

Подключение модулей к программе.

*Форма работы.*

Лекция, практическая работа.

## **Тема 19. Элементы функционального программирования**

*Теория.*

Рекурсия. Рекуррентные функции. Функции и итераторы.

*Практика.*

Программа поиска пути в графе.

*Форма работы.*

Лекция, практическая работа.

## **Тема 20. Объектно-ориентированное программирование:**

*Теория.*

Объекты, классы, наследование. Инкапсуляция, полиморфизм. Агрегация, ассоциация. Методы классов

*Практика.*

Создание классов и объектов

*Форма работы.*

Лекция, практическая работа.

## **Тема 21. Численные алгоритмы. Матричные вычисления:**

*Теория.*

Модуль Numeric, модуль LinearAlgebra.

*Практика.*

Решение вычислительных задач

*Форма работы.*

Лекция, практическая работа.

## **Тема 22. Обработка текстов. Регулярные выражения. Unicode:**

*Теория.*

Работа со строками, форматирование. Регулярные выражения, Unicode.

*Практика.*

Преобразование кодиовок данных.

*Форма работы.*

Лекция, практическая работа.

## **Тема 23. Работа с данными в различных форма-тах:**

*Теория.*

*Форматы данных. Формат CSV. Формат XML*

*Практика.*

Преобразование данных различных форматов.

*Форма работы.*

Лекция, практическая работа.

## **Тема 24. Разработка Web-приложений:**

*Теория.*

CGI-сценарии. Современные среды разработки web-страниц.

*Практика.*

Создание web-страниц.

*Форма работы.*

Лекция, практическая работа.

## **Тема 25. Многопоточные приложения:**

### *Теория.*

Потоки управления и способы их синхронизации. замки, семафоры, события.

### *Практика.*

Изучение взаимодействия потоков.

### *Форма работы.*

Лекция, беседа.

## **Тема 26. Создание приложений с графическим интерфейсом пользователя**

### *Теория.*

Обзор графических библиотек.

### *Практика.*

Создание и конфигурирование виджета.

### *Форма работы.*

Лекция, практическая работа.

## **Тема 27. Интеграция Python с другими языками программирования**

### *Теория.*

Модули расширения.

### *Практика.*

встраивание интерпретатора в другие языки.

### *Форма работы.*

Лекция, практическая работа.

**К концу 2 года обучения учащиеся должны уметь:**

- использовать средства объектно-ориентированного программирования для

создания интерфейсов и разработки программных модулей;

- создавать программы для решения различных задач;
- осуществлять перекомпиляцию программы для работы в другой ОС.

**Учащиеся должны знать:**

- принципы объектно-ориентированного программирования;
- состав и возможности среды программирования;
- назначение библиотек компонентов;
- порядок редактирования свойств и поведения компонентов;
- порядок компиляции, отладки и тестирования программ.

**Учащиеся должны владеть основными приемами объектно-ориентированного программирования**

## Методическое обеспечение

**Основной формой обучения по программе** – является проведение занятий (в группах). Кроме этого широко используются такие формы работы, как организация игровых занятий (индивидуально и командная работа), конкурсов, проектной деятельности.

При этом подготовка комплектов учебно-методических материалов для обучения учащихся подготавливаются с использованием Internet и работой учащихся, в том числе и самостоятельной в информационной сети.

Каждая тема начинается с постановки задач. После изучения нового материала проводится тренинг по отработке умений выполнять технические задачи, соответствующие минимальному уровню планируемых результатов обучения или тестирование. Тренинг переходит в творческую работу по созданию учащимися определенного образовательного продукта или игры.

В ходе обучения периодически проводятся краткие срезовые работы (тесты, творческие работы) по определению уровня знаний и умений учеников по пройденным разделам. В процессе обучения широко используются цифровые образовательные ресурсы:

- электронные учебники;
- интерактивные обучающие ресурсы (с использованием сети Internet);
- виртуальные среды учебно-практической деятельности;
- компьютерные демонстрации;
- компьютерные средства имитационного моделирования;
- электронные источники информации;
- электронные библиотеки;
- электронные тренажеры;
- электронные системы контроля и измерения результативности обучения.

Важным условием реализации программы является использование разнообразных **методов**: объяснительно-иллюстративных, репродуктивных, исследовательских, игровых, методов коррекции знаний учащихся, стимулирования и мотивации обучения и др. Так, объяснительно-иллюстративные методы при исполь-

зовании мультимедийного проектора позволяют заметно повысить познавательную активность учащихся за счет увеличения наглядности и эмоциональной насыщенности (анимация, звук, видео и другие мультимедийные эффекты).

Репродуктивные методы обучения при использовании компьютерных обучающих систем приобретают свойства личностно-ориентированного обучения, при котором учащиеся получают возможность выстраивать индивидуальные образовательные траектории в зависимости от успешности обучения и личностных психологических качеств (восприятия, памяти, мышления и пр.).

В процессе работы с обучающими системами можно активизировать методы коррекции знаний учащихся. Эти средства образовательного назначения так же могут являться средством стимулирования и повышения мотивации обучения, а так же средством повышения познавательного интереса учащихся.

Наряду с общедидактическими методами обучения на уроках информатики применяют частно-методические. Так, при изучении определенных разделов применяются такие методы, как ролевое исполнение задания, "черный ящик", приемы: усложнение задачи, "найди ошибку в алгоритме", таблица значений и др.

Групповая и индивидуальная учебная деятельность так же предполагает проектную форму работы. Выполнение проектов завершается их тестированием и защитой с оценкой самих же создателей и группой.

Каждый проект включает в себя основные составляющие, которые формируются совместно с учащимися и преподавателем:

- тема;
- длительность проекта;
- цели и содержание проекта;
- задачи проекта;
- используемые библиотеки;
- необходимое оборудование;
- результаты проекта.

Уровень реализации программных проектов и их защита является главным показателем и средством оценки достижений учащихся.

Реализация программы предполагает использование элементов психологических методик, в особенности, саморефлексии каждого обучающегося, одновременно с его корректной коллективной оценкой другими воспитанниками группы.

Использование элементов психологических методик необходимо при реализации данной образовательной программы. В ходе коллективной работы (выполнения проектов, заданий лабораторных работ) учащиеся получают определенные социальные роли в каждой группе. В этом случае выполнение роли в коллективе равносильно достижению им определенного показателя. Кроме того, поскольку часть таких ролей являются формальными и выборными, обучающийся имеет возможность занять их через самопрезентацию (демонстрация собственных достижений, поиск способа их показа другим).

Организация процесса обучения по данной программе строится с использованием таких технологий как личностно-ориентированное и развивающее обучение с направленностью на развитие творческих качеств личности.

Конкретные методы, используемые при реализации программы:

- практический – различные упражнения с алгоритмами, создание программ);
- наглядный – создание работающих программ, имитирующих случайные процессы в игре;
- словесный – работа с книгами, ресурсами локальной, глобальной сети, доклад реферата, демонстрация оригинальных программ.

В воспитании:

- методы формирования сознания личности, направленные на формирование устойчивых убеждений (рассказы, дискуссии);
- методы стимулирования поведения и деятельности (соревнования, конкурсы, поощрения).

Опыт практической деятельности по реализации данной программы позволил определить следующие наиболее успешные **формы проведения занятий**: бесе-

да, демонстрация видеоматериалов, лекция, тестирование, практическая работа (написание программы, реализующей алгоритм), игра или соревнования, конкурсы.

К разделам программы разработаны комплекты учебно-методических материалов, включающие образовательные ресурсы - электронный учебник, демонстрационные и раздаточные материалы, методические материалы, тестирование.

В рамках программы был разработан лекционный материал, включающий лекции, сопровождающиеся демонстрационными видеуроками. Лекционный материал позволяет описать достаточно сложные информационные структуры и алгоритмы языком, понятным для учащихся разных возрастных категорий.

Для иллюстрации использования различных алгоритмов разработаны демонстрационные программы решения различных задач.

При выполнении практических работ обучающийся выполняет поэтапно практическое задание, при этом он не может перейти к следующему этапу, пока не выполнит предыдущий. Это позволяет системно продвигаться от простого к более сложному. В случае правильного выполнения всех этапов, результатом является верно выполненное задание и следствии этого обучающийся получает работающую программу.

К программе были так же разработан блок тестовых заданий и раздаточного материала, позволяющий обучающемуся более полно понять и усвоить изучаемый материал.

Тестовый блок представляет собой набор электронных тестов с вопросами, относящимися к конкретной теме или разделу программы. Вопросы представлены четырьмя видами:

- выбором одного правильного ответа из предложенного множества;
- выбором нескольких правильных ответов из предложенного множества;
- ввод ответа (числового или односложного) с клавиатуры;
- установление последовательности в множестве элементов выбором выпадающих счетчиков.

Учебно-методический комплекс к программе пополняется методическими разработками и тестами, доступными учащимся, зарегистрированным в учебной сети через интернет.

### **Материально – техническое обеспечение программы:**

Реализация программы требует наличия следующей материально-технической базы:

- Мультимедийный компьютерный класс (10 компьютеров Intel Pentium IV 3,5 ГГц/ RAM – 1 Gb /HDD – 120 Gb/DVD-RW/LAN).
- Преподавательский компьютер (Intel Pentium Core i2 Duo 2,5 ГГц/ RAM – 1 Gb/HDD – 120 Gb/DVD-RW/LAN).
- Локально-вычислительная сеть (100 Мб/с).
- Глобальная сеть Internet (10 Мб/с).
- Интерактивная доска.
- Сканер.
- Принтер.
- Наушники + микрофоны.
- Доска.
- Раздаточный материал.

### **Программное обеспечение**

- Операционная система Windows Server 2000 / 2003.
- Браузер (Mozilla Firefox, блокнот).
- Пакет Microsoft Office /2003.
- Графические редакторы Adobe Photoshop CS.
- AVP Kaspersky Personal 6.0/7.0.
- Электронные учебники (HTML, Mp4).
- Язык программирования PascalABC.NET
- Язык программирования Python 3.4

### ***ОСОБЕННОСТИ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ***

Данную программу осваивают подростки 14-18 лет с ярко выраженным интересом к информатике и программированию. Мотивом обучения является желание овладеть знаниями и навыками, которые помогут в дальнейшей учебе и профессиональной деятельности.

Большое достоинство подростка — его готовность к освоению высокотехнологичных знаний и навыков, которые делают его взрослым в собственных глазах. Также его привлекают самостоятельные формы организации занятий на уроке, сложный учебный материал, возможность самому строить свою познавательную деятельность за пределами школы.

Сформированной учебной деятельностью считается такая деятельность учащихся, когда они, побуждаясь прямыми мотивами самого учения, могут самостоятельно определять учебные задачи, выбирать рациональные приемы и способы их решения, контролировать и оценивать свою работу.

Одним из резервов повышения эффективности обучения подростков является целенаправленное формирование мотивов учения. Формирование мотивов учения непосредственно связано с удовлетворением доминирующих потребностей возраста.

Одна из таких потребностей подростка - познавательная. При ее удовлетворении у него формируются устойчивые познавательные интересы, которые определяют его положительные отношения к учебным предметам. Подростков привлекает возможность расширить, обогатить свои знания, проникнуть в сущность изучаемых явлений, установить причинно-следственные связи.

Наряду с познавательными интересами первостепенное значение при положительном отношении подростков к учению имеет понимание значимости знаний.

Нужно поддерживать убеждение подростков в том, что только образованный человек может быть по-настоящему полезным обществу человеком.

Эмоциональное благополучие подростка во многом зависит от оценки учебной деятельности взрослыми. Нередко же смысл оценки для него выступа-

ет в стремлении добиться успеха в учебном процессе и тем самым получить уверенность в своих умственных способностях и возможностях.

В подростковом возрасте доминирующую роль начинает играть самооценка. Для эмоционального благополучия подростка очень важно, чтобы оценка и самооценка совпадали.

У старшеклассников, интерес к учению повышается. Это связано с тем, что складывается новая мотивационная структура учения.

Ведущее место занимают мотивы, связанные с самоопределением и подготовкой к самостоятельной жизни.

В связи с высоким уровнем самокритичности они, анализируя условия, мешающие учиться, указывают на проявление у них таких отрицательных черт характера, как безволие, неусидчивость, неумение организовать свое время и самоконтроль, настроение.

Учеба, приобретение знаний становятся прежде всего средством подготовки к будущей деятельности. Внутренняя позиция старшеклассника по отношению к обучению формируется из целой системы отношений - к образовательному учреждению, процессу обучения и знаниям, учителям, сверстникам.

Качественное своеобразие познавательной деятельности старшеклассников заключается в стремлении разобраться в разных точках зрения на заданный вопрос, их больше интересует ход анализа, способы доказательства, чем конкретные сведения.

Очень важно учить старшеклассников планировать свою работу, выбирать индивидуальный оптимальный режим дня, рационально использовать свое время.

## СПИСОК ЛИТЕРАТУРЫ

### Учебно-информационное обеспечение программы

#### Нормативно-правовые акты и документы:

1. Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации».
2. Концепция развития дополнительного образования детей (утверждена распоряжением Правительства РФ от 04 сентября 2014 г. № 1726-р).
3. Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (утвержден Приказом Министерства просвещения РФ от 09 ноября 2018 г. № 196)
4. Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы): приложение к письму Министерства образования и науки РФ от 18 ноября 2015 г. № 09-3242.
5. СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей» (утверждены постановлением Главного государственного санитарного врача РФ от 4 июля 2014 г. № 41).
6. Устав муниципального автономного учреждения дополнительного образования Ставропольского Дворца детского творчества (разработан в соответствии с действующим законодательством на основании Гражданского кодекса Российской Федерации, Федеральным законом от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации», приказа Министерства образования и науки Российской Федерации от 29 августа 2013 года № 1008 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»)

#### Литература для педагога:

1. Дополнительное образование учащихся. Учебное пособие для вузов. Под ред. Лебедева О.Е. -М.. «Владос». 2016.

2. Архангельский А. Я. Программирование в Delphi : учебник по классическим версиям Delphi : [версии 5-7] / А. Я. Архангельский. - М., 2012. - 804, [5] с. : ил. + 1 CD-ROM.

3. Д. Кнут "Искусство программирования", том 1. Основные алгоритмы (djvu)

4. Д. Кнут "Искусство программирования", том 2. Получисленные алгоритмы (djvu)

5. Д. Кнут "Искусство программирования", том 3. Сортировка и поиск (djvu)

6. Д. Кнут "Искусство программирования", том 4. Комбинаторные алгоритмы (djvu)

7. Горский В.А. Технология разработки авторской программы дополнительного образования учащихся. Дополнительное образование - 2012 № 1. с. 30-31.

#### **Для обучающихся:**

1. Язык программирования Паскаль. Электронный учебник.

2. Дискретная математика. Электронный учебник.

3. Новиков Ф.А. Дискретная математика для программистов, СПб., Питер, 2006.

4. Уроки программирования для новичков. Электронное учебное пособие.

5. Уроки Python. Электронное пособие.

#### **Образовательно-информационные ресурсы:**

1. Электронный научный журнал. Информационно- коммуникационные технологии в педагогическом образовании — Электронные учебные пособия, и их важность в учебном процессе [Электронный ресурс]. — Режим доступа: <http://journal.kuzspa.ru/articles/87/> .

2. Язык Pascal. Программирование для начинающих. [Электронный ресурс]. — Режим доступа: <http://pas1.ru>

3. AcademiaXXI — Рекомендации по созданию электронного учебника [Электронный ресурс]. — Режим доступа:  
[http://www.academiaxxi.ru/Meth\\_Papers/AO\\_recom\\_t.htm](http://www.academiaxxi.ru/Meth_Papers/AO_recom_t.htm).
4. DELPHISOURCES. [Электронный ресурс]. — Режим доступа:  
<http://www.delphisources.ru/>.
5. HTML.net—Учебники [Электронный ресурс]. — Режим доступа:  
<http://ru.html.net/tutorials/>.
6. Pascal-Паскаль. [Электронный ресурс]. — Режим доступа:  
<http://www.pascal.helpov.net//>.
7. ProCoder. Некоммерческий электронный журнал [Электронный ресурс].  
— Режим доступа: <http://c> <http://procoder.info//>.
8. SuperInf.ru — Общие требования к электронным средствам обучения [Электронный ресурс]. — Режим доступа: [http://superinf.ru/view\\_article.php?id=416](http://superinf.ru/view_article.php?id=416)

.