

Комитет администрации г. Славгорода Алтайского края по образованию  
Муниципальное бюджетное общеобразовательное учреждение  
«Средняя общеобразовательная школа № 13»  
города Славгорода Алтайского края

Принята на заседании  
Педагогического совета  
МБОУ «СОШ № 13»  
г. Славгорода Алтайского края  
от «30» августа 2022 г.

Утверждено приказом  
директора  
МБОУ «СОШ №13»  
от 30 августа 2022 г. № 300



**Дополнительная общеобразовательная  
общеразвивающая программа  
технической направленности  
«Программируем на языке Pascal»**

Возраст обучающихся: 13-16 лет

Срок реализации: 1 год

Автор- составитель:  
Губер Ирина Анатольевна,  
учитель информатики

Славгород 2022 г.

## **Пояснительная записка**

### **Место курса в образовательном процессе**

Изменение взглядов на предмет информатики как науки, её место в системе научного знания требует существенных изменений в содержании образования по информатике. В связи с этим особую актуальность приобретают раскрытие личностных резервов учащихся и создание соответствующей среды.

Никакая система задач, какой бы хорошей она ни была, никакие тренинги памяти, внимания и т. п. не дают того эффекта, который возникает в случае, если учащиеся осознают необходимость решения тех или иных задач, если у них появляется острая необходимость к преодолению интеллектуальных трудностей, связанных с познанием, если они видят смысл в сотрудничестве с одноклассниками и учителем.

Содержание обучения, представленное в программе курса «Программируем на языке Pascal», позволяет вести обучение школьников в режиме актуального познания. Практическая направленность курса на создание внешних образовательных продуктов — блок-схем, алгоритмов, исполняемых файлов — способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у школьников знаний. Возникающие при этом познавательные переживания обуславливают сознательное отношение к изучению основных теоретических положений информатики.

Проявления трудолюбия, целеустремлённости и одухотворённости, возникающие при воплощении замыслов учащихся в рамках курса «Программируем на языке Pascal», стимулируют развитие индивидуально-личностных качеств школьников.

Курс «Программируем на языке Pascal» является предметом по выбору для учащихся средней и старшей ступени. При создании курса предполагалось, что учащиеся ранее не были знакомы с базовыми понятиями языка Pascal и его возможностями. Здесь они научатся работать с основными типами данных и операторов, познакомятся с построением графических изображений средствами языка Pascal.

Активизация познавательного процесса позволяет учащимся более полно выражать свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт предпосылки по применению освоенных навыков программирования в других учебных курсах, а также способствует возникновению дальнейшей мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

### **Общие требования к образованности учащихся**

Для качественного обучения необходимо, чтобы учащиеся обладали базовыми знаниями по математике на уровне 7 класса и навыками работы в файловой системе Linux.

Также приветствуется умение учиться независимо от других, планировать и организовывать свою деятельность.

Ключевой особенностью курса является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и реализации алгоритма с помощью средств программирования.

Для школьников, выбравших информационно-технологический профиль, этот курс — возможность развить навыки программирования на языке Pascal. Программирование — это стержень как базового, так и профильного курсов информатики. В рамках предлагаемого курса «Программируем на языке Pascal» изучение основ программирования на языке Pascal — это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др. Исключительно велика роль программирования для формирования мышления школьников, приёмов умственных действий, умения строить модели, самостоятельного нахождения и составления алгоритмов решения задач, умения чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования общеинтеллектуальных и общеучебных умений школьников активизирует процесс индивидуально-личностного становления учащихся.

Общепедагогическая направленность занятий — гармонизация индивидуальных и социальных аспектов обучения по отношению к информационным технологиям. Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции — одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных школьников. Особая роль отводится широко представленной в курсе системе рефлексивных заданий. Освоение рефлексии направлено на осознание учащимися того важного обстоятельства, что наряду с разрабатываемыми ими продуктами в виде программ на

компьютере рождается основополагающий образовательный продукт: освоенный инструментарий. Именно этот образовательный продукт станет базой для творческого самовыражения учащихся в форме различных программ.

**Адресат программы.** Программа ориентирована на дополнительное образование обучающихся 13-16 лет, интересующихся программированием.

**Срок и этапы реализации программы.** Программа рассчитана на 1 год обучения - 34 академических часа.

Основной формой обучения являются групповые занятия. В основе образовательного процесса лежит проектный подход.

**Режим занятий:** 1 раз по 1 часу в неделю. Продолжительность 1 занятия: 1 академический час.

Структура занятия: • 20 минут – рабочая часть; • 5 минут – перерыв (отдых); • 15 минут – рабочая часть.

**Основная форма работы теоретической части** – лекционные занятия в группах до 10 человек. Практические задания планируется выполнять индивидуально, в парах и в малых группах. Занятия проводятся в виде бесед, семинаров, лекций: для наглядности изучаемого материала используется различный мультимедийный материал – презентации, видеоролики.

### **Цели изучения курса**

Основными целями курса являются:

- понять значение алгоритмизации как метода познания окружающего мира, принципы структурной алгоритмизации;
- овладеть базовыми понятиями теории алгоритмов;
- освоить понятие алгоритма и особенности реализации алгоритмов в виде программ, написанных на языке программирования Pascal.

### **Задачи курса**

Основными задачами курса являются:

- познакомиться с понятиями «алгоритм», «язык программирования»;
- научиться составлять и читать блок-схемы;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Pascal;
- изучить основные конструкции языка программирования Pascal;
- научиться работать с графическими средствами языка программирования Pascal;
- научиться отлаживать и тестировать программы, делать выводы о работе этих программ.

### **Планируемые результаты курса**

В рамках курса «Программируем на языке Pascal» учащиеся овладевают следующими знаниями, умениями и способами деятельности:

- умеют составлять алгоритмы для решения задач;
- умеют реализовывать алгоритмы на компьютере в виде программ, написанных на языке Pascal;
- владеют основными навыками программирования на языке Pascal;
- умеют отлаживать и тестировать программы, написанные на языке Pascal.

**Методы:** кейс-метод, проектная деятельность.

**Формы работы:**

- практическое занятие;
- занятие
- соревнование;
- консультация;
- самостоятельная работа.

### **Требования к уровню подготовки курса**

#### **Раздел 1. Основы алгоритмизации**

*Учащиеся должны знать / понимать:*

- понятие алгоритма;
- понятие исполнителя;
- назначение и основные команды среды исполнителя;
- типы алгоритмов и их свойства;
- язык блок-схем.

*Учащиеся должны уметь:*

- составлять несложные алгоритмы для исполнителя;
- записывать алгоритм разными способами;
- определять исполнителя алгоритма.
- **Практическая работа:** создание различных алгоритмов

## **Раздел 2. Знакомство с Pascal**

### **Тема 2.1. Структура программы на языке Pascal**

*Учащиеся должны знать / понимать:*

- назначение и основные команды среды разработки;
- общую структуру программы;
- назначение и виды оператора вывода.

*Учащиеся должны уметь:*

- устанавливать и запускать среду программирования в эмуляторе DOSBox;
- пользоваться интерфейсом среды программирования Borland Pascal;
- использовать команды редактора;
- составлять и запускать программы;
- организовывать вывод данных.

**Практическая работа:** создание, сохранение, запуск простейшей программы в среде разработки.

### **Тема 2.2. Числовые типы данных**

*Учащиеся должны знать / понимать:*

- понятие типа данных;
- целые, вещественные типы данных и операции над ними;
- понятие переменной;
- оператор присваивания;
- назначение и виды оператора ввода.

*Учащиеся должны уметь:*

- определять тип числовых данных;
- объявлять необходимые переменные;
- записывать арифметические выражения.

**Практическая работа:** составление вычислительных программ.

### **Тема 2.3. Подпрограммы**

*Учащиеся должны знать / понимать:*

- назначение подпрограмм;
- отличия процедур и функций;
- понятие формальных и фактических параметров.

*Учащиеся должны уметь:*

- объявлять процедуры и функции в программе Pascal;
- вызывать подпрограммы из основной программы.

**Практическая работа:** создание программ с использованием различных видов подпрограмм.

### **Тема 2.4. Использование графического модуля**

*Учащиеся должны знать / понимать:*

- понятие модуля;
- назначение и возможности графического модуля.

*Учащиеся должны уметь:*

- подключить графический модуль;
- инициализировать графический режим;

- использовать графические примитивы.

**Практическая работа:** создание графических программ.

### **Раздел 3. Основные алгоритмические конструкции**

#### **Тема 3.1. Циклы**

*Учащиеся должны знать / понимать:*

- понятие и назначение цикла;
- цикл со счётчиком;
- циклы с условием;
- понятие генератора случайных чисел;
- понятие символьного типа;
- назначение и возможности модуля **CRT**;
- понятие кода клавиши, расширенного кода клавиши.

*Учащиеся должны уметь:*

- использовать все виды циклов для повторения блока действий в программе;
- определять оптимальный вид оператора цикла для решения поставленной задачи;
- использовать генератор случайных чисел;
- использовать символьные переменные и константы;
- принимать коды и расширенные коды клавиш: символьных и служебных.

**Практическая работа:** создание программ, использующих разные виды циклов.

#### **Тема 3.2. Условный оператор. Оператор выбора**

*Учащиеся должны знать / понимать:*

- понятие и назначение условного оператора;
- назначение оператора выбора;
- алгоритм поиска максимального/минимального элемента;
- логические выражения;
- способы тестирования программ.

*Учащиеся должны уметь:*

- использовать условный оператор, оператор выбора при составлении программ;
- осуществлять выбор типа условного оператора/оператора выбора для оптимального решения поставленной задачи;
- составлять сложные логические выражения;
- использовать алгоритм поиска максимального/минимального элемента последовательности;
- составлять тестовую таблицу, тестировать готовую программу.
- **Практическая работа:** создание программ, использующих алгоритмы ветвления.

#### **Тема 3.3. Средства отладки программ**

*Учащиеся должны знать / понимать:*

- механизм отладки;
- возможности отслеживания значений переменных;
- способы пошагового выполнения программы;
- метод дихотомии.

*Учащиеся должны уметь:*

- вывести в окно отладки имена переменных;
- произвести пошаговое выполнение программы;
- тестировать программу, выявлять и исправлять ошибки;
- находить корни произвольных уравнений методом дихотомии.

**Практическая работа:** отладка программ, поиск логических ошибок.

#### **Тема 3.4. Константы**

*Учащиеся должны знать / понимать:*

- понятие константы;
- принципы преобразования экранных координат.

*Учащиеся должны уметь:*

- объявить константу, использовать её значение в программе;
- строить график произвольной функции в заданном масштабе и расположении.

**Практическая работа:** создание программы для построения графика функции.

### Тема 3.5. Компьютерная анимация

*Учащиеся должны знать / понимать:*

- принципы компьютерной анимации;
- назначение и отличия функций **ReadKey** и **KeyPressed**.

*Учащиеся должны уметь:*

- реализовать несложное движение объектов на экране, в том числе и с переменной формой.

**Практическая работа:** создание программ, реализующих передвижение объекта по экрану.

## Раздел 4. Структурированные типы данных

### Тема 4.1. Массивы

*Учащиеся должны знать / понимать:*

- понятие массива;
- понятие двумерного массива как массива массивов;
- способы поиска максимального/минимального элемента массива;
- способы сортировки — сортировка выбором и пузырьковая.

*Учащиеся должны уметь:*

- объявлять одномерные и двумерные массивы;
- использовать массивы для хранения данных в программе;
- осуществлять поиск максимального/минимального элемента в одномерном массиве;
- производить сортировку одномерного массива одним из двух способов: сортировкой выбором или пузырьковой сортировкой.
- **Практическая работа:** создание и обработка массивов.

### Тема 4.2. Типизированные константы

*Учащиеся должны знать / понимать:*

- понятие типизированной константы;
- область применения типизированных констант;
- отличие типизированных констант от переменных и от констант;
- принципы передвижения рисованного объекта по экрану без следа.

*Учащиеся должны уметь:*

- объявлять типизированные константы;
- использовать массив-буфер для сохранения области экрана при передвижении объекта по экрану.

**Практическая работа:** создание программы, реализующей передвижение объекта по экрану

### Тема 4.3. Строковый тип данных

*Учащиеся должны знать / понимать:*

- строковый тип данных;
- понятие строк как массива символов;
- допустимые действия над строковыми данными.

*Учащиеся должны уметь:*

- объявлять и использовать в программе величины строкового типа;
- использовать стандартные процедуры и функции обработки строк.
- Строковый тип данных. Строки как массив символов. Стандартные строковые процедуры и функции.

**Практическая работа:** обработка данных строкового типа.

### Тема 4.4. Записи

*Учащиеся должны знать / понимать:*

- понятие типа «запись»;
- область применения и понятие пользовательского типа «запись»;

*Учащиеся должны уметь:*

- создавать пользовательский тип;
- использовать записи для хранения базы данных.

**Практическая работа:** Заполнение «базы данных» с использованием типа **record**.

**Тема 4.5. Файловый тип данных**

*Учащиеся должны знать / понимать:*

- понятие файлового типа;
- отличия и область применения типизированных и текстовых файловых типов;
- порядок работы с данными файлового типа.

*Учащиеся должны уметь:*

- определить оптимальный тип файловых данных для решения конкретной задачи;
- ассоциировать файловую переменную с файлом на диске;
- открыть файл для чтения или записи; записать/прочитать информацию из файла.

**Практическая работа:** создание программ, позволяющих хранить данные на диске.

**Тематический план кружка**

Наименование разделов и тем	Количество часов	Дата проведения	
		план	факт
<b>Раздел 1. Основы алгоритмизации</b>	<b>4</b>		
Техника безопасности и организация рабочего места. Введение в Паскаль.	1		
Понятие алгоритма	1		
Переменные	3		
Организация ввода-вывода. Оператор присваивания.	4		
<b>Раздел 2. Знакомство с Pascal</b>	<b>6</b>		
• Структура программы на языке Pascal	1		
• Числовые типы данных	2		
• Подпрограммы	1		
• Использование графического модуля	2		
<b>Раздел 3. Основные алгоритмические конструкции</b>	<b>11</b>		
• Циклы	3		
• Условный оператор. Оператор выбора	3		
• Средства отладки программ	2		
• Константы	1		
• Компьютерная анимация	2		
<b>Раздел 4. Структурированные типы данных</b>	<b>13</b>		
• Массивы	6		
• Типизированные константы	1		
• Строковый тип данных	2		
• Записи	1		
• Файловый тип данных	3		
Резерв времени	1		
<b>ВСЕГО</b>	<b>34</b>		

**УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ**

Материально-техническое оснащение

Компьютерное оборудование:

- моноблок – 15 шт.
- маршрутизатор – 1 шт.

Презентационное оборудование:

- проектор – 1 шт.
- 

Методическое обеспечение реализации программы

Методическое обеспечение программы включает приёмы и методы организации образовательного процесса, дидактические материалы, техническое оснащение занятий.

Виды дидактических материалов, используемые при реализации программы:

Для обеспечения наглядности и доступности изучаемого материала педагог использует наглядные пособия следующих видов:

- схематические или символические (оформленные стенды и планшеты, таблицы, схемы, рисунки, графики, плакаты, диаграммы, чертежи, шаблоны и т.п.);
- картинные (иллюстрации, слайды, фотоматериалы и др.);
- звуковые (аудиозаписи);
- смешанные (видеозаписи, учебные кинофильмы и т.д.);
- дидактические пособия (карточки, рабочие тетради, раздаточный материал, вопросы и задания для опроса, тесты, практические задания, упражнения и др.);
- компьютерные программы в электронном виде (компьютеры с программами, CD, флешносители);
- учебные пособия, журналы, книги, Интернет-ресурсы.

При реализации программы с целью повышения качества и эффективности процесса обучения применяются современные эффективные технологии обучения, ориентированные не на накопление знаний, а на организацию активной деятельности обучающихся:

- технологии проектной деятельности;
- компьютерные (информационные) технологии;
- технологии учебно-игровой деятельности (моделирование);
- технологии коммуникативно-диалоговой деятельности;
- модульные технологии;
- квест-технологии;
- технологии лично-ориентированного обучения;
- кейс-технологии.

Информационные технологии используются в различных видах деятельности:

- при подготовке и проведении занятий;
- для создания авторских мультимедийных презентаций;
- в рамках индивидуальной и групповой проектной деятельности;
- для самостоятельной работы;
- для накопления демонстрационных материалов к занятиям (видеоматериалы, таблицы, презентации, карты);

Одним из основных методов является метод проектного обучения, так как он является неотъемлемой частью учебного процесса. Исходный лозунг основателей системы проектного обучения – «Все из жизни, все для жизни». Обучение строится на активной основе, через практическую деятельность ученика, ориентируясь на его личный интерес и практическую востребованность полученных знаний в дальнейшей жизни, обучающийся имеет возможность через проектную деятельность освоить получаемые знания. Проекты представляются в виде готовых программ, презентаций проектов, научных докладов, моделей, демонстрации видеофильма. Достоинствами проектной деятельности являются:

- Уметь работать в коллективе;
- Брать ответственность за выбор решения на себя;



- Разделять ответственность с другими;
- Предоставлять ребенку свободу выбора темы, методов работы;
- Понимание каждым обучающимся важности работы и др.

## **СПИСОК ЛИТЕРАТУРЫ**

Список литературы, используемой педагогом

Основная:

1. Клейнберг Дж. Алгоритмы: разработка и применение. СПб: Питер, 2016. - 800 с.
2. Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017. – 284 с.: ил.
3. Бхаргава А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих. - СПб.: Питер, 2017. - 288 с.

Дополнительная:

1. Луц М. Изучаем Python. СПб: Симво-плюс, 2011. - 1280 с.
2. Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. М: Ямб, 2012. – 520 с.
4. Златопольский Д.М. Сборник задач по программированию. СПб: БХВ-Петербург, 2011. - 295 с.