

Муниципальное бюджетное общеобразовательное учреждение  
г. Мурманска  
средняя общеобразовательная школа № 31

**Утверждено**

Директор С.А. Багурина  
Приказ № 131/ 3 от 29 августа 2014г.

**Рабочая программа по курсу**  
**Программирование повышенной сложности на языке Паскаль**  
**10 класс**

уровень – профильный (ресурсный центр)

Количество часов по учебному плану - 60

часов в неделю - 2

Программу разработала: Тоболева Е.А.,  
учитель информатики МБОУ СОШ № 31

Программа рассмотрена на заседании  
МО учителей математики и информатики  
МБОУ СОШ № 31  
Протокол № 1 от 29 августа 2014 г.  
Рук. МО Иванова Т.И.

Программа рассмотрена на методическом  
совете МБОУ СОШ № 31  
Протокол № 1 от 29 августа 2014 г.

**Мурманск**  
**2014 г.**

## **Пояснительная записка.**

Развитие школьной информатики характеризуется уменьшением удельного веса вопросов, связанных с алгоритмизацией и программированием, и увеличением доли «пользовательского» направления в изучении предмета. Это связано с поставками в школы новых компьютеров, на которых возможна установка современных офисных программ (и, следовательно, их изучение, что не было возможным ранее), уменьшением числа учителей информатики — бывших инженеров и программистов и с другими причинами.

Преподавание программирования в школе остается одной из важных и актуальных тем на сегодняшний день. Исторически сложилось так, что программирование возникло и развивалось как процедурное программирование, которое предполагает, что основой программы является алгоритм, процедура обработки данных.

Нынешнее положение курса алгоритмики и программирования в школе можно назвать незавидным. В настоящее время изучение курса информатики тесно связывают с работой на персональном компьютере. При этом бытует мнение, что в этом курсе школьникам достаточно освоить некоторые наиболее популярные компьютерные технологии, а программированию учить школьников не к чему, ведь качественные программы могут создавать только профессионалы могучих компьютерных фирм.

Это мнение ошибочное. Можно провести аналогию: хотя настоящие произведения искусства создаются в большинстве своём профессиональными музыкантами, композиторами и художниками, это не мешает всем школьникам без исключения изучать музыку и основы изобразительного искусства.

## **Цели и задачи курса**

Многие проблемы в преподавании информатики связаны с нечётким целеполаганием. Основной целью изучения образовательной области "Информатика" считается подготовка школьников к практической деятельности, а развитие мышления и формирование основ научного мировоззрения как бы отступают на второй план.

Изучение программирования позволяет внести свой вклад в достижение этих целей. В первую очередь следует рассматривать программирование как средство развития мышления школьника. Профессиональной подготовкой занимаются специализированные учреждения. Только при самостоятельном решении задач можно говорить о развитии у школьников способности принятия решения и ответственности за его последствия.

Другой важной целью является "понимание единства информационных принципов строения и функционирования самоуправляемых систем различной природы", процессов управления в природе, технике, социуме. При правильном подборе учебных задач эта цель не кажется надуманно-недостижимой.

Но "правильный подбор" ещё не гарантирует достижения результата. Важна и форма организации учебной деятельности, и методика преподавания. Вот здесь и открываются широкие возможности реализации стратегической линии образования на развитие личности, внедрения эвристических методов в образовательный процесс. Другими словами, можно отказаться от традиционного построения курса, когда учащимся сообщается большое количество готовых знаний, в надежде, что немногим "сильным" ученикам удастся самостоятельно выстроить закономерности и постигнуть общие принципы изучаемой дисциплины. Это по силам будет немногим. Гораздо труднее для учителя инициировать на

занятиях самостоятельную познавательную деятельность учащихся, заставляя их постоянно задумываться "над природой вещей", сообщив лишь небольшое количество исходной информации. Именно эта задача гораздо важнее "зубрёжки" и имеет положительный результат в долгосрочной перспективе.

На этапе становления информатики казалась вполне логичной мысль о том, что, выучив язык, учащиеся начнут применять его в своей повседневной учебной и исследовательской деятельности. Сейчас эта ниша занята программными пакетами, относящимися к категории новых информационных технологий. Таким образом, идея практического применения полученных знаний по отношению к программированию стала неактуальной. Это означает лишь одно: программирование напрямую не связано с дальнейшей практической деятельностью будущего гражданина и в рамках этого курса не имеет смысла говорить о третьей цели, провозглашённой проектом стандарта — подготовке школьников к практической деятельности. Здесь она может упоминаться разве что в контексте формирования элементарных навыков работы на ПК.

В тоже время развитие алгоритмического стиля мышления школьников является важной задачей школьной информатики.

Изучение основ программирования связано с развитием целого ряда таких умений и навыков, которые носят общеинтеллектуальный характер и формирование которых – одна из приоритетных задач современной школы. Изучение программирования развивает мышление школьников, способствует формированию у них многих приемов умственной деятельности. Здесь роль информатики сродни роли математики в школьном образовании. Поэтому не использовать действительно большие возможности программирования для развития мышления школьников, формирования многих общеучебных, общеинтеллектуальных умений и навыков было бы, наверное, неправильно.

Изучая программирование на Паскале, учащиеся прочнее усваивают основы алгоритмизации, приобщаются к алгоритмической культуре, познают азы профессии программиста.

Изучение основ программирования в курсе информатике занимает особое место. Всегда этому разделу отводилось достаточно большое внимание, наработано большое количество материала и к настоящему времени уже можно выделить следующие проблемы в подготовке учащихся:

1. Недостаточная математическая подготовка;
2. Неумение формулировать свои мысли;
3. Низкая культура алгоритмического мышления;
4. Неумение самостоятельно работать с учебным материалом.

Это небольшая, но одна из основных частей того наследства, с которым приходит ученик к учителю информатики на уроки программирования. Что остается делать, кроме как разгребать этот воз ученических проблем? В том-то и прелесть программирования, что в его рамках можно достаточно просто и эффективно закрыть те пробелы в знаниях учеников, которые ярчайше проявляются на уроках программирования.

Таким образом, тот материал который изучается по учебной программе, обязан быть насыщен учителем элементарным повторением математических понятий, построением интуитивно понятных ассоциаций и многими другими методическими приемами, но конечно же к месту и в контексте изучаемого языка программирования.

Не следует забывать о главном: лучше меньше, но лучше; лучше заострить внимание ученика на чем-то главном, существенном, фундаментальном и отработать это, чем расплыться на все многообразие существующих алгоритмов и приемов программирования.

Курс ориентирован на учащихся 10-х классов средних школ, имеющих предшествующую компьютерную подготовку. Курс рассчитан на 1 год по 2 академических часа в неделю.

Целью обучения курса "Программирование повышенной сложности на языке Паскаль" является предпрофессиональная подготовка учащихся.

В результате обучения **учащиеся должны знать:**

- Структуру Паскаль-программы.
- Иерархию типов данных, стандартные функции.
- Выражения: понятие, типы, правила записи.
- Способы описания и использования переменных и констант.
- Основные операторы языка Паскаль (присваивания, вывода, ввода, условный, выбора, циклы с параметром, с пред- и постусловиями): общий вид, ограничения на параметры.
- Понятие подпрограммы, виды подпрограмм (процедуры и функции), способы описания и использования в программах. Понятия локальных и глобальных переменных, параметр-значений и параметр-переменных.
- Понятия массива, размерности массива, способы задания и методы сортировки элементов массива.
- Строковый тип данных и основные функции для работы со строками.
- Понятие множества, действия над ними.
- Основные графические процедуры и функции.
- Комбинированные, файловые и ссылочные типы.

**Учащиеся должны уметь:**

- Записывать арифметические выражения по правилам языка Паскаль.
- Описывать переменные и константы, задавать их значения, использовать в арифметических выражениях.
- Организовывать вывод на экран текста, значений переменных, констант, и выражений.
- Организовывать ввод значений переменных.
- Организовывать ветвления в программах.
- Реализовывать циклические алгоритмы.
- Описывать и использовать в основной программе подпрограммы-процедуры и подпрограммы-функции.
- Описывать, задавать и преобразовывать элементы одномерного и двумерного массивов.
- Задавать и преобразовывать элементы строки.
- Задавать и преобразовывать элементы множества.
- Использовать в программе основные графические процедуры и функции.
- Использовать в программе данные комбинированных, файловых и ссылочных типов.

## ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ КУРСА

№	Тема	Часы
1.	Чтение блок-схем	2
2.	Грамотное составление блок-схем	2
3.	Структура программы на языке Паскаль. Рекомендации по стилю записи программы, использование комментариев. Алфавит языка. Интегрированная среда TP7.	1
4.	Типы данных: целый и вещественный, логический и символьный. Константы. Переменные. Организация ввода-вывода. Операции div и mod	2
5.	Стандартные функции. Арифметические выражения. Правила записи арифметических выражений. Оператор присваивания. Решение задач.	1
6.	Организация ветвлений в программах. Основные понятия математической логики. Условный оператор. Оператор выбора (CASE). Оператор перехода.	2
7.	Понятие подпрограммы. Подпрограмма-процедура. Подпрограмма-функция.	2
8.	Программирование циклических алгоритмов, виды циклов. Операторы организации циклов.	2
9.	Перечислимые и ограниченные типы данных.	2
10.	Одномерные массивы: описание и задание элементов, действия над ними.	2
11.	Понятие двумерного массива. Действия над элементами массива. Вложенные циклы. Обработка элементов двумерных массивов.	8
12.	Строковый тип данных: нахождение, замена, вывод на экран элементов строк, подчиненных определенным условиям с использованием стандартных функций.	6
13.	Понятие множества. Использование множеств при обработке текстов и массивов.	4
14.	Основы машинной графики с использованием языка Паскаль. Основные графические процедуры и функции, константы, переменные.	6
15.	Работа в текстовом режиме (создание окон). Система координат и построение графиков функций.	8
16.	Комбинированные типы (записи). Оператор присоединения.	6
17.	Файловые типы	4

### Рекомендуемая литература

1. Ушаков Д.М., Юркова Т.А. Паскаль для школьников. - СПб.: Питер, 2005.
2. Культин Н.Б. Turbo Pascal в задачах и примерах. – СПб.:БХВ-Петербург,2008.
3. Мендель А.В., Коллегаева Е.М. Информатика. 9-11 классы: подготовка учащихся к олимпиадам. – Волгоград: Учитель,2009.
4. Горяинов В.С., Карайчев Г.В., Коваленко М.И. Школьные олимпиады: физика, математика, информатика. – Ростов н/Д: Феникс, 2007.
5. Довгаль С. И., Литвинов Б. Ю., Сбитнев А. И. Персональные ЭВМ: Турбо Паскаль 6.0, объектное программирование, локальные сети (учебное пособие). Киев: Информсистема сервис, 1993.
6. Лунин С. Н. Turbo Pascal 7.0: Самоучитель для начинающих. М: Диалог-МИФИ, 2004.
7. Турбо Паскаль 7.0: Для пользователя. Киев: Торгово-издательское бюро ВНУ, 1996.
8. Фаронов В. В. Основы Турбо Паскаля (6.0). М.: МВТУ-Фесто Дидактик