

Негосударственное образовательное учреждение  
«Открытый молодёжный университет»

**КОМПЛЕКСНАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА  
«ШКОЛЬНЫЙ УНИВЕРСИТЕТ»**

**УТВЕРЖДАЮ**

Директор  
Негосударственного  
образовательного учреждения  
«Открытый молодёжный университет»

\_\_\_\_\_ И. В. Дмитриев

«\_\_» \_\_\_\_\_ 2011 г.

**Основы алгоритмизации  
и программирование на языке Pascal**

*Учебная программа*

Абдрашитова И. В. Основы алгоритмизации и программирование на языке Pascal: Учебная программа. — Томск: НОУ «Открытый молодёжный университет», 2011. — 15 с.

Анализ профессиональной деятельности людей массовых профессий позволяет сделать вывод о возрастании роли подготовки молодёжи в области информатики и информационных технологий (ИТ-подготовки). Информатика как учебный предмет открывает школьникам для систематического изучения одну из важнейших областей действительности — область информационных процессов в живой природе, обществе и технике. Наличие и значительная роль информационных процессов в системах различной природы определяют востребованность учебных курсов по информатике в различных профилях обучения на старшей ступени школы.

Программа содержит пояснительную записку, в которой указываются место курса в образовательном процессе, конкретные цели и задачи курса, формы учения, излагаются планируемые результаты обучения, приводятся тематическое планирование и основное содержание курса. В разделе «Содержание курса» перечисляются темы курса, требования к знаниям и умениям учащихся, а также темы практических занятий и самостоятельной работы учащихся.

Программа обеспечена учебно-методическим комплектом, который состоит из учебного пособия, рабочей тетради, интерактивного электронного учебника, контрольно-измерительных материалов для проведения текущего контроля.

Учебная программа рассмотрена и рекомендована к изданию на методическом семинаре Негосударственного образовательного учреждения «Открытый молодёжный университет» и Образовательного центра «Школьный университет» 26 января 2011 года.

Редактор И. Ю. Васильева  
Компьютерная вёрстка В. Д. Пяткова

## **Пояснительная записка**

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

Курс «Основы алгоритмизации и программирование на языке Pascal» является базовым курсом предметной области «Информатика» на профильном уровне и базируется на знаниях и умениях школьников, сформированных при изучении общеобразовательного предмета «Информатика и информационно-коммуникационные технологии» основной школы. Курс предназначен для учащихся старших классов информационно-технологического профиля. В то же время курс может быть использован для удовлетворения познавательных интересов школьников, а также для подготовки к Единому государственному экзамену по теме «Алгоритмизация и программирование» учащихся, обучающихся в классах различных профилей.

Программа составлена с учётом стандарта среднего (полного) общего образования по информатике и информационно-коммуникационным технологиям в едином комплексе с другими программами дисциплин старшей школы.

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 основными алгоритмическими конструкциями.

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

Для качественного обучения необходимо, чтобы учащиеся обладали базовыми знаниями по математике на уровне 9 класса и навыками работы в файловой системе ОС Windows.

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

### **Концепция курса**

Ключевой особенностью курса является его направленность на формирование у учащихся навыков составления алгоритма решения поставленной задачи 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 «Основы алгоритмизации и программирование на языке Pascal» изучение основ программирования — это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков. Исключительно велика роль программирования для формирования мышления школьников, общеучебных приёмов и действий, умения строить модели,

самостоятельно находить и составлять алгоритмы 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.

Умение составлять алгоритмы решения и навыки программирования является элементом информационной компетентности — одной из ключевых для современной школы, как необходимое условие подготовки школьников. Особая роль отводится широко представленной в курсе системе рефлексивных заданий. Она направлена 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.

### **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 от записи алгоритмов на языке блок-схем, перевода этих алгоритмов на алгоритмический язык, до разработки программ средствами языка программирования Pascal;
- изучить основные конструкции языка программирования Pascal;
- научиться отлаживать и тестировать программы, делать выводы о работе этих программ.

### **Методы обучения**

Отбор методов обучения обусловлен необходимостью формировать у старшеклассников информационную и коммуникативную компетентности, реализовывать лично-ориентированное обучение,

направлять их на самостоятельное решение разнообразных проблем, развивать исследовательские и творческие способности. Решение данных задач кроется в организации деятельностного подхода к обучению, в проблемном изложении материала учителем, в переходе от репродуктивного вида работ к самостоятельным, поисково-исследовательским видам деятельности.

### **Формы организации учебных занятий**

Организация учебного процесса с использованием учебно-методического комплекта предусматривает наличие двух взаимосвязанных и взаимодополняющих форм:

- *урочная форма*, когда учитель во время урока консультирует учащихся в процессе выполнения ими практических заданий в рабочей тетради или программ на компьютере;
- *внеурочная форма*, когда учащийся вне уроков самостоятельно изучает теоретический материал и выполняет задания.

### **Планируемые результаты курса**

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

- умеют составлять алгоритмы для решения задач из разных областей знаний;
- умеют записывать составленные алгоритмы в разной форме;
- 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.

### **Способы оценивания уровня достижений учащихся**

Предметом диагностики и контроля в курсе «Основы алгоритмизации и программирование на языке Pascal» являются внешние образовательные продукты учащихся (созданные блок-схемы, программы), а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам курса.

Диагностика и контроль — необходимые части учебного процесса, но увеличение их доли в общем объеме времени неизбежно приводит к сокращению времени на изучение материала. Поэтому столь важно извлечение максимума информации на каждом шаге обучения. Контроль и диагностика должны быть действенными. Поэтому необходимо анализировать результаты проверки и принимать меры по коррекции образовательного процесса. От этого зависит, станут ли способы оценивания уровня достижений учащихся результативными.

Качество внешней образовательной продукции желательно оценивать по следующим параметрам:

- алгоритм должен быть оптимальным по его восприятию и максимально простым в реализации на языке программирования;
- программа должна выполнять поставленные задачи;
- по степени «читаемости кода» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.).

Созданными внешними образовательными продуктами учащиеся могут пополнять собственные портфолио.

Проверка достигаемых школьниками результатов проводится в следующих формах:

- текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
- текущая диагностика и оценка учителем деятельности школьников в виде трёх контрольных работ по следующим темам: «Алгоритмы»; «Типы данных. Основные алгоритмические конструкции»; «Операторы цикла».

Итоговый контроль осуществляется в конце курса в форме дифференцированного зачёта по итогам трёх контрольных работ.

### **Состав учебно-методического комплекта**

Программа курса обеспечивается учебным пособием «Основы алгоритмизации и программирование на языке Pascal», рабочей тетрадью «Основы алгоритмизации и программирование на языке Pascal», интерактивным электронным учебником «Pascal — это просто!», контрольно-измерительными материалами для проведения текущего контроля.

*Аппаратное обеспечение:*

1. Процессор не ниже Pentium III.
2. Оперативная память не меньше 256 Мб.
3. Дисковое пространство не меньше 500 Мб.
4. Монитор с 24-битной видеокарткой.
5. Разрешение монитора не ниже 1024×768.

*Программное обеспечение:*

1. Операционная система: Windows XP (или выше).
2. Одна из сред разработки:  
Borland Pascal 7.0,  
Turbo Pascal 7.0,  
Free Pascal 2.0 (или выше).

## Тематический план курса

Наименование разделов и тем	Количество часов	
	Всего	Практич. занятия
<b>Введение</b>	<b>1</b>	
<b>Раздел 1. Основы алгоритмизации</b>	<b>10</b>	<b>5</b>
1.1. Понятие алгоритма. Свойства алгоритмов	2	
1.2. Способы описания алгоритмов	2	
1.3. Основные алгоритмические конструкции	5	5
Контрольная работа № 1	1	
<b>Раздел 2. Основы языка программирования Pascal</b>	<b>21</b>	<b>8</b>
2.1. Структура программы на языке Pascal. Типы данных	3	1
2.2. Логический тип данных. Условный оператор. Составной оператор	5	3
2.3. Символьный тип данных. Оператор выбора варианта	3	
2.4. Операторы цикла с условием	4	4
2.5. Оператор цикла с параметром	4	
Контрольная работа № 2	2	
<b>Раздел 3. Сложные типы данных</b>	<b>13</b>	<b>4</b>
3.1. Одномерные массивы	4	2
3.2. Многомерные массивы	4	2
3.3. Строковый тип данных	1	
3.4. Множественный тип данных	2	
Контрольная работа № 3	2	
<b>Раздел 4. Шаги к профессионализму</b>	<b>23</b>	<b>8</b>
4.1. Комбинированный тип данных. Оператор присоединения	4	
4.2. Подпрограммы	4	4
4.3. Работа с файлами	4	
4.4. Построение графических изображений средствами языка Pascal	4	
4.5. Подготовка к ЕГЭ	4	4
4.6. Ссылочные типы. Динамические переменные	3	
<b>Резерв времени</b>	<b>2</b>	
<b>ВСЕГО</b>	<b>70</b>	<b>25</b>

# Содержание курса

## Раздел 1. Основы алгоритмизации

### Тема 1.1. Понятие алгоритма. Свойства алгоритмов

### Тема 1.2. Способы описания алгоритмов

### Тема 1.3. Основные алгоритмические конструкции

*Учащиеся должны знать / понимать:*

- понятие алгоритма;
- свойства алгоритма;
- понятие исполнителя;
- способы описания алгоритмов;
- типы алгоритмов;
- основные алгоритмические конструкции;
- язык блок-схем.

*Учащиеся должны уметь:*

- составлять алгоритмы для исполнителя;
- записывать алгоритм разными способами;
- определять исполнителя алгоритма;
- использовать при решении задач основные алгоритмические конструкции.

Алгоритмы. Способы записи алгоритма. Исполнители алгоритмов. Типы алгоритмов: линейные, разветвляющиеся, циклические. Определение и свойства алгоритма.

#### **Практическая работа:**

- глава 1 учебного пособия;
- задания рабочей тетради;
- урок 1 «Линейные алгоритмы» интерактивного электронного учебника;
- урок 2 «Разветвляющиеся алгоритмы» интерактивного электронного учебника;
- урок 3 «Циклические алгоритмы: циклы с условием» интерактивного электронного учебника;
- урок 4 «Циклические алгоритмы: цикл с параметром» интерактивного электронного учебника;
- урок 5 «Подготовка к контрольной работе» интерактивного электронного учебника.

#### **Контрольная работа № 1.**

## Раздел 2. Основы языка программирования Pascal

### Тема 2.1. Структура программы на языке Pascal. Типы данных

*Учащиеся должны знать / понимать:*

- назначение и основные команды среды разработки;
- общую структуру программы;
- понятие переменной;
- понятие константы;
- оператор присваивания;
- назначение и виды команд ввода-вывода;
- понятие типа данных.

*Учащиеся должны уметь:*

- устанавливать среду программирования Borland / Turbo Pascal;
- использовать интерфейс среды программирования;
- использовать команды редактора;
- определять тип числовых данных;
- объявлять необходимые переменные;
- объявить константу, использовать её значение в программе;
- записывать арифметические выражения.
- составлять и запускать программы;
- организовывать ввод-вывод данных.

Язык программирования Pascal и его характерные особенности. Структура программы на языке Pascal. Типы данных языка программирования Pascal. Целые и вещественные типы. Значения. Оператор присваивания. Операции, допустимые с переменными и значениями целого и вещественного типа. Простейшая программа. Элементы языка Pascal. Создание и исполнение программ в среде разработки. Операторы ввода **Read** и **ReadLn**, операторы вывода **Write** и **WriteLn**.

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

- задания рабочей тетради;
- урок 6 «Программа на Pascal. Команды ввода-вывода» интерактивного электронного учебника.

### Тема 2.2. Логический тип данных. Условный оператор. Составной оператор

*Учащиеся должны знать / понимать:*

- понятие простых типов данных, диапазон их значений и операции над ними;
- понятие логического типа данных, логического выражения;
- понятие и назначение условного оператора;
- понятие и назначение составного оператора;

- логические операторы **OR, AND, NOT**;
- операторные скобки **BEGIN...END**.

*Учащиеся должны уметь:*

- использовать условный оператор при составлении программ;
- осуществлять выбор типа условного оператора для оптимального решения поставленной задачи;
- составлять сложные логические выражения;
- составлять тестовую таблицу, тестировать готовую программу.

Условный оператор. Полная и неполная формы условного оператора. Составной оператор.

Тестирование готовой программы.

**Практическая работа:** создание программ, использующих алгоритмы ветвления:

- задания рабочей тетради;
- урок 7 «Условный оператор» интерактивного электронного учебника.

### **Тема 2.3. Символьный тип данных. Оператор выбора варианта**

*Учащиеся должны знать / понимать:*

- понятие символьного типа;
- назначение и возможности модуля **CRT**;
- понятие кода клавиши, расширенного кода клавиши.

*Учащиеся должны уметь:*

- использовать символьные переменные и константы;
- принимать коды и расширенные коды клавиш: символьных и служебных.

Символьный тип данных. Использование возможностей модуля CRT для приёма и обработки сигналов клавиш.

**Практическая работа:** задания рабочей тетради по созданию программ с использованием символьного типа данных.

### **Тема 2.4. Операторы цикла с условием**

### **Тема 2.5. Операторы цикла с параметром**

*Учащиеся должны знать / понимать:*

- понятие и назначение цикла;
- цикл со счётчиком (параметром);
- циклы с условием;
- понятие генератора случайных чисел.

*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;

Цикл. Цикл со счётчиком. Цикл с предусловием, цикл с постусловием. Генератор случайных чисел.

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

- задания рабочей тетради;
- урок 8 «Операторы циклов» интерактивного электронного учебника.

## **Контрольная работа № 2.**

### **Раздел 3. Сложные типы данных**

#### **Тема 3.1. Одномерные массивы**

#### **Тема 3.2. Многомерные массивы**

*Учащиеся должны знать / понимать:*

- понятие одномерного 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 методом «пузырька».

Массив. Одномерный массив. Двумерный массив. Объявление массивов. Обработка массивов: поиск элемента по заданным признакам, заполнение массива, вывод массива на экран.

Сортировка массива. Пузырьковая сортировка.

**Практическая работа:** создание и обработка массивов:

- задания рабочей тетради;
- урок 9 «Массивы и работа с ними» интерактивного электронного учебника.

### Тема 3.3. 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:** задания рабочей тетради по обработке данных строкового типа.

### Тема 3.4. Множественный тип данных

*Учащиеся должны знать / понимать:*

- понятие множества;
- способы описания множества;
- операторы работы с множествами.

*Учащиеся должны уметь:*

- описывать множества;
- задавать константы-множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.

Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству.

**Практическая работа:** задания рабочей тетради по использованию множественного типа данных.

### Контрольная работа № 3.

## Раздел 4. Шаги к профессионализму

### Тема 4.1. Комбинированный тип данных. Оператор присоединения

*Учащиеся должны знать / понимать:*

- понятие записи;
- назначение записи;

- способ описания записи;
- обращение к полю записи;
- оператор присоединения.

*Учащиеся должны уметь:*

- разрабатывать программы, использующие записи;
- использовать оператор присоединения.

Комбинированный тип данных. Записи. Поля записей. Описание записей. Способы обращения к полям записи. Оператор присоединения.

**Практическая работа:** задания рабочей тетради по использованию комбинированного типа данных.

### **Тема 4.2. Подпрограммы**

*Учащиеся должны знать / понимать:*

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

*Учащиеся должны уметь:*

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

Подпрограмма. Процедуры и функции. Параметры, формальные и фактические параметры.

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

- задания рабочей тетради;
- урок 10 «Функции и процедуры» интерактивного электронного учебника.

### **Тема 4.3. Работа с файлами**

*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.

Файловый тип данных. Типизированные файлы. Текстовые файлы.

**Практическая работа:** задания рабочей тетради по созданию программ, использующих файловый тип данных.

#### **Тема 4.4. Построение графических изображений средствами языка Pascal**

*Учащиеся должны знать / понимать:*

- понятие модуля;
- назначение и возможности графического модуля.

*Учащиеся должны уметь:*

- подключить графический модуль;
- инициализировать графический режим;
- использовать графические примитивы.

Модули. Модуль **Graph**, назначение и возможности. Графический экран (режим). Основные графические примитивы. Управление цветом. Штриховка.

**Практическая работа:** создание графических программ:

- задания рабочей тетради;
- урок 11 «Графика на Pascal» интерактивного электронного учебника.

#### **Тема 4.5. Подготовка к ЕГЭ**

*Учащиеся должны знать / понимать:*

- понятие типа данных;
- понятие «основные алгоритмические конструкции»;
- область применения различных типов данных;
- особенности применения процедур и функций.

*Учащиеся должны уметь:*

- создавать программы для решения задач части С.

**Практическая работа:**

- урок 11 «Массивы: поиск элементов по заданному условию» интерактивного электронного учебника;
- урок 12 «Строки: обработка данных» интерактивного электронного учебника.

#### **Тема 4.6. Ссылочные типы. Динамические переменные**

*Учащиеся должны знать / понимать:*

- что такое динамическая память и как с ней работать;

- иметь представление о ссылочных типах и указателях;
- иметь представление о списках и работе с ними.

**Цель:** познакомиться с понятиями «динамическая память», «ссылочный тип», «указатель», «списки».

**Практическая работа:** задания рабочей тетради, параграфы 9.1–9.2.

### **Учебно-методические материалы**

1. Абдрашитова И. В. Основы алгоритмизации и программирование на языке Pascal: Учебное пособие.
2. Абдрашитова И. В. Основы алгоритмизации и программирование на языке Pascal: Рабочая тетрадь.
3. Pascal — это просто!: Интерактивный электронный учебник.
4. Абдрашитова И. В. Основы алгоритмизации и программирование на языке Pascal: Учебная программа.
5. Абдрашитова И. В. Основы алгоритмизации и программирование на языке Pascal: Методические рекомендации.
6. Абдрашитова И. В. Основы алгоритмизации и программирование на языке Pascal: Задания для проведения контрольной работы № 1 «Алгоритмы».
7. Абдрашитова И. В. Основы алгоритмизации и программирование на языке Pascal: Задания для проведения контрольной работы № 2 «Типы данных. Основные алгоритмические конструкции».
8. Абдрашитова И. В. Основы алгоритмизации и программирование на языке Pascal: Задания для проведения контрольной работы № 3 «Операторы цикла».