

# ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ ОБЪЕКТ PASCAL

## ВВЕДЕНИЕ

Традиционная технология программирования складывалась в условиях, когда основными потребителями программ были научные учреждения, вычислительные ресурсы были ограничены, а проблемы сопровождения, по существу, неизвестны. Основными критериями качества программы считалась ее узко понимаемая эффективность и компактность. Со временем сложность программ возросла настолько, что на их разработку уходили годы труда большого коллектива, а в результате программы появлялись с большим опозданием и содержали много ошибок.

Кризис программного обеспечения привел к необходимости создания нового способа разработки программ, который снижал бы общие затраты на протяжении всего цикла – от замысла до завершения эксплуатации. Такая технология появилась в начале 70-х годов XX в. и была названа структурным программированием. В его основе лежит сочетание теории программирования и личного опыта высококвалифицированных программистов, а также учет современных требований к программам и промышленного характера их производства.

Структурное программирование – это технология создания программ, позволяющая путем соблюдения определенных правил уменьшить время разработки и количество ошибок, а также облегчить возможность модификации программы. Структурный подход охватывает все стадии разработки проектов: спецификацию, проектирование, собственно программирование и тестирование.

С увеличением объема программы становится все более сложным удерживать в памяти все детали. Естественным способом борьбы со сложностью любой задачи является ее разбиение на части. В Паскале задача может быть разделена на более простые подзадачи с помощью *функций и процедур*, после чего программу можно рассматривать на уровне взаимодействия функций и процедур.

Использование функций и процедур является первым шагом к повышению степени абстракции программы и ведет к упрощению структуры.

Следующим шагом в повышении уровня абстракции программы является группировка функций и процедур в отдельные файлы (модули), компилируемые отдельно. Получившиеся в результате компиляции объектные модули объединяются в исполняемую программу с помощью компоновщика.

Следующий шаг – описание собственных типов данных, позволяющих структурировать и группировать информацию, представляя ее в более естественном виде. Для этого данные объединяются с обрабатывающими их алгоритмами. Данный подход получил название *объектно-ориентированное программирование* (ООП); он позволяет разрабатывать крупные программные системы.

Бурное развитие вычислительной техники, потребность в эффективных средствах разработки программного обеспечения привели к появлению на рынке программ целого ряда систем программирования, ориентированных на быструю разработку приложений, среди которых следует отметить *Microsoft Visual Basic, Borland Delphi, Borland C++ Builder, Microsoft Visual Studio*. В основе систем быстрой разработки лежит технология визуального проектирования и событийно-управляемого программирования, суть которой заключается в том, что среда разработки берет на себя большую часть работы по генерации программного кода, оставляя программисту работу по конструированию диалоговых окон и написания функций обработки событий, возникающих в программе. Понятно, что такие системы резко повышают производительность работы программиста.

Представленная учебная программа позволяет слушателю пройти все технологии программирования на языке *Object Pascal* в их естественном историческом развитии, поскольку каждая технология строится основе предыдущих, либо пропустить какие-то этапы, если слушатель знаком с основами данных технологий.

### **Часть 1. Структурное программирование**

Происхождение языка Паскаль. Алфавит и объекты языка. Структура программы. Типы данных. Функции языка. Операторы ввода и вывода. Оператор присваивания. Организация ветвей вычислительного процесса: операторы условного и безусловного переходов, оператор выбора. Использование логических операций. Организация циклических вычислений. Индексированные переменные (массивы). Простейшие алгоритмы обработки массивов. Динамические массивы. Обработка символьной информации. Структурированные переменные: записи и множества.

### **Часть 2. Модульное программирование**

Структурирование программ: процедуры и функции. Стандартные процедуры и функции. Пользовательские процедуры и функции. Параметры функций и процедур. Передача массивов в качестве параметров. Перегрузка функций.

Графические возможности языка. Работа с файлами. Понятие файла данных. Файловые типы. Этапы работы с файлами данных. Типы доступа к файлам данных. Типизированные файлы. Нетипизированные файлы. Текстовые файлы. Модульное программирование.

Понятие модуля. Стандартные модули. Пользовательский модуль *Unit*.

### **Часть 3. Объектно-ориентированное программирование**

Объектно-ориентированное программирование. Основные понятия ООП. Описание объектов: поля и методы. Раннее и позднее связывание. Динамические объекты. Размещение объектов в модулях.

### **Часть 4. Визуальное проектирование и событийно-управляемое программирование**

Визуальное программирование в среде *Delphi*. Особенности событийно-управляемого программирования в Windows. Инструментальная среда пакета. Особенности объектно-ориентированного программирования в *Delphi*. Базовые компоненты, методы, события. Создание отчуждаемого приложения.

Графика в *Delphi*. Работа с файлами. Мультимедиа. Работа с базами данных. Компонент программиста.