



Автономная некоммерческая организация  
дополнительного профессионального образования  
(АНО ДПО «Инфосфера»)  
Центр профессиональной подготовки  
ИНСТИТУТ ПРОГРАММНЫХ СИСТЕМ

**Рабочая программа дисциплины**  
**«Основы программирования»**

Разработал:  
преподаватель ИПС  
АНО ДПО «Инфосфера»  
Ю.В. Усков

Йошкар-Ола, 2017

# Содержание

Содержание.....	2
1. Цели и задачи курса .....	3
2. Содержание курса .....	5
2.1. Общий план изучения дисциплины .....	5
2.1.1. Часть первая.....	5
2.1.2. Часть вторая.....	5
2.3. Наименование и краткое содержание лекций.....	6
2.3.1. Часть первая.....	6
2.3.2. Часть вторая .....	7
2.5 Наименование и краткое содержание тем лабораторных работ .....	9
2.5.1. Часть первая.....	9
2.5.2. Часть вторая .....	10
3. Учебно-методические материалы .....	11
3.1. Основная литература.....	11
3.2. Дополнительная литература .....	11

# 1. Цели и задачи курса

- 1) Ознакомить студентов с парадигмой структурного программирования.
- 2) Ознакомить студентов с теоретическими основами проектирования, разработки, тестирования и верификации программ.
- 3) Ознакомить студентов с основными возможностями процедурных языков программирования высокого уровня на примере языка программирования Pascal:
  - последовательное выполнение
  - ветвление
  - циклы
  - типы данных
  - блоки
  - переменные
  - область видимости
  - рекурсия
  - подпрограммы
  - передача параметров
  - модули
  - целочисленные вычисления
  - вычисления с плавающей точкой
  - файлы
  - строки
  - массивы
  - указатели
  - связанные структуры данных.

В результате изучения дисциплины студенты должны освоить следующие навыки:

1. Проектировать, разрабатывать, тестировать и отлаживать программы на языке Pascal.
2. Применять существующие алгоритмы по обработке данных для решения конкретных прикладных программ.
3. Разрабатывать собственные алгоритмы обработки данных.

## Планируемые результаты обучения

Осуществляется предварительная самостоятельная или под руководством разработка алгоритмов с использованием графических средств (блок-схемы, UML-диаграммы и др.). Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором

программного обеспечения. Осуществляется решение типовых задач. Полученные результаты представляются руководителю разработки программного обеспечения.

Осуществляется самостоятельное или под руководством оформление программного кода в соответствии с внутренними нормативными документами организации (регламентами, приказами, порядками) и, при необходимости, ГОСТами. Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Осуществляется решение типовых задач. Полученные результаты представляются руководителю разработки программного обеспечения.

Проверка работоспособности программного кода осуществляется на основании функциональных требований и технических спецификаций на программное обеспечение. Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Осуществляется решение типовых задач. Полученные результаты представляются руководителю разработки программного обеспечения.

Разработка тестовых наборов данных предполагает формулирование правил создания, структуры и требований к тестовым наборам данных, подготовка наборов данных, используемых в процессе проверки работоспособности. Решаются задачи с элементами проектирования. Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Полученные результаты представляются руководителю разработки программного обеспечения.

Проверка работоспособности программного обеспечения осуществляется программистом на основании функциональных требований и технических спецификаций на программное обеспечение самостоятельно путем разработки и исполнения сценариев проверки с применением методов и технологий тестирования и ревьюирования кода.

В ходе проверки работоспособности осуществляется анализ нарушений требований к программному обеспечению, принимаются решения и вносятся изменения в программный код. Программист несет ответственность за решение поставленных задач или результат деятельности группы работников. Полученные результаты представляются руководителю разработки программного обеспечения.

Рефакторинг и оптимизация программного кода осуществляется на основании функциональных требований и технических спецификаций на программное обеспечение, в том числе с использованием специализированных программных средств. Программист несет ответственность за решение поставленных задач или результат деятельности группы работников. Полученные результаты представляются руководителю разработки программного обеспечения.

Выполняются самостоятельная разработка процедур сборки модулей и компонент программного обеспечения и верификация выпусков программного продукта. Производится разработка процедур развертывания и обновления программного обеспечения, процедур миграции и преобразования (конвертации) данных и программных интерфейсов с использованием выбранных программных средств, технологий создания открытых систем. Осуществляется решение различных типов задач проектирования программных комплексов различной сложности, выбор способов реализации взаимодействия программных компонент/модулей. Требуется взаимодействие с программистами-разработчиками модулей, архитектором программного обеспечения. Полученные результаты представляются руководителю разработки программного обеспечения.

## 2. Содержание курса

### 2.1. Общий план изучения дисциплины

#### 2.1.1. Часть первая.

Наименование тем	Виды занятий (количество часов)			
	Лекции	Лабор. занятия	Самост. работа	Консультации
Введение в курс	2	2	4	
Введение в CF Pascal	4	4	8	
Разработка программ на CF Pascal	4	4	8	
Условное выполнение	2	2	4	
Текстовые файлы и циклическое выполнение	4	4	8	
Введение в программное исчисление	4	4	8	
Исчисление последовательных программ	4	4	8	
Исчисление условных выражений	4	4	8	
Исчисление циклических выражений	4	4	8	
Итого	32	31	63	

#### 2.1.2. Часть вторая.

Наименование тем	Виды занятий (количество часов)			
	Лекции	Лабор. занятия	Самост. работа	Консультации
Процедуры с параметрами	2	2	4	
Программирование с рекурсией	2	2	4	
Систематическое проектирование и корректность программ	2	2	4	
Абстракция данных, программные модули, очередь	4	4	8	
Порядковые типы данных, целочисленные вычисления	4	4	8	
Константы, типы, передача параметров, функции	2	2	4	
Сложные типы данных: множества, записи, файлы	2	4	8	
Дополнительные средства управления потоком выполнения	2	2	4	
Массивы, стек	4	4	8	
Связанные структуры данных, списки, деревья, сортировка вставкой	6	6	12	
Заключительный обзор курса	2			
Итого	32	31	63	

## 2.3. Наименование и краткое содержание лекций

### 2.3.1. Часть первая

Номер	Тема лекции. Краткое содержание	Часы
1	Введение в курс Развитие программирования обеспечения Систематическое программирование CF Pascal	2
2,3	Введение в CF Pascal Виртуальные машины. Pascal-машина Последовательное выполнение Символьные данные Переменные Оператор присвоения Условное выполнение Условие Оператор IF Оператор BEGIN Оператор WHILE Бесконечное выполнение Поиск максимального среди последовательности символов Аннотированный вывод и эхо ввода	4
4,5	Разработка программ на CF Pascal Введение в синтаксис CFP Синтаксис программ. Нотация Бекуса-Наура Синтаксическое дерево Синтаксис операторов Проектирование и разработка программ Разделы проекта и разработочные программы Фазы проектирования Принципы разработки и проектирования программ Организация программ с подпрограммами (процедурами) Простые процедуры Подсчет символов Фаза разработки Разработка программы параллельно с тестированием Тестирование программы параллельно с разработкой	4
6	Условное выполнение Сортировка с использованием условных выражений Логика Буля Операции OR AND NOT Логические выражения Логические условия CFP Сравнение эффективности различных стратегий сортировки	2
7,8	Текстовые файлы и циклическое выполнение Тип данных TEXT Операции с файлами Копирование файлов Маркеры текстовых файлов Маркер конца линии Маркер конца файла Копирование строк Сортировка методом пузырька	4

9,10	Введение в программное исчисление Символьные строки Конкатенация строк Подстроки Композиция строк Декомпозиция строк Списковые структуры Операции над списками Списочное описание файлов Разрешенные операции и состояния файла Структура множеств Способы описания множеств Операции над множествами Отношения и функции Значение программ Вох-нотации Прямое определение значения программы	4
11, 12	Исчисление последовательных программ Значение частей программы Состояния выполнения Значение заголовка программы и точки Композиция отношений и функций Значение объявлений (деклараций) Значение блока Значение последовательных операторов Оператор присвоения Пустой оператор Оператор BEGIN Оператор WRITE Оператор READ	4
13, 14	Исчисление условных выражений	4
15, 16	Исчисление циклических выражений	4
	<b>Итого</b>	<b>32</b>

### 2.3.2. Часть вторая

Номер	Тема лекции. Краткое содержание	Часы
17	Процедуры с параметрами Процедуры с параметрами Простые процедуры Описание параметров Программа Split Наложение имен Aliasing Процедуры с объявлениями Процедуры обработки строк Область видимости идентификаторов	2
18	Программирование с рекурсией Рекурсивные процедуры в CFP Решение задач с рекурсией Выполнение рекурсивной процедуры Сортировка и реверсирование с помощью рекурсии Исчисление рекурсивных процедур	2

19	Программное исчисление и корректность программ Спецификации Доказательство корректности программ Проектирование программ через спецификации	2
20, 21	Абстракция данных, программные модули, очередь Абстракция данных. Автоматы с памятью Модуль очереди	4
22, 23	Порядковые типы данных, целочисленные вычисления Перечислимые типы Синтаксис и семантика перечислимых типов Ввод-вывод перечислимых типов Логический тип Синтаксис и семантика логических выражений Ввод-вывод логических выражений Целый тип Синтаксис и семантика целых типов Переполнение значений целых типов Тип диапазон Проектирование и анализ задач с порядковыми типами	4
24	Константы, типы, передача параметров, функции Внешние файлы Константы Пользовательские типы данных Параметры значения и параметры-переменные Функции	2
25	Сложные типы данных: множества, записи, файлы Множества Файлы Записи	2
26	Дополнительные средства управления потоком выполнения Оператор CASE Оператор FOR Оператор REPEAT	2
27, 28	Массивы, стек Массивы Синтаксис и семантика для массивов Параметры-массивы Реализация абстрактных типов данных (стек) с помощью массивов	4
29, 30, 31	Связанные структуры данных, списки, деревья, сортировка вставкой Реализация связанных структур с использованием массивов Указатели Реализация связанных структур с использованием указателей Сортировка включением Сортировка с помощью бинарного дерева	6
32	Заключительный обзор курса Технология программирования Методы решения задач Структура и состав языков программирования Средства организации данных Средства организации потока команд Средства организации структуры программы Стандартная библиотека	2
	<b>Итого</b>	<b>32</b>



## 2.5 Наименование и краткое содержание тем лабораторных работ

### 2.5.1. Часть первая

Номер	Наименование лабораторной работы	Часы
1	Лабораторная работа № 1 Вводное занятие. Знакомство со средой Free Pascal. Форматирование кода.	2
2	Лабораторная работа № 2 Последовательное выполнение, условное выполнение, циклы.	2
3	Лабораторная работа № 3 Синтаксические правила, синтаксические деревья.	2
4	Лабораторная работа № 4 Проектирование, разработка, сборка программ	4
5	Лабораторная работа № 5 Сборка программ из разделов проекта, CountChars	2
6	Лабораторная работа № 6 Сортировка 3-строк, IFSort3, MinSort3	2
7	Лабораторная работа № 7 Текстовые файлы, Split, SelectSort	2
8	Лабораторная работа № 8 Текстовые файлы, сортировка пузырьком, структурное тестирование	2
9	Лабораторная работа № 9 Введение в программное исчисление, строки, списки, множества	4
10	Лабораторная работа № 10 Непосредственное определение значений последовательных программ	4
11	Лабораторная работа № 11 Реверсирование строк. Вычисление частного значения программ	4
12	Лабораторная работа № 12 Условные присваивания, значение циклов, удаление пробелов	2
	<b>Итого</b>	<b>31</b>

### 2.5.2. Часть вторая

Номер	Наименование лабораторной работы	Часы
1	Лабораторная работа № 13 Процедуры с параметрами, Split, BubbleSort	2
2	Лабораторная работа № 14 Рекурсивная сортировка и реверсирование	2
3	Лабораторная работа № 15 Формальные спецификации	2
4	Лабораторная работа № 16 Программные модули	4
5	Лабораторная работа № 17 Модуль очереди, автоматы	4
6	Лабораторная работа № 18 Целочисленные вычисления, функции	2
7	Лабораторная работа № 19 Решето Эратосфена, сложные типы данных, сортировка дат	4
8	Лабораторная работа № 20 Множества, псевдографика	4
9	Лабораторная работа № 21 Массивы, множества, шифрование	4
10	Лабораторная работа № 22 Сортировка связанного списка вставкой, сортировка с помощью бинарного дерева	4
	<b>Итого</b>	<b>31</b>

## **3. Учебно-методические материалы**

### ***3.1. Основная литература***

1. Mills, Harlan D. Principles of Computer Programming: A Mathematical Approach —: William C Brown, 1986. — 670 с. ISBN 0205079962.
2. Фаронов, В.В. Turbo Pascal 7.0. Начальный курс —: Нолидж, 2000. ISBN 5892510549.

### ***3.2. Дополнительная литература***

1. Кнут, Дональд Э. Искусство программирования. Том 1. Основные алгоритмы —: Вильямс, 2010. — 720с. ISBN 9785845900807
2. Бентли, Джон Жемчужины программирования —: Питер, 2002. — 272с. ISBN 5318007155
3. Фаронов В.В. TurboPascal 7.0: Практика программирования: Учебное пособие Изд. 7-е, перераб. —: Нолидж, 2001. — 416с. ISBN 5892510557