



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ И РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего образования  
«Дальневосточный федеральный университет»  
(ДФУ)

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

«СОГЛАСОВАНО»  
Руководитель ОП

А.И. Сухомлинов

«УТВЕРЖДАЮ»  
Заведующий кафедрой  
«Информационные системы управления»

А.И. Сухомлинов

« 13 » февраля 2020 г.



**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**  
Алгоритмизация и программирование  
**Направление подготовки 09.03.03 Прикладная информатика**  
Прикладная информатика в управлении предприятием  
**Форма подготовки очная**

курс 1 семестр 1, 2  
лекции 70 час.  
практические занятия 0 час.  
лабораторные работы 122 час.  
в том числе с использованием МАО лек. 0/пр./лаб. 72 час.  
всего часов аудиторной нагрузки 192 час.  
в том числе с использованием МАО 72 час.  
самостоятельная работа 384 час.  
в том числе на подготовку к экзамену 90 час.  
контрольные работы (количество) не предусмотрены  
курсовая работа / курсовой проект 2 семестр  
зачет не предусмотрен  
экзамен 1, 2 семестр

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта по направлению подготовки 09.03.03 Прикладная информатика, утвержденного приказом Министерства образования и науки РФ от 19 сентября 2017 г. № 922

Рабочая учебная программа обсуждена на заседании кафедры «Информационные системы управления», протокол № 6 от «13» февраля 2020 г.

Заведующий кафедрой ИСУ, к.т.н., доцент А.И. Сухомлинов  
Составитель: старший преподаватель Е.А. Елсукова

Владивосток  
2020

I. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « \_\_\_\_ » \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

II. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « \_\_\_\_ » \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

III. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « \_\_\_\_ » \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

IV. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « \_\_\_\_ » \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

## 1. Цели и задачи освоения дисциплины:

Цель дисциплины: получение студентами базовых теоретических знаний по алгоритмизации и программированию и приобретение практических навыков программирования на языке высокого уровня (ЯВУ).

Задачи дисциплины:

- изучить приемы алгоритмизации и программирования при решении задач на ЭВМ;
- дать представление информации в ЭВМ и различных структур данных;
- рассмотреть типовые задачи программирования и методы их решения, оценить сложность рассмотренных алгоритмов;
- освоить систему программирования PascalABC.NET;
- получить навыки оформления программной документации.

Результаты освоения (формирование компетенций):

Профессиональные компетенции выпускников и индикаторы их достижения:

Задача профессиональной деятельности	Объект или область знания	Код и наименование профессиональной компетенции	Код и наименование индикатора достижения профессиональной компетенции	Основание (ПС, анализ иных требований, предъявляемых к выпускникам)
Тип задач профессиональной деятельности: производственно-технологический				
<p>Проведение работ по изготовлению компонентов информационных систем в виде спецификаций программного обеспечения. Определение и установка параметров для используемых программных пакетов.</p> <p>Проведение работ по установке программного обеспечения информационных систем и загрузке базы данных.</p> <p>Разработка технической документации и руководств для пользователей.</p>	<p>Прикладные и информационные процессы</p> <p>Информационные системы</p> <p>Информационные технологии</p>	<p>ПК-4. Способность изготавливать компоненты информационных систем, включая программные комплексы, базы данных и интерфейсы "человек - электронно-вычислительная машина", использовать современные инструментальные средства разработки, и программно-технологические платформы информационных систем</p>	<p>ПК-4.1. знает назначение и содержание основных понятий языка программирования высокого уровня; основные структуры данных Pascal и способы их реализации; основные конструкции Pascal.</p> <p>ПК-4.2. умеет осуществлять постановку задачи программирования, разрабатывать алгоритм ее решения и создавать компьютерные программы, используя технологию и средства структурного программирования.</p> <p>ПК-4.3. владеет навыками алгоритмизации и</p>	<p>Анализ иных требований</p> <p>Определено самостоятельно</p>

Задача профессиональной деятельности	Объект или область знания	Код и наименование профессиональной компетенции	Код и наименование индикатора достижения профессиональной компетенции	Основание (ПС, анализ иных требований, предъявляемых к выпускникам)
Контроль качества систем, включая тестирование компонентов ИС по заданным сценариям. Начальное обучение и консультирование пользователей по вопросам эксплуатации информационных систем. Осуществление технического сопровождения информационных систем в процессе ее эксплуатации.			программирования при решении задач на ЭВМ; владеет методами отладки и решения задач на ЭВМ в различных режимах (использования инструментальных средств разработки программ).	

## **I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА (72 час.)**

### **Раздел I. Введение в дисциплину (10 час.)**

#### **Тема 1. Введение в дисциплину (2 час.)**

Цели и задачи курса. Роль дисциплины в образовательной программе. Литература: обязательная, дополнительная, ресурсы Интернет. Место программирования в современном IT-обществе. Основные этапы решения задач на ЭВМ; Модель «Семь +».

#### **Тема 2. Основы алгоритмизации вычислительных процессов (4 час.)**

Понятие алгоритма; его свойства. Способы представления алгоритмов. Преимущества и недостатки способов. Правила построения схем алгоритмов. Системные методы разработки алгоритмов. Метод структурной алгоритмизации. Алгоритмизация линейных, разветвляющихся и циклических вычислительных процессов. Понятие сложности алгоритма. Алгоритмически неразрешимые задачи.

#### **Тема 3. Языки программирования: эволюция, основные понятия (2 час.)**

Эволюция языков программирования (ЯП). Типы ЯП: языки программирования низкого уровня, языки программирования высокого уровня. Связь между языком высокого уровня и машинным языком. Системы

программирования. Развитие систем программирования: визуальные среды быстрого проектирования. Структура языка программирования. Синтаксис и семантика.

#### **Тема 4. Pascal - ЯП высокого уровня. Программа на языке высокого уровня (2 час.)**

Алфавит языка Pascal. Элементарные конструкции. Имена, правила выбора имени. Числа: целые, действительные. Структура программы. Опции транслятора. Комментарии: назначение и правила составления.

### **Раздел II. Данные языка программирования Pascal (26 час.)**

#### **Тема 1. Основные положения концепции данных языка программирования высокого уровня Pascal (4 час.)**

Концепция данных: описание, перечень допустимых операций. Базовые типы данных (Integer, Real, Boolean, Char) - описание, диапазон значений, размещение в оперативной памяти, перечень допустимых операций; функции для работы с данным типом. Тип данных определяемый пользователем. Порядковый тип данных. Функции преобразования типов, идентичность и совместимость типов.

#### **Тема 2. Структурированный тип данных (2 час.)**

Классификация данных по структурному признаку. Структурированные типы данных - диапазон, перечисляемый тип. Структурированный тип данных массивы: способы описания, виды массивов (одномерные, многомерные), способы формирования массивов.

#### **Тема 3. Обработка массивов (2 час.)**

Типовые задачи обработки одномерных массивов. Типовые задачи обработки матриц: задача минимакса, вычисление следа матрицы, учет диагональных элементов.

#### **Тема 4. Строки (2 час.)**

Способы описания строковых данных. Таблица ASCII, UNICODE. Подпрограммы обработки строк. Обработка длинных строк.

#### **Тема 5. Записи (2 час.)**

Способы описания записи. Структура записи: иерархические записи, записи с вариантами, строки и массивы в качестве полей записи. Доступ к полям записи; оператор присоединения. Записи как операнды и параметры. Использование записей при обработке баз данных.

#### **Тема 6. Множества (2 час.)**

Описание множеств. Формирование множеств с помощью конструкторов. Виды множеств: литеральные, множества со значениями перечислимого типа. Операции для работы с множествами.

## **Тема 7. Организация хранения данных во внешней памяти (4 час.)**

Понятие физического и логического файла. Файловый тип данных в ЯП Pascal. Внешние и внутренние имена файлов. Организация работы с файлами. Виды файлов: текстовые, типизированные, нетипизированные. Системные файлы INPUT, OUTPUT. Процедуры и функции работы с файлами. Прямой и последовательный доступ к данным.

## **Тема 8. Динамические структуры данных (6 час.)**

Динамическая память, адреса и указатели. Ссылочный тип данных: описание, стандартные процедуры и функции. Линейные списки: основные виды и способы реализации; линейный список как абстрактный тип данных. Сложные динамические структуры.

## **Тема 9. Графы (2 час.)**

Основные понятия теории графов. Способы задания графов. Типовые задачи теории графов. Алгоритмы, основанные на обходах графов.

## **Раздел III. Основные управляющие структуры программирования (14 час)**

### **Тема 1. Операторы Pascal (2 час.)**

Понятия: операция, операнд, функция, знак операции. Классификации операторов. Выражения, тип выражения. Оператор присваивания. Смешанное присваивание. Представление математических формул на Pascal. Стандартные математические функции.

### **Тема 2. Представление основных управляющих структур программирования (6 час.)**

Структуры выбора: операторы if и case. Организация ветвлений в программах. Условный оператор. Оператор безусловного перехода. Оператор выбора case.

Циклические структуры - цикл-счетчик (for), циклы с условием (while, repeat). Методика создания циклов. Вложенные циклы. Циклы, управляемые событием. Циклы без итераций. Использование шаблонов при создании циклов: счетчики, сигнальные метки, Boolean-флаги.

### **Тема 3. Правила вывода для основных структур программирования (2 час.)**

Организация ввода/вывода. Процедуры Read, Write; ReadLn, WriteLn. Форматированный вывод целочисленных и вещественных значений. Организация ввода/вывода структурированных данных.

### **Тема 4. Процедуры и функции (4 час.)**

Виды подпрограмм на Pascal: процедуры и функции. Стандартные подпрограммы Pascal, подпрограммы определяемые пользователем. Способы

оформления подпрограмм. Локальные и глобальные имена; область действия идентификаторов. Формальные и фактические параметры. Вызов подпрограмм. Организация обмена данными между главной программой и подпрограммами.

#### **Раздел IV. Разработка прикладных программ (12 час.)**

##### **Тема 1. Алгоритмы целочисленной арифметики (2 час.)**

Операции целочисленной арифметики (div, mod). Особенности алгоритмов целочисленной арифметики. Алгоритм поиска простых чисел (Решето Эратосфена).

##### **Тема 2. Рекурсия (2 час.)**

Понятия рекурсии: рекурсивное определение, рекурсивный спуск, рекурсивный подъём, глубина рекурсии, рекурсивный алгоритм. Программирование рекурсивных алгоритмов.

##### **Тема 3. Сортировка (2 час.)**

Понятие упорядоченных данных. Классификация методов сортировки. Оценка эффективности сортировки. Метод выбора, модификации. Метод простого обмена (пузырька), модификация - "шейкер-сортировка". Метод простых вставок; модификации - метод бинарных вставок, метод Шелла. Метод Хоара.

##### **Тема 4. Эвристические алгоритмы поиска (2 час.)**

Понятие эвристического алгоритма. Алгоритмы линейного и бинарного поиска. Алгоритмы нахождения пути на карте: волновой алгоритм (Алгоритм Ли), маршрутный алгоритм.

##### **Тема 5. Численные методы (4 час.)**

Нелинейные уравнения: постановка задачи, методы решения. Задача численного интегрирования: метод прямоугольников, метод трапеции, метод Симпсона; точность вычислений.

#### **Раздел V. Структурное программирование (10 час.)**

##### **Тема 1. Модули в Pascal (2 час.)**

Структура модуля. Доступ к объектам модуля. Компиляция и подключение модулей. Разработка модулей пользователя (Unit).

##### **Тема 2. Стандартные модули (библиотеки) Pascal (2 час.)**

Структура системы программирования Pascal. Модуль DOS: типы константы, переменные, подпрограммы. Модуль CRT - Основные процедуры и функции текстового режима.

##### **Тема 3. Библиотека работы с графикой (2 час.)**

Программное обеспечение графического режима работы монитора.

Особенности работы в графическом режиме. Алгоритмы и основные операторы формирования графических изображений. Создание анимации.

#### **Тема 4. Методология и технология программирования (2 час.)**

Императивное программирование, Теорема структуры. Структурное программирование: основные характеристики. Жизненный цикл программы.

#### **Тема 5. Этап отладки и тестирования (2 час.)**

Корректность программ; способы конструирования и верификации программ. Анализ программ.

## **II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА**

### **Практические занятия курса (54 час.)**

#### **Практические занятия 1 семестра (18 час.)**

**Занятие 1.** Способы описания алгоритмов. Исполнение алгоритмов с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 2.** Решение физических задач с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 3.** Разветвляющиеся алгоритмы с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 4.** Циклические алгоритмы с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 5:** Алгоритмы с использованием подпрограмм с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 6.** Алгоритмы целочисленной арифметики с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 7.** Алгоритмы обработки одномерных массивов с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 8.** Алгоритмы обработки двумерных массивов (матриц) с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 9.** Организация рекурсии с использованием метода активного обучения работа в малых группах (2 час.)

#### **Практические занятия 2 семестра (36 час.)**

## **Модуль “Курсовая работа” (12 час.)**

**Занятие 1.** Составление плана курсовой работы с использованием метода активного обучения кейс-метод (2 час.)

**Занятие 2.** Представление теоретической части курсовой работы с использованием метода активного обучения дискуссия (4 час.)

**Занятие 3.** Представление практической части курсовой работы с использованием метода активного обучения кейс-метод (4 час.)

**Занятие 4.** Оформление отчета по курсовой работы с использованием метода активного обучения кейс-метод (2 час.)

## **Практикум (24 час.)**

**Занятие 1:** Алгоритмы с использованием подпрограмм с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 2:** Работа с записями с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 3:** Работа с множествами с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 4:** Работа с файлами с использованием метода активного обучения работа в малых группах (4 час.)

**Занятие 5:** Алгоритмы поиска с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 6:** Динамические структуры с использованием метода активного обучения работа в малых группах (4 час.)

**Занятие 7:** Работа с библиотекой CRT с использованием метода активного обучения работа в малых группах (2 час.)

**Занятие 8:** Алгоритмы создания анимации с использованием метода активного обучения работа в малых группах (4 час.)

## **Лабораторные работы (72 час.)**

### **Лабораторные работы 1 семестра (36 час.)**

**Лабораторная работа 1.** Описание функции по её графическому представлению с использованием метода активного обучения коллективные решения творческих задач (6 час.)

**Лабораторная работа 2.** Табуляция функций с использованием метода активного обучения коллективные решения творческих задач (4 час.)

**Лабораторная работа 3.** Вычисление суммы ряда с заданной точностью с использованием метода активного обучения коллективные решения творческих задач (4 час.)

**Лабораторная работа 4.** Работа с массивами с использованием метода активного обучения коллективные решения творческих задач (4 час.)

**Лабораторная работа 5.** Основные действия на матрицами с использованием метода активного обучения коллективные решения творческих задач (6 час.)

**Лабораторная работа 6.** Реализация метода сортировки и оценка его эффективности с использованием метода активного обучения коллективные решения творческих задач (6 час.)

**Лабораторная работа 7.** Геометрия на плоскости с использованием метода активного обучения коллективные решения творческих задач (6 час.)

### **Лабораторные работы 2 семестра (36 час.)**

**Лабораторная работа 1.** Работа с перечисляемыми типами с использованием метода активного обучения коллективные решения творческих задач (6 час.)

**Лабораторная работа 2.** Использование множеств при обработке символьных и числовых массивов с использованием метода активного обучения коллективные решения творческих задач (4 час.)

**Лабораторная работа 3.** Использование записей при обработке статистических данных с использованием метода активного обучения коллективные решения творческих задач (6 час.)

**Лабораторная работа 4.** Работа с различными типами файлов с использованием метода активного обучения коллективные решения творческих задач (6 час.)

**Лабораторная работа 5.** Программирование численных методов с использованием метода активного обучения коллективные решения творческих задач (8 час.)

**Лабораторная работа 6.** Работа с динамическими структурами с использованием метода активного обучения коллективные решения творческих задач (6 час.)

### **III. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ**

#### **План-график выполнения самостоятельной работы»**

##### **1 семестр**

<b>№ п/п</b>	<b>Дата/сроки выполнения</b>	<b>Вид самостоятельной работы</b>	<b>Примерные нормы времени на выполнение</b>	<b>Форма контроля</b>
1.	В течение семестра	Подготовка к лабораторной работе (активная форма проведения)	36	УО-1
2.	В течение семестра	Оформление отчета по лабораторной работе	18	УО-1
3.		Подготовка к экзамену	36	ПР-2
<b>Всего в семестре</b>			<b>90 часов</b>	

##### **2 семестр**

<b>№ п/п</b>	<b>Дата/сроки выполнения</b>	<b>Вид самостоятельной работы</b>	<b>Примерные нормы времени на выполнение</b>	<b>Форма контроля</b>
1.		Подготовка к экзамену	36	ПР-2
<b>Всего в семестре</b>			<b>36 часов</b>	

#### **Методические рекомендации к оформлению отчетов по лабораторным работам**

Отчеты представляются в электронной форме, оформленные в MSWord по правилам, принятым в ДВФУ. Отчет должен содержать:

1. Титульный лист \*.
2. Содержание \*.
3. Задание.
4. Краткое изложение теоретических положений, необходимых для выполнения работы.

5. Основная часть: материалы выполнения заданий.
6. Анализ полученных результатов.
7. Выводы по работе \* (какие задачи решены, что освоено при выполнении работы).
8. Приложения \* (при необходимости, обычно в Приложении выносят листинг программы)

\* Включаются в отчет с новой страницы

#### IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые разделы дисциплины	Коды и этапы формирования компетенций		Оценочные средства - наименование	
				текущий контроль	промежуточная аттестация
1.	Введение в дисциплину	ПК-4.1.	Знает назначение и содержание основных понятий языка программирования высокого уровня;	Собеседование (УО-1)	Вопросы 2÷3,5, 8, 14 (I); 1-10 (II);
2.	Данные языка программирования Pascal	ПК-4.1.	Знает основные структуры данных Pascal и способы их реализации.	Собеседование (УО-1); Тестирование (ПР-2)	Вопросы 2, 3, 7,14 -15 (I), 1-10 (II)
		ПК-4.2.	Умеет организовать обработку различных типов и структур данных	Практические занятия (ПР-6) 7, 8 (I), 3-5,7 (II) Лабораторные работы (ПР-6) 4,5 (I), 1-4, 6 (II)	
		ПК-4.3.	Владеет навыками алгоритмизации и программирования при решении задач на ПК.	Практические занятия (ПР-6) 7, 8 (I), 3-5,7 (II) Лабораторные работы (ПР-6) 4,5 (I), 1-4, 6 (II)	
3.	Основные управляющие структуры программирования	ПК-4.1.	Знает основные конструкции Pascal.	Собеседование (УО-1); Тестирование (ПР-2)	Вопросы 1÷13 (I)
		ПК-4.2	Умеет осуществлять постановку	Практические занятия (ПР-6) 1,3-5,9 (I)	

№ п/п	Контролируем ые разделы дисциплины	Коды и этапы формирования компетенций		Оценочные средства - наименование	
				текущий контроль	промежуточна я аттестация
			задачи программирован ия	Лабораторные работы (ПР-6) 2,3 (I)	
		ПК- 4.3.	Владеет методами отладки и решения задач на ЭВМ в различных режимах.	Лабораторные работы (ПР-6) 2,3,5 (I)	
4.	Разработка прикладных программ	ПК-4.2	умеет осуществлять постановку задачи программирован ия, разрабатывать алгоритм ее решения	Практические занятия (ПР-6) 1-4 (I), 1,6,9 (II) Лабораторные работы (ПР-6) 1,2,3 (I) 1,2,3,5 (II),	Вопросы 15, 19 (I),
		ПК-4.3.	владеет навыками алгоритмизации и программирован ия при решении задач на ЭВМ.	Практические занятия (ПР-6) 6-8 (I), 1,6,9 (II) Лабораторные работы (ПР-6) 6,7 (I), 1,2,3,5 (II),	
5.	Структурное программиров ание	ПК-4.2	Умеет создавать компьютерные программы, используя технологии и средства структурного программирован ия.	Практические занятия (ПР-6) 5, 9 (I); Лабораторные работы (ПР-6) 6,7 (I)	Вопросы 1÷4,20 (I); 11-15 (II)
		ПК-4.3.	владеет методами отладки и решения задач на ЭВМ в различных режимах (использования инструментальны х средств	Практические занятия (ПР-6) 1,8,9 (II) Лабораторные работы (ПР-6) 4-6 (II)	

№ п/п	Контролируемые разделы дисциплины	Коды и этапы формирования компетенций		Оценочные средства - наименование	
				текущий контроль	промежуточная аттестация
			разработки программ).		

## V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### Основная литература

*(электронные и печатные издания)*

1. Павловская Т. А. Паскаль. Программирование на языке высокого уровня. Учебник для вузов. 2-е изд. – СПб: Питер, 2010. – 464с.
2. Немнюгин С. Турбо Паскаль. Программирование на языке высокого уровня. Учебник для вузов. 2-е изд. – СПб: Питер, 2008. – 544с.
3. Павловская Т. А. Паскаль. Программирование на языке высокого уровня. Практикум. 2-е изд. – СПб: Питер, 2007. – 317с.
4. Немнюгин С. Турбо Паскаль: Практикум. 2-е изд. – СПб: Питер, 2007. – 272с.

### Дополнительная литература

*(печатные и электронные издания)*

1. Афанасьева Т.В. Основы визуальной алгоритмизации: Учеб. пособие для студентов спец. 5102, 5525, 5501/Под ред. Валеева С.Г. – Ульяновск, изд-во УлГТУ, 2002. – 70с.
2. Бентли Дж. Жемчужины программирования. 2-е изд. – СПб: Питер, 2002. – 272с.
3. Кнут Д. Искусство программирования, Т.1,2,3. – М.: Издательский дом “Вильямс”, 2010.
4. Коффман Э. Б. Turbo Pascal, 5-е издание. – М: Издательский дом “Вильямс”, 2003. – 896с.
5. Малыхина М.П. Программирование на языке высокого уровня. Turbo Pascal. Уч. Пособие. – СПб: БХВ-Петербург, 2006. – 544с.
6. Меженный О.А. Самоучитель TURBO PASCAL. – М: Диалектика, 2007. – 330с.
7. Методические указания и задания по курсу «Алгоритмические языки и программирование» для студентов специальности 220200. – Березкина

Г.Л., Елсукова Е.А., Покотило А.Л., Церабаев А.В. – Владивосток, ДВГТУ, изд-во ДВГТУ, 2006. – 60с.

8. Павловская Т.А. Паскаль. Карманный справочник. – СПб: Питер, 2005. – 160с.

9. Попов В.Б Паскаль и Дельфи. Учебный курс. – СПб: Питер, 2005. – 576с.

10. Порублев И.Н., Ставровский А.Б. Алгоритмы и программы. Решение олимпиадных задач. — М.: Издательский дом “Вильямс”, 2007. – 480с.

11. Рапаков Г.Г., Ржеуцкая С.Ю. Turbo Pascal для студентов и школьников. – СПб.: БХВ-Петербург, 2012. – 352с.

12. Симонович С.В. Информатика. Базовый курс: Учебник для вузов. 3-е изд. Стандарт третьего поколения.– СПб: Питер, 2012. – 640с.

### **Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**

1. <http://pascal.guti.ru/> – Электронный учебник, примеры программ
2. <http://pascal.sources.ru/> - примеры программ
3. <http://www.pascal-c.com/> - примеры программ.

### **Электронно-библиотечная система IPRbooks**

1. Окулов С.М. Основы программирования [Электронный ресурс]: учебное пособие/ Окулов С.М.— Электрон. текстовые данные.— М.: БИНОМ. Лаборатория знаний, 2012.— 340с.— Режим доступа: <http://www.iprbookshop.ru/6449>.

2. Культин, Н. Б. Turbo Pascal в задачах и примерах. - СПб.: БХВ-Петербург, 2012. – 256с. .— Режим доступа: <http://www.iprbookshop.ru/>

3. Культин, Н. Б. Программирование в Turbo Pascal 7.0 и Delphi. Самоучитель. 3-е изд. .- СПб.: БХВ-Петербург.- 2012. – 390с. .— Режим доступа: <http://www.iprbookshop.ru>

### **Перечень информационных технологий и программного обеспечения**

На основе учебно-методического комплекса дисциплины «Введение в программирование» разработан электронный учебный курс в интегрированной платформе электронного обучения Blackboard ДВФУ.

Идентификатор курса в Blackboard – FU50706-09.03.01-Pr-01.

## **VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

Результаты освоения дисциплины достигаются за счет использования в процессе обучения: лекций с применением мультимедийных технологий, активных методов обучения с использованием LMS Blackboard; лабораторных работ и практических занятий на базе компьютерной сети.

Целью практических занятий является изучение базовых алгоритмов. На практических занятиях рассматриваются типовые задачи и соответствующие алгоритмы. Все практические занятия проходят с использованием активных методов обучения: работа в малых группах, кейс-метод.

Цель лабораторных занятий - закрепить теоретические знания, полученные на лекционных занятиях и приобрести практические навыки разработки программ.

Задание на лабораторную работу обучающиеся получают заранее на лекции соответствующей тематики для качественной подготовки к ее выполнению. Перед выполнением лабораторной работы студент должен ознакомиться с заданием и порядком выполнения на странице курса в LMS Blackboard.

### **Общие требования к курсовой работе**

По данной дисциплине предусмотрено выполнение курсовой работы во 2 семестре.

Выполнение курсовой работы является заключительным этапом освоения студентами дисциплины “Основы программирования”.

Цель курсовой работы - закрепление приобретённых навыков алгоритмизации задач с использованием структурного подхода и самостоятельное освоение этапов создания программного продукта.

**Порядок выдачи задания:** Задание на курсовую работу выдается руководителем курсового проектирования в первые две недели второго семестра на консультации. Задание выдается индивидуально каждому студенту. Возможна групповая работа над курсовым проектом с большим объемом работы. В этом случае каждому студенту выдается задание на определенную часть работы.

Если у студента есть пожелания по выбору темы, то необходимо об этом заранее сообщить руководителю курсового проектирования. Студент может предложить индивидуальную тему, ее сложность должна быть сопоставима с типовыми. Тематика, содержание курсовой работы должно быть предварительно обсуждено с преподавателем и оформлено в виде задания. После официального получения задания тема не может быть изменена.

### **Тематика курсовых работ**

1. Реализация методов вычислительной математики (Обработка табличных данных - Интерполяция; Аппроксимация; Решение нелинейных уравнений; Численное интегрирование; Одномерная оптимизация)
2. Обработка данных различных предметных областей (Библиотека, Цветочный магазин, Туристическое агентство, Игровой клуб)
3. Игры.
4. Обучающие программы по разделам курса.

### **Состав курсовой работы**

К работе оформляется отчет (описание / пояснительная записка). Отчет должен иметь объем не менее 15 страниц текста, выполнен в соответствии с ГОСТом. Отчет должен содержать следующие разделы:

1. Титульный лист
2. Задание
3. Оглавление
4. Ключевые слова (Словосочетания)
5. Теоретический материал
6. Структурное описание разработки
7. Функциональное описание
8. Заключение
9. Список используемой литературы
10. Приложения.

#### **Содержание разделов отчета**

Титульный лист (1) должен быть оформлен в соответствии с общими требованиями к курсовым работам, действующим в ДВФУ.

Задание (2) на специальном бланке выдается руководителем курсового проектирования.

Оглавление (3) должно строиться средствами текстового редактора.

В п.4 необходимо привести 5-6 ключевых слов (или ключевых словосочетаний), которые отражают основной смысл, являются характерными (ключевыми) для данной работы.

В теоретической части (5) должен быть рассмотрен один из аспектов программирования, связанный с реализацией поставленной задачи. Теоретический материал должен быть переработан применительно к содержанию работы.

В разделе “Структурное описание разработки” (6) нужно рассмотреть как будет использоваться в практической части работы (программе) изложенный теоретический материал; какие основные решения приняты при

разработке. Привести содержательное описание основных алгоритмов, их особенностей; описание используемых данных, особенности проектирования.

Структурное описание – словесное, с привлечением необходимых графических иллюстраций (схем, рисунков), фрагментов оригинальных алгоритмов и структур данных. Рисунки и диаграммы должны быть выполнены стандартными средствами MS Office и содержать сквозную нумерацию и подписи.

В разделе “Функциональное описание” (7) следует рассмотреть работу программы. Привести описание: форматов данных (отдельно указав какие данные являются статическими, какие – динамическими); используемых подпрограмм, их интерфейсов; фрагментов реализации нетривиальных алгоритмов; работы программы на контрольных примерах (например, на большом текстовом файле). Для описания использовать смешанное текстово-формальное представление программы с включением фрагментов программы в связный текст изложения материала.

Заключение (8) содержит краткие выводы по результатам курсовой работы и применения теоретических знаний в практическом программировании.

В приложения (10) обычно включают: схемы, рисунки, графическое описание данных, формат выходного документа, схему последовательности обработки данных, листинг программы с комментариями по существу алгоритма и структур данных.

### **Защита курсовой работы**

Курсовая работа должны быть представлена к защите в течение трех последних учебных недель второго семестра. Курсовая работа предварительно сдается для проверки руководителю. Работа сдается в бумажном (отчет) и электронном (файлы с программой) виде за 2-3 дня до защиты.

Программа должна быть представлена на внешнем носителе всеми исходными файлами и подготовленным для установки и выполнения на любом ПК EXE-файлами. При архивировании файлов использовать самораспаковывающиеся архивы.

К защите следует подготовить доклад в форме презентации MS Power Point с 6-7 слайдами или средствами HTML, который должен отражать основные положения курсовой работы.

Защита выполняется с демонстрацией на ПК работы программы и необходимыми пояснениями по ее проектированию и программированию. Студент должен свободно ориентироваться в представленном материале.

Студент демонстрирует свою работу на тестовых примерах. Если у преподавателя имеются замечания принципиального характера, а работа защищается в назначенный срок или раньше, то автор может отсрочить защиту, исправив замечания, для получения более высокой оценки.

Замечания, выявленные в процессе проверки должны быть устранены в течение 2-х недель после ее предъявления. Для всех, кто защищает курсовые на зачетной неделе, возможность доработки на более высокую оценку не предоставляется.

При выставлении оценки за курсовую работу учитывается реализация требований, полнота решения поставленной задачи, качество программной реализации, оформление отчета (пояснительной записки).

## **VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

В процессе изучения дисциплины используются мультимедийные и технические средства обучения. При проведении занятий используются аудитории со средствами вычислительной техники:

- рабочие станции или виртуальные машины на рабочих станциях с доступом к сети Интернет.
- Персональный компьютер преподавателя с мультимедиа-проектором и экраном, программным обеспечением для демонстрации презентаций.

Для самостоятельной работы с медиаматериалами каждому студенту требуется персональный компьютер или планшет, широкополосный доступ в сеть Интернет, браузер последней версии, устройство для воспроизведения звука (динамики, колонки, наушники и др.).

## **VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ**

### **Методические рекомендации, определяющие процедуры оценивания результатов освоения дисциплины**

Промежуточная аттестация студентов по дисциплине «Основы программирования» проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной.

По данной дисциплине предусмотрен следующие виды промежуточной аттестации – экзамен, экзамен проводится в письменной форме с использованием оценочного средства: тест / письменный экзамен.

### **Промежуточная аттестация студентов**

#### **Вопросы к экзамену (1 семестр)**

1. Основные этапы решения задач на ЭВМ.
2. Алгоритм: понятие, основные свойства, способы описания.
3. Виды алгоритмов: линейный, разветвляющийся, циклический, подпрограмма.
4. Методика составления алгоритма с циклом
5. Элементы языка программирования Pascal: алфавит, имена, константы, операторы, операции, выражения.
6. Понятие программы. Структура программы на Pascal.
7. Понятие идентификатора (имени). Правила выбора имени.
8. Описание базовых (стандартных) типов данных на Pascal.
9. Операторы языка Pascal и их классификация.
10. Оператор присваивания. Пустой оператор. Составной оператор.
11. Операторы ввода/вывода на Pascal.
12. Реализация условий на Pascal: условный оператор, оператор варианта.
13. Реализация циклов на Pascal: цикл с параметром, цикл с проверкой условия.
14. Организация хранения данных в памяти ЭВМ. Переменные. Массивы.
15. Описание и обработка массивов на Pascal.
16. Подпрограммы языка Pascal: формальные и фактические параметры, локальные и глобальные переменные.
17. Описание функций, обращение к функциям.
18. Описание процедур, обращение к процедурам.
19. Организация рекурсии.
20. Структурное программирование: назначение, реализация.

#### **Вопросы к экзамену (2 семестр)**

1. Типы данных Pascal.
2. Перечисляемый тип данных, особенности работы.
3. Тип данных – диапазон, особенности работы.
4. Строковый тип данных: операции, стандартные функции и процедуры обработки.
5. Тип данных – множества: определение, операции над множествами.

6. Тип данных – записи: объявление, обращение к записи, допустимые операции над целой записью и элементами, запись с вариантами.
7. Файловый тип данных: типы файлов, работа с файлами.
8. Ссылочный тип данных (указатели): определение, допустимые операции.
9. Динамические структуры – списки, стеки.
10. Совместимость данных.
11. Модули: назначение, создание, структура модуля.
12. Системные модули (библиотеки) Pascal: назначение и содержание библиотек CRT, DOS, GRAPH.
13. Основные процедуры и функции текстового режима.
14. Особенности работы в графическом режиме.
15. Этапы работы программы.

### **Примеры билетов письменного экзамена**

#### **Экзаменационный лист 1 (1 семестр)**

1. Перечислите способы описания алгоритмов
2. Укажите недостатки(-) и преимущества(+) описания алгоритмов с помощью графических схем.
3. Перечислите стандартные (базовые) типы данных, используемые в Pascal:
4. Что определяет тип данных, выбранный для конкретного объекта в Pascal-программе
5. Укажите назначение и общий вид оператора присваивания:
6. Какого типа будет результат, если в арифметическом выражении оператора присваивания есть данные и вещественного и целого типа?
7. Определите тип и значения следующих выражений:
  - a.  $2 * 5 + 9 \text{ div } 4 * 3 + \text{TRUNC}(3.67)$ ;
  - b.  $C + \text{Ord}(\text{Round}(A) + \text{Ord}(B)) - \text{Trunc}(A)$ ;  
(здесь  $A=7.91(\text{real})$ ;  $B=\text{true}(\text{Boolean})$ ;  $C=5(\text{Integer})$ ).
8. При каких значениях переменных A, B, C будет ложным логическое выражение:
 
$$\text{Not}(A \text{ And } \text{Not}(B \text{ Or } \text{Not } C))$$
9. В каких случаях в программе используется цикл с предусловием?
10. Запишите вид операторов цикла с предусловием на языке Pascal.
11. Возможно ли, что цикл с предусловием не отработает ни разу? Ответ поясните.
12. Сколько раз выполнится тело цикла в данном фрагменте программы:
 
$$B := \text{False}; r := 45;$$

While Not B Do Begin B := r = 15; r := r mod 4 + 15 End;

13. В каких случаях предпочтительнее использовать отдельную переменную для хранения данных, а в каких – организовывать для этого массив?
14. Как объявить в программе матрицу вещественных чисел из 7 столбцов и 15 строк:
15. Какое логическое выражение нужно вписать во фрагмент программы, определяющий индекс первого элемента массива P[1..n], не равного x?  
m := 0;  
for i := n downto 1 do  
if \_\_\_\_\_ then m := i;
16. Дано натуральное число N. Что вычисляется в данном фрагменте программы?  
M := 0;  
While N <> 0 Do  
Begin If N Mod 10 > M Then M := N mod 10; N := N div 10  
End;
17. Как организовать вывод вещественных данных в нестандартной форме? Укажите вид этого оператора вывода. Приведите пример.
18. Перечислите типы реализации подпрограмм в Pascal.
19. Укажите общий вид описания (заголовка) процедуры.
20. Какие виды параметров могут быть указаны в заголовке процедуры при ее описании?

### Экзаменационный лист 2 (2 семестр)

1. Дано следующее описание переменных:  
Var A,B:extended; W:Word; J:integer; M, N:byte;  
Какие из этих переменных совместимы по присваиванию?
2. К процедурам/функциям работы со строками не относятся  
1) Insert 2) Dispose 3)Copy 4)Concat 5)Delete.
3. Определить результат выполнения функции Val,  
здесь – IntNum переменная Integer, RealNum переменная Real;  
a. Val(' -3507', IntNum, Error);  
b. Val('1.23E3', IntNum, Error);  
c. Val('1.23E\_3', RealNum, Error);
4. Вычислить следующие выражения:  
a. [9] <= [1 .. 9];  
b. ['A'..'D', 'K'..'M'] + ['D'..'K'];  
c. [7, 1,3..6] = [1 .. 7];  
d. 15 IN [1.. 10];

5. Могут ли компоненты (поля) записи быть различных типов? Приведите пример, поясняющий ваш ответ
6. Как в записи задается Вариантная часть? Приведите пример, поясняющий ваш ответ.
7. Объясните разницу между внутренним и внешним именем файла.
8. Перечислите функции/процедуры, используемые для работы только текстовыми файлами:
9. Дано описание `var f: file of integer; x, y: integer;` Файл `f` содержит элементы 3,7. Определить значение `Y` после выполнения операторов:  
`reset (f); y :=0;`  
`while not eof(f) do begin read(f,x); y := y+x; end;`
10. К файлам какого типа в ТР можно обратиться с помощью прямого доступа?
11. Типизированный указатель – это...  
Приведите примеры описания нетипизированных указателей в виде типов и переменных.
12. Опишите структуру библиотечного модуля
13. Назначение библиотеки DOS:
14. Перечислите 3-4 процедуры/функции библиотеки DOS
15. Кратко опишите механизм построения изображения в графическом режиме.

Баллы рейтинговой оценки	Оценка экзамена (стандартная)	Требования к сформированным компетенциям
От 86% до 100%	«Отлично»	Выставляется студенту, если он глубоко и прочно усвоил программный материал курса, четко и последовательно излагает его, знает базовые алгоритмы; способы представления различных видов информации на ЭВМ.; владеет уверенными навыками работы в интегрированной среде разработки программ, использует технологию и средства структурного программирования..
От 76% до 85%	«Хорошо»	Выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.
От 61% до 75%	«Удовлетворительно»	Выставляется студенту, если он имеет знания только основного материала, но не

Баллы рейтинговой оценки	Оценка экзамена (стандартная)	Требования к сформированным компетенциям
		усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических заданий: реализовывает базовые алгоритмы с ошибками.
Менее 61%	«Неудовлетворительно»	Выставляется студенту, который не знает значительной части программного материала: основные понятия языка программирования высокого уровня; основные структуры данных Pascal и способы их реализации; основные конструкции Pascal., неуверенно, с большими затруднениями выполняет практические задания по составлению алгоритмов и программ.

## Оценочные средства для текущей аттестации

### Вопросы для собеседования (УО-1)

#### Раздел 1. Введение в дисциплину

1. Перечислить особенности решения задач с использованием ЭВМ?
2. На какие этапы разбивается решение задач на ЭВМ, какими операциями они могут быть дополнены?
3. Чем отличается этап постановки задачи от этапа формализации?
4. Что такое алгоритм, и его свойства?
5. Какие существуют способы описания алгоритмов? Охарактеризуйте (преимущества и недостатки) эти способы.
6. Какие правила должны выполняться при оформлении схем алгоритмов?
7. Что содержит команда на машинном языке?
8. Перечислить основные составляющие язык программирования высокого уровня:
9. Что образует синтаксис языка программирования?
10. Назначение семантики языка программирования.

#### Раздел 2. Данные языка программирования Pascal

1. Что определяет тип данных?

2. Для данных целого, логического и символьного типов указать: диапазон значений, допустимые операции, представление в оперативной памяти.
3. Что соответствует простой величине?
4. Чем определяется представление данных вещественного типа?
5. Привести собственные примеры задач, в которых используются данные только целого (вещественного) типа.
6. Для каких данных следует использовать перечисляемый тип? Приведите примеры.
7. Приведите собственные примеры корректного описания диапазонов.
8. Указать характеристики структурной величины.
9. Для каких задач данные следует организовать в массив, а не представлять простыми переменными?
10. Какого типа не может быть индекс массива?
11. Как определяется положение элемента в массиве?
12. Обязательно ли указывать при объявлении строкового типа данных длину строки? Ответ поясните.
13. Структура таблицы ASCII.
14. Назначение процедур VAL, STR. Приведите примеры задач, для решения которых следует использовать данные процедуры?
15. Для данной строки символов привести несколько способов описания.
16. Поясните, почему длина строки не может превышать 255 символов?
17. В чем заключается существенное отличие между массивами и записями?
18. Какие служебные слова определяют список полей записи?
19. Каких типов не может быть идентификатор поля записи?
20. Укажите все способы обращения к полю записи.
21. Какое определение называется рекурсивным? Приведите собственные примеры рекурсивных определений.
22. Какой вспомогательный алгоритм (подпрограмма) называется рекурсивными? Приведите собственные примеры содержательных задач, где для решения может быть использован рекурсивный вспомогательный алгоритм.
23. Что такое граничное условие и каково его назначение в рекурсивной подпрограмме?
24. Что такое рекурсивный спуск, рекурсивный подъём, глубина рекурсии?
25. На каком этапе выполнения рекурсивной подпрограммы могут выполняться её операторы?

26. Перечислить основные этапы работы с файлами.
27. Какие из процедур должны обязательно предшествовать процедуре Read?
28. Какие файлы называются системными? Укажите их назначение, приведите собственные примеры описания.
29. Чем отличается структура текстового файла от типизированного файла?
30. Для каких типов файлов используются прямой доступ к данным?

### **Раздел 3. Основные управляющие структуры ЯП**

1. Перечислить основные этапы создания модуля пользователя.
2. Для чего предназначена при создании модуля секция инициализации?
3. В каком разделе модуля (библиотеки) содержатся тексты подпрограмм?
4. Как формируется изображение в графическом режиме работы монитора?
5. Перечислить основные этапы создания анимации.