Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Амурский государственный университет»

Кафедра математического анализа и моделирования

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ ИНФОРМАТИКА

Основной образовательной программы по направлению подготовки 010500.62 – Прикладная математика и информатика

Благовещенск 2012

УМКД разработан канд. физ.-мат. наук, доцентом Веселовой Еленой Михайловной

Рассмотрен и рекомендован на заседании кафедры
Протокол заседания кафедры от «__» _____ 2012 г. № __
И.о. зав. кафедрой _____ Н.Н. Максимова

СОДЕРЖАНИЕ

1	Рабочая программа учебной дисциплины	4
1.1	Цели и задачи освоения дисциплины	4
1.2	Место дисциплины в структуре ООП ВПО	4
1.3	Структура и содержание дисциплины «Информатика»	4
1.4	Содержание разделов и тем дисциплины	5
1.5	Самостоятельная работа	7
1.6	Образовательные технологии	8
1.7	Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов	9
1.8	Учебно-методическое и информационное обеспечение дисциплины «Информатика»	10
1.9	Материально-техническое обеспечение дисциплины	11
1.10	Рейтинговая оценка знаний студентов по дисциплине	11
2	Краткое изложение программного материала	13
3	Методические указания	24
3.1	Методические указания по изучению дисциплины	24
3.2	Методические указания к практическим занятиям	26
3.3	Методические указания по самостоятельной работе студентов	28
4	Контроль знаний	29
4.1	Текущий контроль знаний	29
4.2	Итоговый контроль	33
5	Интерактивные технологии и инновационные методы, используемые в образовательном процессе	34

1 Рабочая программа учебной дисциплины

1.1 Цели и задачи освоения дисциплины

Цель дисциплины: обучение студентов основам работы на ЭВМ, техническим и программным средствам реализации информационных процессов, программированию, теоретическим основам и практическим навыкам проектирования и реализации программ на современных ЭВМ, а также знакомство и получение навыков работы с современными информационными технологиями и сетями.

После изучения дисциплины студенты должны знать и уметь применять: 1) аппаратное и программное обеспечение компьютера на уровне опытного пользователя; 2) среду программирования Turbo Pascal на уровне разработчика консольных и оконных приложений, активно использующего структурное, модульное и объектно-ориентированное программирование.

В результате освоения дисциплины обучающийся должен демонстрировать следующие результаты образования:

Знать:

- основные понятия основ программирования;
- о конструировании алгоритмов;
- методы структурного и модульного программирования;
- основные структуры данных (списки, множества и т.п.) и методах их обработки и способах реализации,
- методы и технологии программирования;

Уметь:

- разрабатывать алгоритмы;
- реализовывать алгоритмы на языке программирования высокого уровня;
- описывать основные структуры данных;
- реализовывать методы обработки данных;
- применять теории, методы, алгоритмы, системы и средства информационных технологий при решении профессиональных задач.

Владеть:

- общепрофессиональными знаниями теории, методов, систем, предназначенных для решения практических задач в области информационных технологий с использованием современных языков и инструментальных средств.

Предварительное усвоение иных дисциплин, предусмотренных государственным стандартом высшего профессионального образования для данного направления, не требуется.

1.2 Место дисциплины в структуре ООП ВПО

Дисциплина «Информатика» включена в цикл общих математических и естественнонаучных дисциплин, индекс дисциплины ЕН.Ф.2.

1.3 Структура и содержание дисциплины «Информатика»

Общая трудоемкость дисциплины составляет 153 часа.

№ п/п	Раздел (модуль) дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля успеваемости *(по неделям семестра)* Форма промежуточной аттестации *(по семестрам)*
				Лекц.	Прак.	Лабор.	Самост.	
1	2	3	4	5	6	7	8	9

1	2	3	4	5	6	7	8	9
1	Вычислительная техника в работе инженера	1	1	2	1		4	Подготовка индивидуального задания №1 – «Выполнение арифметических действий в различных системах счисления»
2	Технические и программные средства реализации информационных процессов	1	2	2	1		4	Подготовка индивидуального задания №2 – «Численные методы решения уравнений: метод половинного деления, метод хорд, метод Ньютона»
3	Современные информационные технологии и сети	1	3	2			2	Подготовка к устному опросу по теме лекционного занятия
4	Алгоритмы и алгоритмизация	1	4-5	4	4		4	Подготовка к устному опросу по темам практической работы и лекционного занятия
5	Этапы и методы разработки программ	1	6	2	2		4	Подготовка к устному опросу по темам практической работы и лекционного занятия. Подготовка индивидуального задания №3 – «Экстремум функции»
6	Основы программирования на языке Паскаль	1	7-16	20	8	14	20	Подготовка к устному опросу по темам практической и лабораторной работ, подготовка отчета по работе
7	Работа с графикой в среде Турбо Паскаль	1	17-18	4	2	4	12	Подготовка к устному опросу по темам практической и лабораторной работ, подготовка отчета по работе. Подготовка индивидуального задания №4 – «Использование нетипизированных файлов для создания эффекта движения графических примитивов по экрану»
8	Подготовка к итоговому тестированию по темам лекционных и практических работ						10	
9	Подготовка к экзамену						21	
Итого				**36**	**18**	**18**	**81**	

1.4 Содержание разделов и тем дисциплины

1.4.1 Лекции

Тема 1. Вычислительная техника в работе инженера.

Предмет, цели и задачи курса. Вычислительная техника и научно-технический прогресс. Представление об информационном обществе. Информационные ресурсы, продукты и

услуги. ЭВМ, их назначение и использование в различных областях человеческой деятельности.

Понятие информации и кодирования. Виды и свойства информации. Этапы обработки информации в ЭВМ. Общая характеристика сбора, хранения, обработки, накопления и передачи информации. Способы представления и преобразования информации.

Тема 2. Технические и программные средства реализации информационных процессов.

ЭВМ как универсальное устройство обработки информации, классификации ЭВМ. Структура и функции аппаратной части ПК. Тенденции развития ЭВМ.

Программное обеспечение ЭВМ, его состав и назначение: Системное программное обеспечение. Понятие операционной системы, назначение. Характеристика и основы работы основных операционных систем - MS-DOS, WINDOWS, UNIX, OS/2 и др. Программы-драйверы, программы - оболочки (NC, WC), утилиты.

Прикладное программное обеспечение. Классификация прикладных программ. Текстовые, табличные, графические редакторы, базы данных, научные редакторы и системы, интегрированные и издательские системы - краткая характеристика, назначение и классификация. Языки и системы программирования.

Тема 3. Современные информационные технологии и сети.

Понятие компьютерных сетей, общие принципы организации и функционирования. Архитектура сетей. Локальные и глобальные компьютерные сети. Интернет.

Понятие информационных технологий и систем, классификации, назначение, области применения. ГИС-технологии, назначение и основные возможности. Экспертные системы

Тема 4. Алгоритмы и алгоритмизация.

Алгоритм, способы задания и описания алгоритмов. Основные конструкции алгоритмов. Типы алгоритмов. Организация алгоритмов линейной, разветвляющейся, циклической и сложной комбинированной структуры.

Тема 5. Этапы и методы разработки программ.

Этапы внутреннего и внешнего проектирования. Отладка и тестирование программ. Модели надежности.

Тема 6. Основы программирования на языке Паскаль.

Общая характеристика Языка. Основные конструкции языка. Понятие блочной структуры. Синтаксические диаграммы. Структура программы. Константы и переменные. Типы данных. Ввод-вывод данных. Операторы. Процедуры и функции. Рекурсия. Структурированные типы данных: строки, массивы, множества, записи. Файлы в Паскале. Особенности различных реализаций Паскаля.

Тема 7. Работа с графикой в среде Турбо Паскаль.

Аппаратная и программная поддержка графики, инициализация графики. Базовые процедуры и функции: построения графических образов, работы с цветовой палитрой, работы с текстом.

1.4.2 Практические занятия

№ п/п	Наименование практических работ	Трудоемкость в часах
1	Информация, ее виды и свойства. Способы представления и преобразования информации. Системы счисления	2
2	Алгоритмы линейной, разветвленной и циклической структуры	4
3	Итерационные алгоритмы	2
4	Обработка массивов данных	2
5	Функции и процедуры	2

6	Обработка символьных данных	2
7	Работа с записями	2
8	Работа с внешними файлами	2
Итого		18

В ходе практических работ по данному курсу студенты приобретают знания о способах представления и преобразования информации, ее кодирования, технических и программных средствах реализации информационных процессов, алгоритмизации, основных этапах решения задач на ЭВМ, современных языках программирования.

1.4.3 Лабораторные занятия

№ п/п	Наименование лабораторных работ	Трудоемкость в часах
1	Изучение среды программирования Turbo Pascal. Программирование алгоритмов линейной структуры.	2
2	Программы разветвляющей структуры. Табулирование функций.	2
3	Программирование алгоритмов циклической структуры. Нахождение корней уравнений.	2
4	Нахождение суммы ряда с точностью. Вычисление интеграла.	2
5	Работа с одномерными массивами	2
6	Работа с двумерными массивами	2
7	Работа со строковыми переменными. Обработка символьных данных. Множества.	2
8	Записи. Файлы.	2
9	Инициализация графики. Построение графиков функций.	2
Итого		18

При выполнении лабораторных работ по данному курсу студенты должны ознакомиться с современными методами и средствами программирования, языком Турбо Паскаль, его синтаксисом и семантикой, типах данных, способах и механизмах управления данными, методах и основных этапах трансляции; продемонстрировать умение программирования различных функциональных и вычислительных задач.

Лабораторная работа выполнятся строго в соответствии с выданным преподавателем заданием и вариантом. По окончанию занятия студент обязан сдать разработанную программу и объяснить алгоритм решения поставленной задачи.

1.5 Самостоятельная работа

Самостоятельная работа – 81 часов. По данному курсу в рамках самостоятельной работы студента предполагается подготовка к устной защите лабораторных работ, текущая подготовка по темам практических занятий, подготовка к контрольному тестированию и итоговому контролю в конце семестра.

№ п/п	№ раздела дисциплины	Форма (вид) самостоятельной работы	Трудоемкость в часах
1	1	Подготовка индивидуального задания №1 – «Выполнение арифметических действий в различных системах счисления»	4
2	2	Подготовка индивидуального задания №2 – «Численные методы решения уравнений: метод половинного деления, метод хорд, метод Ньютона»	4

3	3	Подготовка к устному опросу по теме лекционного занятия	2
4	4	Подготовка к устному опросу по темам практической работы и лекционного занятия	4
5	5	Подготовка к устному опросу по темам практической работы и лекционного занятия. Подготовка индивидуального задания №3 – «Экстремум функции»	4
6	6	Подготовка к устному опросу по темам практической и лабораторной работ, подготовка отчета по работе	20
7	7	Подготовка к устному опросу по темам практической и лабораторной работ, подготовка отчета по работе. Подготовка индивидуального задания №4 – «Использование нетипизированных файлов для создания эффекта движения графических примитивов по экрану»	12
9	Подготовка к итоговому тестированию по темам лекционных и практических работ		10
10	Подготовка к экзамену		21
Итого			81

1.6 Образовательные технологии

В соответствии с требованиями ГОС ВПО по направлению подготовки 010500.62 – Прикладная математика и информатика реализация компетентностного подхода предусматривает широкое использование в учебном процессе активных и интерактивных форм проведения занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся.

При преподавании дисциплины «Информатика» используются как традиционные (лекция, проблемная лекция, лекция-семинар), так и инновационные технологии (применение мультимедийного проектора при изучении отдельных тем, применение рейтинговой системы оценки знаний студентов, «мозговой штурм», «метод проектов», возможно использование ресурсов сети Internet и электронных учебников).

Лекционные, лабораторные и практические занятия проводятся с использованием традиционной, активной и интерактивной форм обучения.

Распределение образовательных технологий соответствует проведению занятий в интерактивной форме в объеме не менее 20% от аудиторных занятий – 14 часов.

Интерактивные формы обучения используются на лекционных, практических и лабораторных занятиях, темы которых приведены в таблице:

Наименование тем	Лек.	Практ.	Σ
Вычислительная техника в работе инженера *проблемная лекция*	2	-	2
Алгоритмы и алгоритмизация *проблемная лекция, метод группового решения задач*	2	2	4
Основы программирования на языке Паскаль *проблемная лекция, метод группового решения задач, мозговой штурм, использование ресурсов сети Internet*	2	2	4
Работа с графикой в среде Турбо Паскаль *метод группового решения задач, использование ресурсов сети Internet*	2	2	4
Итого			14

1.7 Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

Оценочные средства для текущего контроля успеваемости и для промежуточной аттестации: рейтинговая система оценки знаний учащихся.

Текущий контроль за аудиторной и самостоятельной работой обучаемых осуществляется во время проведения практических и лабораторных занятий посредством устного опроса по контрольным вопросам соответствующего раздела, а также проверки отчетов по лабораторным работам. Промежуточный контроль осуществляется два раза в семестр в виде анализа итоговых отчетов на аттестационные вопросы. Итоговый контроль осуществляется после успешного прохождения студентами текущего и промежуточного контроля в виде экзамена.

Зачет сдается в конце семестра. Форма сдачи зачет – устная. Необходимым условием допуска на зачет является сдача всех практических работ.

Учебно-методическое обеспечение самостоятельной работы студентов: основная и дополнительная литература, официальные ресурсы сети Internet, установленное в вузе программное обеспечение.

Примерные вопросы к экзамену:

1. Современные ЭВМ, их классификации, назначение и использование в инженерной практике.
2. Системное и прикладное программное обеспечение. Технические средства ЭВМ.
3. Операционные системы. Языки программирования.
4. Современные информационные технологии и сети.
5. Представление информации в ЭВМ. Системы счисления.
6. Алгоритмы, способы задания и описания. Основные конструкции алгоритмов.
7. Константы: целые, вещественные, логические, символьные. Понятие идентификатор.
8. Переменные, их описание в программе. Скалярные типы. Тип integer. Операции и функции, применимые для данного типа.
9. Переменные, их описание в программе. Скалярные типы. Тип real. Операции и функции, применимые для данного типа.
10. Переменные, их описание в программе. Скалярные типы. Тип boolean. Логические операции AND,OR,NOT.
11. Переменные, их описание в программе. Скалярные типы. Тип char. Операции и функции, применимые для данного типа.
12. Структура программы на языке Паскаль. Комментарии.
13. Оператор присваивания. Приоритет операций.
14. Операторы WRITE, WRITELN.
15. Оператор безусловного перехода GOTO.
16. Составной оператор.
17. Оператор условного перехода IF.
18. Оператор CASE. Операции сравнения.
19. Оператор цикла WHILE.
20. Оператор цикла REPEAT.
21. Оператор цикла FOR.
22. Процедура ввода READ, READLN.
23. Массивы. Множества.
24. Подпрограммы общего вида (PROCEDURE).
25. Подпрограммы-функции (FUNCTION).
26. Рекурсия. Рекурсивные программы.
27. Локальные и глобальные переменные.
28. Формальные и фактические параметры в процедуре.
29. Понятие о файлах. Процедуры открытия и закрытия файлов.

30. Процедуры и функции работы с файлами: GET, PUT, EOF, READ, WRITE.
31. Текстовые файлы.
32. Записи. Записи с вариантами.
33. Работа с графикой в среде Turbo Pascal.
34. Процедуры и функции, предназначенные для построения графических образов.
35. Процедуры и функции, предназначенные для работы с цветовой палитрой.

1.8 Учебно-методическое и информационное обеспечение дисциплины «Информатика»

а) основная литература:

1. Давыдов, В. Г. Программирование и основы алгоритмизации [Текст]: учеб. пособие: рек. УМО / В. Г. Давыдов. - 2-е изд., стер. - М. : Высш. шк., 2005. - 448 с.

2. Енашенков, А. М. Программирование в среде Turbo Pascal 7.0 [Электронный ресурс // biblioclib.ru]: учебник / А.М. Енашенков. - 4-е изд., испр. и доп. – М. : Диалог-Мифи, 2004. – 367 с.

б) дополнительная литература:

1. Павловская, Т. А. Паскаль. Программирование на языке высокого уровня [Текст]: учебник : рек. Мин. обр. РФ / Т.А. Павловская. - СПб. : Питер, 2003, 2004. - 393 с.

2. Чалкина, Н. А. Практикум по решению задач на языке Turbo Pascal 7.0 по курсу "Информатика" (для нематематических спец.) [Текст]: практикум / Н. А. Чалкина, Т. В. Масленко ; АмГУ, ФМиИ. - Благовещенск : Изд-во Амур. гос. ун-та, 1999. - 67 с.

3. Ковалевская Е. В. Методы программирования [Электронный ресурс // biblioclib.ru]: учебно-методический комплекс / Е.В. Ковалевская, Н.В. Комлева. – М.: Евразийский открытый институт, 2011. – 319 с.

4. Зеленяк О. П. Практикум программирования на Turbo Pascal. Задачи, алгоритмы и решения [Текст] / О.П. Зеленяк. – 3-е изд., испр. и доп. – Спб.: ДиаСофрЮП, М.: ДМКПресс, 2010. – 320 с.

5. Зубов, В. С. Программирование на языке TURBO PASCAL (версии 6.0 и 7.0) [Текст]: справочник по процедурам, функциям, диагностическим сообщениям / В.С. Зубов. - 3-е изд., испр. - М. : Филинъ, 1997. - 320 с.

6. Методы программирования [Текст] : учеб. пособие / Ред. Г.А. Угольницкий. - 2-е изд. - М. : Вузовская книга, 2000. - 280 с.

в) Периодические издания

1. Журнал «Открытые системы СУБД»
2. Журнал «Программирование»
3. Журнал «Информационные системы и технологии»
4. Журнал «Информационно-управляющие системы»
5. Журнал «Информационные технологии и вычислительные системы»
6. Журнал «Вычислительные технологии»

г) программное обеспечение и Интернет-ресурсы

№	Наименование ресурса	Краткая характеристика
1	http://www.iqlib.ru	Интернет-библиотека образовательных изданий, в которой собраны электронные учебники, справочные и учебные пособия. Удобный поиск по ключевым словам, отдельным темам и отраслям знания
2	http://exponenta.ru/	Имеются ресурсы: Internet-класс по Высшей Математике; работа с примерами, решенными в средах ППП; банк решенных студенческих задач; обсуждение на форуме

3	http://biblioclub.ru/	Электронная библиотечная система «Университетская библиотека – online» специализируется на учебных материалах для ВУЗов по научно-гуманитарной тематике, а также содержит материалы по точным и естественным наукам

1.9 Материально-техническое обеспечение дисциплины

Лекции и практические занятия проводятся в стандартной аудитории (ауд.338[а]), оснащенной в соответствии с требованиями преподавания теоретических дисциплин, включая мультимедиа-проектор.

Лабораторные работы проводятся в компьютерном классе, рассчитанном на 10 посадочных рабочих мест пользователей (ауд. 327, 329), в котором установлен пакет прикладных программ Scilab 5.3.3 (свободно-распространяемое программное обеспечение).

1.10 Рейтинговая оценка знаний студентов по дисциплине

№	Вид деятельности	Кол-во баллов (1 срок сдачи)	Кол-во баллов (2 срок сдачи)	Примечания
1	2	3	4	5
1	Лабораторная работа 1 на тему: Основы работы в текстовом редакторе	3	2	
2	Лабораторная работа 2 на тему: Основы работы в текстовом редакторе	3	2	
3	Лабораторная работа 1 на тему: Основы работы в табличном редакторе	3	2	
4	Лабораторная работа 2 на тему: Основы работы в табличном редакторе	3	2	
5	Самостоятельная работа по теме: Построение алгоритмов линейной, разветвляющейся и циклических структуры	2 1,5 1	1,5 1 0,5	
6	Лабораторная работа на тему: Изучение среды программирования Pascal. Программирование формул	3	2	
7	Лабораторная работа на тему: Программы разветвляющей структуры. Табулирование функций	3	2	
8	Лабораторная работа на тему: Программирование алгоритмов циклической структуры. Простые циклы. Нахождение корней уравнений	3	2	
9	Самостоятельная работа по теме: Программирование формул и алгоритмов разветвляющей структуры	2 1,5 1	1,5 1 0,5	
10	Лабораторная работа на тему: Нахождение суммы ряда с точностью	3	2	
11	Лабораторная работа на тему: Вычисление интеграла	3	2	
12	Лабораторная работа на тему: Вложенные оператора цикла. Нахождение экстремума функции	3	2	

1	2	3	4	5
13	Самостоятельная работа по теме: Циклические операторы и программирование задач с помощью простых циклов	2 1,5 1	1,5 1 0,5	
14	Лабораторная работа на тему: Работа с одномерными массивами	3	2	
15	Индивидуальное домашнее задание на тему: Алгоритмы циклической структуры	2 1,5 1	1,5 1 0,5	
16	Лабораторная работа на тему: Обработка матриц	3	2	
17	Лабораторная работа на тему: Подпрограммы в языке Паскаль. Процедуры	3	2	
18	Индивидуальное домашнее задание на тему: Обработка массивов данных	2 1,5 1	1,5 1 0,5	
19	Лабораторная работа на тему: Подпрограммы в языке Паскаль. Функции	3	2	
20	Лабораторная работа на тему: Обработка символьных данных. Множества	3	2	
21	Индивидуальное домашнее задание на тему: Обработка символьных данных	2 1,5 1	1,5 1 0,5	
22	Лабораторная работа на тему: Работа с записями	3	2	
23	Лабораторная работа на тему: Работа с файлами	3	2	
24	Лабораторная работа на тему: Работа со строковыми переменными	3	2	
25	Итоговый тест по теоретическому и практическому курсу дисциплины	10 8 6	8 6 4	
Итого максимальное количество баллов		70		

Учебная дисциплина «Информатика» относится к категории дисциплин с экзаменом и оценивается в 100 баллов за семестр. Пересчет рейтинговой оценки дисциплины проводится по шкале:

менее 51 балла – «неудовлетворительно»;

от 51 до 74 баллов – «удовлетворительно»;

от 75 до 90 баллов – «хорошо»;

от 91 до 100 баллов – «отлично».

Рейтинговая оценка студента по дисциплине «Информатика» складывается из баллов, набранных по текущему контролю, баллов набранных за экзамен и премиальных баллов. Сумма баллов по итогам текущего контроля составляет 70 баллов, по итогам промежуточного контроля студент может получить 30 баллов. За своевременное выполнение студентами домашних работ и лабораторных заданий, за активную работу на лекционных и лабораторных занятиях могут начисляться премиальные 10 баллов. Предполагается использование штрафных баллов за пропуск занятий без уважительной причины.

2 Краткое изложение программного материала

Название темы 1 Вычислительная техника в работе инженера

План лекции:

Предмет, цели и задачи курса. Вычислительная техника и научно-технический прогресс. Представление об информационном обществе. Информационные ресурсы, продукты и услуги. ЭВМ, их назначение и использование в различных областях человеческой деятельности.

Понятие информации и кодирования. Виды и свойства информации. Этапы обработки информации в ЭВМ. Общая характеристика сбора, хранения, обработки, накопления и передачи информации. Способы представления и преобразования информации.

Цели, задачи: введение в предмет «Информатика», глубокое разъяснение и системное изложение учебного материала по теме «Вычислительная техника в работе инженера»

Ключевые вопросы:

Информация – это отражение внешнего мира с помощью знаков или сигналов.

Информационная ценность сообщения заключается в новых сведениях, которые в нем содержатся (в уменьшении незнания).

Свойства информации: полнота - свойство информации исчерпывающе (для данного потребителя) характеризовать отображаемый объект или процесс;

актуальность - способность информации соответствовать нуждам потребителя в нужный момент времени;

достоверность - свойство информации не иметь скрытых ошибок. Достоверная информация со временем может стать недостоверной, если устареет и перестанет отражать истинное положение дел;

доступность - свойство информации, характеризующее возможность ее получения данным потребителем;

релевантность - способность информации соответствовать нуждам (запросам) потребителя;

защищенность - свойство, характеризующее невозможность несанкционированного использования или изменения информации;

эргономичность - свойство, характеризующее удобство формы или объема информации с точки зрения данного потребителя.

Информация всегда связана с материальным носителем.

Носителем информации может быть:

любой материальный предмет (бумага, камень и т.д.);

волны различной природы: акустическая (звук), электромагнитная (свет,радиоволна) и т.д.;

вещество в различном состоянии: концентрация молекул в жидком растворе, температура и т.д.

Машинные носители информации: перфоленты, перфокарты, магнитные ленты, и т.д.

Сигнал - способ передачи информации. Это физический процесс, имеющий информационное значение. Он может быть непрерывным или дискретным.

Сигнал называется дискретным, если он может принимать лишь конечное число значений в конечном числе моментов времени.

Аналоговый сигнал - сигнал, непрерывно изменяющийся по амплитуде и во времени.

Сигналы, несущие текстовую, символическую информацию, дискретны.

Аналоговые сигналы используют в телефонной связи, радиовещании, телевидении.

Говорить об информации вообще, а не применительно к какому-то ее конкретному виду беспредметно. Классифицировать ее можно:

по способам восприятия (визуальная, тактильная и т.д.);

по форме представления (текстовая, числовая, графическая и т. д.);

по общественному значению (массовая, специальная, личная).

Информатика – область человеческой деятельности, связанная с процессами преобразования информации с помощью компьютеров и других средств вычислительной техники. С информатикой часто связывают одно из следующих понятий: это либо совокупность определенных средств преобразования информации, либо фундаментальная наука, либо отрасль производства, либо прикладная дисциплина.

Информатика как совокупность средств преобразования информации включает технические средства (hardware), программные продукты (software), математические методы, модели и типовые алгоритмы (brainware). В состав технических средств входят компьютеры и связанные с ними периферийные устройства (мониторы, клавиатуры, принтеры и плоттеры, модемы и т.д.), линии связи, средства оргтехники и т.п., т.е. те материальные ресурсы, которые обеспечивают преобразование информации, причем главенствующую роль в этом списке играет компьютер. По своей специфике компьютер нацелен на решение очень широкого круга задач по преобразованию информации, при этом выбор конкретной задачи при использовании компьютера определяется программным средством, под управлением которого функционирует компьютер. К программным продуктам относятся операционные системы и их интегрированные оболочки, системы программирования и проектирования программных продуктов, различные прикладные пакеты, такие, как текстовые и графические редакторы, бухгалтерские и издательские системы и т.д. Конкретное применение каждого программного продукта специфично и служит для решения определенного круга задач прикладного или системного характера. Математические методы, модели и типовые алгоритмы являются тем базисом, который положен в основу проектирования и изготовления программного, технического средства или другого объекта в силу исключительной сложности последнего и, как следствие, невозможности умозрительного подхода к созданию.

Перечисленные выше три ресурсных компонента информатики играют разную роль в процессе информатизации общества. Так, совокупность программных и технических средств, имеющихся в том или ином обществе, и позволяет сделать его информационным, когда каждый член общества имеет возможность получить практически любую (исключая, естественно, секретную) интересующую его информацию (такие потребители информации называются конечными пользователями). В то же время, сложность технических и программных систем заставляет использовать имеющиеся технические и программные продукты, а также нужные методы, модели и алгоритмы для проектирования и производства новых и совершенствования старых технических и программных систем. В этом случае можно сказать, что средства преобразования информации используются для производства себе подобных. Тогда их пользователем является специалист в области информатики, а не конечный пользователь.

Информатика как фундаментальная наука занимается разработкой абстрактных методов, моделей и алгоритмов, а также связанных с ними математических теорий. Ее прерогативой является исследование процессов преобразования информации и на основе этих исследований разработка соответствующих теорий, моделей, методов и алгоритмов, которые затем применяются на практике.

Информатика как отрасль производства практически использует результаты исследований фундаментальной науки информатики. В самом деле, широко известны западные фирмы по производству программных продуктов, такие как Microsoft, Lotus, Borland, и технических средств – IBM, Apple, Intel, Hewlett Packard и другие. Помимо производства самих технических и программных средств разрабатываются также и технологии преобразования информации.

Информатика как прикладная дисциплина занимается подготовкой специалистов в области преобразования информации. Она изучает закономерности протекания информационных процессов в конкретных областях и методологии разработки конкретных информационных систем и технологий.

Таким образом, главная функция информатики состоит в разработке методов и средств преобразования информации с использованием компьютера, а также в применении их при реализации технологического процесса преобразования информации.

Ссылки на литературные источники:

1.10.1-1.10.4, 1.10.11, 1.10.13

Название темы 2 Технические и программные средства реализации информационных процессов.

План лекции:

ЭВМ как универсальное устройство обработки информации, классификации ЭВМ. Структура и функции аппаратной части ПК. Тенденции развития ЭВМ.

Программное обеспечение ЭВМ, его состав и назначение: Системное программное обеспечение. Понятие операционной системы, назначение. Характеристика и основы работы основных операционных систем - MS-DOS, WINDOWS, UNIX, OS/2 и др. Программы-драйверы, программы - оболочки (NC, WC), утилиты.

Прикладное программное обеспечение. Классификация прикладных программ. Текстовые, табличные, графические редакторы, базы данных, научные редакторы и системы, интегрированные и издательские системы - краткая характеристика, назначение и классификация. Языки и системы программирования.

Цели, задачи: глубокое разъяснение и системное изложение учебного материала по теме «Технические и программные средства реализации информационных процессов»

Ключевые вопросы:

Назначением ЭВМ является выполнение программ. Программа содержит команды, определяющие порядок действий компьютера. Совокупность программ для компьютера образует программное обеспечение (ПО). По функциональному признаку различают следующие виды ПО:

- системное;
- прикладное.

Под системным (базовым) понимается программное обеспечение, включающее в себя операционные системы, сетевое ПО, сервисные программы, а также средства разработки программ (трансляторы, редакторы связей, отладчики и пр.).

Основные функции операционных систем (ОС) заключаются в управлении ресурсами (физическими и логическими) и процессами вычислительных систем. Физическими ресурсами являются: оперативная память, процессор, монитор, печатающее устройство, магнитные и оптические диски. К логическим ресурсам можно отнести программы, файлы, события и т. д. Под процессом понимается некоторая последовательность действий, предписанная соответствующей программой и используемыми ею данными.

В настоящее время существует большое количество ОС, разработанных для ЭВМ различных типов. На ЭВМ Единой Системы (ЕС ЭВМ), например, использовались такие операционные системы, как СВМ и ОС ЕС, на малых ЭВМ (СМ-4, СМ-1420 и др.) - ОС РВ и RSX-11. На персональных ЭВМ долгое время эксплуатировалась ОС-MS-DOS. В настоящее время получили распространение системы Windows 98/Me, Windows 2000, Linix.

Сетевое ПО предназначено для управления общими ресурсами в распределенных вычислительных системах: сетевыми накопителями на магнитных дисках, принтерами, сканерами, передаваемыми сообщениями и т. д. К сетевому ПО относят ОС, поддерживающие работу ЭВМ в сетевых конфигурациях (так называемые сетевые ОС), а также отдельные сетевые программы (пакеты), используемые совместно с обычными, не сетевыми ОС.

Например, большое распространение получили следующие сетевые ОС: NetWare 4.1 (фирма Novell), Windows NT Server 3.5 (фирма Microsoft) и LAN Server 4.0 Advanced (фирма IBM). Однако в последнее время лидирующие позиции начинает занимать ОС Windows 2000 Server фирмы Microsoft.

Для расширения возможностей операционных систем и предоставления набора дополнительных услуг используются сервисные программы. Их можно разделить на следующие группы:

- интерфейсные системы;
- оболочки операционных систем;
- утилиты.

Интерфейсные системы являются естественным продолжением операционной системы и модифицируют как пользовательский, так и программный интерфейсы, а также реализуют дополнительные возможности по управлению ресурсами ЭВМ. В связи с тем, что развитая интерфейсная система может изменить весь пользовательский интерфейс, часто их также называют операционными системами. Это относится, например, к Windows 3.11 и Windows 3.11 for WorkGroups (для рабочих групп).

Оболочки операционных систем, в отличие от интерфейсных систем, модифицируют только пользовательский интерфейс, предоставляя пользователю качественно новый интерфейс по сравнению с реализуемым операционной системой. Такие системы существенно упрощают выполнение часто запрашиваемых функций, например, таких операций с файлами, как копирование, переименование и уничтожение, а также предлагают пользователю ряд дополнительных услуг. В целом, программы-оболочки заметно повышают уровень пользовательского интерфейса, наиболее полно удовлетворяя потребностям пользователя.

На ПЭВМ широко используются такие программы-оболочки, как Norton Commander, FAR Manager и Windows Commander.

Утилиты предоставляют пользователям средства обслуживания компьютера и его ПО. Они обеспечивают реализацию следующих действий:

- обслуживание магнитных дисков;
- обслуживание файлов и каталогов;
- предоставление информации о ресурсах компьютера;
- шифрование информации;
- защита от компьютерных вирусов;
- архивация файлов и др.

Существуют отдельные утилиты, используемые для решения одного из перечисленных действий, и многофункциональные комплекты утилит. В настоящее время для ПЭВМ среди многофункциональных утилит одним из наиболее совершенных является комплект утилит Norton Utilities. Существуют его версии для использования в среде DOS и Windows.

Средства разработки программ используются для разработки нового программного обеспечения как системного, так и прикладного.

Прикладным называется ПО, предназначенное для решения определенной целевой задачи из проблемной области. Часто такие программы называют приложениями.

Спектр проблемных областей в настоящее время весьма широк и включает в себя по крайней мере следующие: промышленное производство, инженерную практику, научные исследования, медицину, управление (менеджмент), делопроизводство, издательскую деятельность, образование и т. д.

Из всего разнообразия прикладного ПО выделяют группу наиболее распространенных программ (типовые пакеты и программы), которые можно использовать во многих областях человеческой деятельности.

К типовому прикладному ПО относят следующие программы:

- текстовые процессоры;
- табличные процессоры;
- системы иллюстративной и деловой графики (графические процессоры);
- системы управления базами данных;
- экспертные системы;
- программы математических расчетов, моделирования и анализа экспериментальных данных.

Предлагаемые на рынке ПО приложения, в общем случае, могут быть выполнены как отдельные программы либо как интегрированные системы. Интегрированными системами обычно являются экспертные системы, программы математических расчетов, моделирования и анализа экспериментальных данных, а также офисные системы. Примером мощной и широко распространенной интегрированной системы является офисная система Microsoft Office.

Поскольку разработка ПО любого назначения, как правило, является довольно сложным и трудоемким процессом, дальнейший материал настоящего раздела посвятим общим вопросам разработки программ и инструментальному ПО.

Ссылки на литературные источники:

1.10.1-1.10.4, 1.10.11, 1.10.13

Название темы 3 Современные информационные технологии и сети

План лекции:

Понятие компьютерных сетей, общие принципы организации и функционирования. Архитектура сетей. Локальные и глобальные компьютерные сети. Интернет.

Понятие информационных технологий и систем, классификации, назначение, области применения. ГИС-технологии, назначение и основные возможности. Экспертные системы.

Цели, задачи: глубокое разъяснение и системное изложение учебного материала по теме «Современные информационные технологии и сети»

Ключевые вопросы:

Компьютерная сеть - система двух или более компьютеров, связанных каналами передачи информации. Для передачи информации могут быть использованы различные физические явления, как правило - различные виды электрических сигналов или электромагнитного излучения.

Создание компьютерных сетей вызвано практической потребностью пользователей удаленных друг от друга компьютеров в одной и той же информации. Сети предоставляют пользователям возможность не только быстрого обмена информацией, но и совместной работы на принтерах и других периферийных устройствах, и даже одновременной обработки документов.

Все многообразие компьютерных сетей можно классифицировать по группе признаков, например:

территориальная распространенность;

ведомственная принадлежность;

скорость передачи информации;

тип среды передачи;

По территориальной распространенности сети могут быть локальными, региональными и глобальными. Следует указать, что такое, деление, все же, довольно условно.

Локальная сеть объединяет компьютеры, установленные в одном помещении (например, школьный компьютерный класс, состоящий из 8—12 компьютеров) или в одном здании (например, в здании школы могут быть объединены в локальную сеть несколько десятков компьютеров, установленных в различных предметных кабинетах).

Локальная вычислительная сеть, ЛВС (англ. Local Area Network, LAN) — компьютерная сеть, покрывающая относительно небольшую территорию.

В небольших локальных сетях все компьютеры обычно равноправны, т. е. пользователи самостоятельно решают, какие ресурсы своего компьютера (диски, каталоги, файлы) сделать общедоступными по сети. Такие сети называются одноранговыми.

Ели к локальной сети подключено более десяти компьютеров, то одноранговая сеть может оказаться недостаточно производительной. Для увеличения производительности, а также в целях обеспечения большей надежности при хранении информации в сети некоторые компьютеры специально выделяются для хранения файлов или программ-приложений. Такие компьютеры называются серверами, а локальная сеть - сетью на основе серверов.

Каждый компьютер, подключенный к локальной сети, должен иметь специальную плату (сетевой адаптер). Между собой компьютеры (сетевые адаптеры) соединяются, например, с помощью кабелей.

Локальные сети не позволяют обеспечить совместный доступ к информации пользователям, находящимся, например, в различных частях города. На помощь приходят региональные сети, объединяющие компьютеры в пределах одного региона (города, страны, континента).

Региональная сеть – компьютерная сеть в пределах одного региона.

Глобальная вычислительная сеть, ГВС (англ. Wide Area Network, WAN) представляет собой компьютерную сеть, охватывающую большие территории и включающую в себя десятки и сотни тысяч компьютеров.

ГВС служат для объединения разрозненных сетей так, чтобы пользователи и компьютеры, где бы они ни находились, могли взаимодействовать со всеми остальными участниками глобальной сети. Лучшим примером ГВС является Интернет, но существуют и другие сети.

Глобальную компьютерную сеть еще называют телекоммуникационной сетью, а процесс обмена информацией по такой сети называют телекоммуникацией (от греч. «tele» - далеко и лат. «comunicato» - связь).

Общая схема соединения компьютеров в сети называется топологией сети.

Сетевая топология может быть физической (описывает реальное расположение и связи между узлами сети) и логической (описывает хождение сигнала в рамках физической топологии. Топологии сети могут быть различными). Мы будем рассматривать физическую.

Существует множество способов соединения сетевых устройств, из них можно выделить четыре базовых топологии: шина, кольцо, звезда и ячеистая топология. Остальные способы являются комбинациями базовых.

Локальные сети чаще всего могут иметь топологию «шина» или «звезда». В первом случае все компьютеры подключены к одному общему кабелю (шине), во втором - имеется специальное центральное устройство (хаб), от которого идут «лучи» к каждому компьютеру, т.е. каждый компьютер подключен к своему кабелю.

Интернет (от англ. Internet) - глобальная компьютерная сеть, построенная на использовании протоколов TCP/IP.

До середины 1990 годов Интернет был доступен относительно узкому академическому сообществу, а его наполнение не отличалось богатством и разнообразием. В настоящее время на десятках миллионов компьютеров, подключенных к Интернету, хранится громадный объем информации (сотни миллионов файлов, документов и т. д.) и сотни миллионов людей пользуются информационными услугами глобальной сети.

Основу, «каркас» Интернета составляют более ста миллионов серверов, постоянно подключенных к сети. К серверам Интернета могут подключаться с помощью локальных сетей или коммутируемых телефонных линий сотни миллионов пользователей сети.

В каждой локальной или корпоративной сети обычно имеется, по крайней мере, один компьютер, который имеет постоянное подключение к Интернету с помощью линии связи с высокой пропускной способностью (сервер Интернета).

Надежность функционирования глобальной сети обеспечивается избыточностью линий связи: как правило, серверы имеют более двух линий связи, соединяющих их с Интернетом.

Информационная система - взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели.

Ссылки на литературные источники:

1.10.1-1.10.4, 1.10.11, 1.10.13

Название темы 4 Алгоритмы и алгоритмизация

План лекции:

Алгоритм, способы задания и описания алгоритмов. Основные конструкции алгоритмов. Типы алгоритмов. Организация алгоритмов линейной, разветвляющейся, циклической и сложной комбинированной структуры.

Цели, задачи: глубокое разъяснение и системное изложение учебного материала по теме «Алгоритмы и алгоритмизация»

Ключевые вопросы:

Основным в процессе программирования является разработка алгоритма. Это один из наиболее сложных этапов решения задачи с использованием ЭВМ. В начале обучения программированию, на наш взгляд, целесообразно не привязываться сразу к какому-либо языку, разрабатывать алгоритмы без записи на языках программирования высокого уровня (ЯПВУ), а, например, с помощью блок-схем или иным аналогичным способом. После такой "чистой" алгоритмизации учащимся или студентам проще перейти к записи того же алгоритма на определённом языке программирования. В настоящей публикации продемонстрирован именно такой подход.

Напомним, что основными алгоритмическими структурами (ОАС) являются следование, ветвление и цикл. В более сложных случаях используются суперпозиции (вложения) ОАС.

Ссылки на литературные источники:

1.10.1-1.10.2, 1.10.5, 1.10.7, 1.10.10, 1.10.18

Название темы 5 Этапы и методы разработки программ

План лекции:

Этапы внутреннего и внешнего проектирования. Отладка и тестирование программ. Модели надежности.

Цели, задачи: глубокое разъяснение и системное изложение учебного материала по теме «Этапы и методы разработки программ»

Ключевые вопросы:

Решение задач на компьютере включает в себя следующие основные этапы, часть из которых осуществляется без участия компьютера.

1. Постановка задачи:
- сбор информации о задаче;
- формулировка условия задачи;
- определение конечных целей решения задачи;
- определение формы выдачи результатов;
- описание данных (их типов, диапазонов величин, структуры и т. п.).

2. Анализ и исследование задачи, модели:
- анализ существующих аналогов;
- анализ технических и программных средств;
- разработка математической модели;
- разработка структур данных.

3. Разработка алгоритма:
- выбор метода проектирования алгоритма;
- выбор формы записи алгоритма (блок-схемы, псевдокод и др.);
- выбор тестов и метода тестирования;
- проектирование алгоритма.

4. Программирование:
- выбор языка программирования;
- уточнение способов организации данных;
- запись алгоритма на выбранном языке программирования.

5. Тестирование и отладка:

- синтаксическая отладка;
- отладка семантики и логической структуры;
- тестовые расчеты и анализ результатов тестирования;
- совершенствование программы.

6. Анализ результатов решения задачи и уточнение в случае необходимости математической модели с повторным выполнением этапов 2-5.

7. Сопровождение программы:

- доработка программы для решения конкретных задач;
- составление документации к решенной задаче, к математической модели, к алгоритму, к программе, к набору тестов, к использованию.

Ссылки на литературные источники:

1.10.4, 1.10.9, 1.10.12, 1.10.17

Название темы 6 Основы программирования на языке Паскаль

План лекции:

Общая характеристика Языка. Основные конструкции языка. Понятие блочной структуры. Синтаксические диаграммы. Структура программы. Константы и переменные. Типы данных. Ввод-вывод данных. Операторы. Процедуры и функции. Рекурсия. Структурированные типы данных: строки, массивы, множества, записи. Файлы в Паскале. Особенности различных реализаций Паскаля.

Цели, задачи: глубокое разъяснение и системное изложение учебного материала по теме «Основы программирования на языке Паскаль»

Ключевые вопросы:

С понятием величины связаны следующие характеристики (атрибуты):

- идентификатор – это ее обозначение и место в памяти;

- тип – множество допустимых значений и множество применимых операций к ней;

- значение – динамическая характеристика, может меняться многократно в ходе исполнения алгоритма. Во время выполнения программы в каждый конкретный момент величина имеет какое-то значение или не определена.

Тип выражения определяется типами входящих в него величин, а также выполняемыми операциями. В языке Pascal тип величины задают заранее, т.к. все переменные, используемые в программе, должны быть объявлены в разделе описания с указанием их типа.

Различают переменные следующих простых типов: целые (Integer, Byte, ShortInt, Word, LongInt), вещественные (Real, Double, Single, Extended), логический (Boolean), символьный (Char), перечисляемый, диапазонный (интервальный).

В целом, иерархия типов в языке Pascal следующая:

1. Что такое величина?
2. Какие величины называют аргументами, результатами, промежуточными величинами? Приведите примеры.
3. Каковы атрибуты величины?
4. Какие величины называют постоянными? переменными? Приведите примеры.

5. Какие простые типы величин существуют в языке Pascal?
6. Что определяет тип величины?
7. Расскажите о простых типах данных и их атрибутах.
8. Как осуществляется ввод данных в языке Pascal? Приведите примеры.
9. Как осуществляется вывод данных в языке Pascal? Приведите примеры.
10. Какова общая структура программы в языке Pascal?
11. Расскажите об операторе присваивания и совместимости типов.
12. Что такое формат вывода?
13. Расскажите о правилах вычисления алгебраического выражения. Приведите примеры.
14. Расскажите о правилах вычисления логического выражения. Приведите примеры.
15. Расскажите о логических операциях. Приведите примеры.
16. Приведите примеры задач, имеющих линейный алгоритм решения.
17. Оператор присваивания.
18. Операторы (процедуры) ввода-вывода.
19. Когда возникает необходимость в организации ветвления?
20. Выражение какого типа может выступать в качестве условия при организации ветвления? Какие значения принимают такие выражения?
21. Назовите отличия итерационных циклов и цикла с параметром.
22. Какова структура оператора цикла с параметром? Каков алгоритм выполнения цикла с параметром?
23. Какого типа должны быть параметр цикла, его начальное и конечное значения в цикле с параметром в языке Pascal?
24. Могут ли параметр цикла, его начальное и конечное значения в цикле с параметром в языке Pascal быть разных типов? Обоснуйте ответ.
25. Может ли один цикл быть вложен внутрь другого? Если да, то какова глубина этой вложенности?
26. Какова структура циклов с пред- и постусловием? как выполняются эти циклы?
27. Каково минимальное и максимальное количество исполнений циклов с пред- и постусловием? С чем это связано?
28. Что такое массив?
29. Почему массив является структурированным типом данных?
30. Что такое размерность массива? Существуют ли ограничения на размерность массива?
31. Какого типа могут быть элементы массива?
32. Какого типа могут быть индексы элементов массива?
33. Какие простые типы данных относятся к порядковым?
34. Какими способами может быть заполнен массив? Приведите примеры.
35. Как определить минимальный объём памяти, отводимой под массив?
36. Какие действия выполняют обычно над элементами массива?
37. Может ли массив быть элементом массива?
38. В каком случае массивы совместны по присваиванию?
39. Как можно объявить величину строкового типа?
40. К каким типам данных относятся строки?
41. Какова максимально возможная длина строки?
42. С величиной какого типа данных совместим по присваиванию отдельный символ строки?
43. Расскажите об операциях, которые можно выполнять над строковыми величинами.
44. Расскажите о функциях, определенных для величин строкового типа.
45. Расскажите о процедурах, определенных для величин строкового типа.
46. Как осуществляется доступ к отдельному символу строки?

47. Что такое множество?

48. Почему множество является структурированным типом данных?

49. Как хранится множество в памяти ЭВМ? Какой максимальный объем оперативной памяти может быть отведен под хранение одного множества?

50. Какие операции можно выполнять над множествами?

51. Чем отличается тип "запись" от других структурированных типов?

52. Могут ли поля записи быть одного и того же типа?

53. Как обратиться к отдельному полю записи?

54. Что такое "оператор присоединения"? В каких целях он используется?

55. Что такое "запись с вариантами"?

56. Какие алгоритмы называют вспомогательными?

57. Какое количество вспомогательных алгоритмов может присутствовать в основном алгоритме?

58. Можно ли вспомогательные алгоритмы, написанные для решения данной задачи, использовать при решении других задач, где их применение было бы целесообразно?

59. Какие параметры называют формальными? фактическими?

60. Какое соответствие должно соблюдаться между формальными и фактическими параметрами?

61. Может ли фактических параметров процедуры (функции) быть больше, чем формальных? А меньше?

62. Существуют ли подпрограммы без параметров?

63. Существуют ли ограничения на число параметров подпрограмм? Если нет, то чем же всё-таки ограничивается это количество в Turbo Pascal?

64. В каком разделе объявляются и реализуются подпрограммы в Turbo Pascal?

65. Какие виды формальных параметров существуют? Чем они отличаются друг от друга?

66. В чём состоит отличие процедур и функций?

67. В каких случаях целесообразно использовать функции?

68. Почему, если в функции используются параметры-переменные, необходимо преобразовать её в процедуру?

69. Какого типа может быть значение функции?

Ссылки на литературные источники:

1.10.4, 1.10.9, 1.10.12, 1.10.17

Название темы 7 Работа с графикой в среде Турбо Паскаль

План лекции:

Аппаратная и программная поддержка графики, инициализация графики. Базовые процедуры и функции: построения графических образов, работы с цветовой палитрой, работы с текстом.

Цели, задачи: развитие навыков работы с системой программирования с использованием модуля Graph.

Ключевые вопросы:

Аппаратная поддержка графики персонального компьютера обеспечивается двумя основными модулями: видеоадаптером и видеомонитором. Видеомонитор – это устройство, на котором появляется выводимый текст или графические изображения. Видеоадаптеры – это электронные устройства, управляемые собственным микропроцессором.

При работе с экраном в графическом режиме очень важным является то, что любая информация представляет собой совокупность пикселей. Каждый пиксель определяется своими координатами, т.е. положением относительно левого верхнего угла экрана. С помощью программ можно управлять светимостью и цветом любого пикселя, т.е. создавать любое изображение.

Графический режим ПК существенно отличается от текстового как по принципам функционирования, так и по возможностям.

Графика применяется практически во всех серьезных программных разработках, так как позволяет увидеть результаты расчетов в виде чертежей, графиков, иллюстраций в движении.

Подключение графической библиотеки при программировании

в среде Turbo Pascal производится оператором: Uses Graph;

Переход из текстового режима к графическому (инициализация графики) осуществляется оператором: InitGraph (Gd, Gm, 'way');

где Gd - имя графического драйвера (параметр-переменная), Gm - номер графического режима монитора (параметр-переменная); 'way' - дорожка DOS к файлам с графическими драйверами.

Обычно драйверы подключаются в режиме автоопределения используемого монитора ПК. Для этого перед инициализацией графики задается Gd:=Detect; или Gd:=0;. В этом случае по умолчанию устанавливается режим с наибольшим числом точек на экране, а значение параметра Gm игнорируется. Номер наибольшего режима для текущего драйвера возвращает функция GetMaxMode;.

Разрешающую способность для текущего графического режима можно определить функциями, возвращающими максимальные значения координат экрана:

GetMaxX; - по оси X,

GetMaxY; - по оси Y.

Для возврата из графического режима в текстовый можно использовать операторы:

CloseGraph; - полное прекращение работы графической системы,

RestoreCrtMode; - переключение в текстовый режим с возможностью возврата к текущим установкам графического режима.

После инициализации графического режима экран очищается и можно использовать процедуры рисования.

SetBkColor (N); - установить цвет N для пикселов фона. По умолчанию установлен черный цвет фона.

Setcolor (N); - установить цвет N для выводимых линий. По умолчанию установлен белый цвет линий.

PutPixel (X, Y, N); - высветить цветом N пиксел с координатами X, Y

FloodFill (X, Y, N) - заполнение замкнутой области с границей цвета N.

GetPixel (X, Y); - функция возвращает цвет пикселя с координатами X, Y.

Цветовая шкала

Цвет	Код
Black- черный	0
Blue - сини	1
Green- зеленый	2
Cyan-бирюзовый	3
Red-красный	4
Magenta-малиновый	5
Brown-коричневый	6
LightGray-светло-серый	7
DarkGray-темно-серый	8
LightBlue- ярко-голубой	9
LightGreen-ярко-зеленый	10
LightCyan-ярко-бирюзовый	11
LightRed-ярко-красный	12
LightMagenta-ярко-малиновый	13
Yellow-желтый	14
White-белый	15

Ряд графических процедур выполняет построение заполненных фигур - фигур с замкнутым контуром, автоматически заполняемых сразу после построения :

SetFillStyle(P, N); - процедура установки орнамента P=0,1, .., 12 и цвета с номером N для заполняемых фигур.

P=0 - сплошное заполнение цветом фона,

P=1 - сплошное заполнение цветом с номером N,

P=2..11 - стандартный набор орнаментов BGI,

P=12 - орнамент и цвет определяет пользователь.

Bar (x1, y1, x2, y2); - построить заполненный прямоугольник с заданными координатами левого верхнего (x1,y1) и правого нижнего (x2,y2) углов.

Bar3d (x1, y1, x2, y2, d, t); - построить параллелепипед с заполненной передней гранью. Координаты углов (x1,y1), (x2,y2), глубина d. Переменная t типа boolean задает вывод верхней грани. При t=TopOn верхняя грань показывается, (true) t=TopOff (false) - нет.

FillEllipse (X, Y, Rx, Ry); - построить заполненный эллипс с центром в точке (X,Y) и радиусами (полуосями) Rx - по горизонтали, Ry - по вертикали.

Sector(x,y,a1,a2,Rx,Ry); - построить заполненный сектор эллипса. a1 и a2 начальный и конечный углы (в) градусах, отсчитываемые против часовой стрелки относительно направления оси X.

PieSlice(x, y, a1,a2,R); - построить заполненный сектор круга с центром в точке (x,y) и радиусом R.

FillPoly (N, M); - заполнить произвольную плоскую фигуру с границей, описанной массивом точек, N - число точек границы.

Ссылки на литературные источники:

1.10.1-1.10.4, 1.10.11, 1.10.13

3. Методические указания

3.1 Методические указания по изучению дисциплины

Дисциплина «Информатика» изучается на первом курсе бакалавриата на протяжении первого семестра.

В ходе изучения дисциплины уделяется внимание как теоретическому усвоению основ программирования, так и приобретению, развитию и закреплению практических навыков и умений по программированию.

На лекциях раскрываются основные вопросы рассматриваемой темы, делаются акценты на наиболее важные, сложные и проблемные положения изучаемого материала, которые должны быть приняты студентами во внимание.

На практических занятиях, ориентированных на предметную область будущей профессиональной деятельности студентов, выборочно контролируется степень усвоения студентами основных теоретических положений.

Целями практических занятий являются:

- выработка навыков разрабатывать алгоритмы;
- выработка навыков реализовывать алгоритмы на языке программирования высокого уровня;
- умение описывать основные структуры данных;
- умение реализовывать методы обработки данных;
- умение применять теории, методы, алгоритмы, системы и средства информационных технологий при решении профессиональных задач.

После изучения каждой темы предусматривается выполнение студентами самостоятельной работы с проверкой как степени усвоения ими теоретических знаний, так и объема и качества приобретенных практических навыков и умений.

В конце семестра предусмотрена контрольная работа, целями которой является комплексная проверка практических навыков и умений студентов по применению инструментальных средств программирования.

Для более глубокого изучения теоретического материала, приобретения и развития студентами навыков научно-исследовательской, аналитической работы с научной и учебной литературой, электронными источниками информации, информационно-поисковыми и справочными правовыми системами, развития навыков решения прикладных, профессионально-ориентированных задач.

Для лучшего усвоения положений дисциплины студенты должны:

- постоянно и систематически с использованием рекомендованной литературы и электронных источников информации закреплять знания, полученные на лекциях;
- находить решения проблемных вопросов, поставленных преподавателем в ходе лекций и практических заданий;
- регулярно и своевременно изучать материал, выданный преподавателем на самостоятельную проработку;
- с использованием средств программирования, электронных учебников, практикумов и информационных ресурсов сети Интернет выполнить на компьютере тематические практические задания, предназначенные для самостоятельной работы;
- найти, используя разные источники информации, ответы на теоретические и практические контрольные вопросы по темам дисциплины.

Студенты очной формы обучения обязаны присутствовать на занятиях и выполнять все предусмотренные учебно-методическим комплексом дисциплины формы учебной работы; проходить промежуточный и итоговый контроль в виде защит лабораторных и практических работ, аттестации в форме тестового контроля знаний; сдачи зачета и экзамена в предлагаемой преподавателем форме.

Дисциплина «Информатика» изучается студентами в 1 семестре обучения, который включает 36 часов лекционных занятий, 18 часов практических занятий, 18 часов лабораторных занятий и заканчивается экзаменом. На самостоятельную работу студентов отводится 81 час.

Теоретическая часть курса включает следующие темы (в скобках указан объем каждой лекции в часах).

Тема 1. Вычислительная техника в работе инженера (2).

Предмет, цели и задачи курса. Вычислительная техника и научно-технический прогресс. Представление об информационном обществе. Информационные ресурсы, продукты и услуги. ЭВМ, их назначение и использование в различных областях человеческой деятельности.

Понятие информации и кодирования. Виды и свойства информации. Этапы обработки информации в ЭВМ. Общая характеристика сбора, хранения, обработки, накопления и передачи информации. Способы представления и преобразования информации.

Тема 2. Технические и программные средства реализации информационных процессов (2).

ЭВМ как универсальное устройство обработки информации, классификации ЭВМ. Структура и функции аппаратной части ПК. Тенденции развития ЭВМ.

Программное обеспечение ЭВМ, его состав и назначение: Системное программное обеспечение. Понятие операционной системы, назначение. Характеристика и основы работы основных операционных систем - MS-DOS, WINDOWS, UNIX, OS/2 и др. Программы-драйверы, программы - оболочки (NC, WC), утилиты.

Прикладное программное обеспечение. Классификация прикладных программ. Текстовые, табличные, графические редакторы, базы данных, научные редакторы и системы, интегрированные и издательские системы - краткая характеристика, назначение и классификация. Языки и системы программирования.

Тема 3. Современные информационные технологии и сети (2).

Понятие компьютерных сетей, общие принципы организации и функционирования. Архитектура сетей. Локальные и глобальные компьютерные сети. Интернет.

Понятие информационных технологий и систем, классификации, назначение, области применения. ГИС-технологии, назначение и основные возможности. Экспертные системы

Тема 4. Алгоритмы и алгоритмизация (4).

Алгоритм, способы задания и описания алгоритмов. Основные конструкции алгоритмов. Типы алгоритмов. Организация алгоритмов линейной, разветвляющейся, циклической и сложной комбинированной структуры.

Тема 5. Этапы и методы разработки программ (2).

Этапы внутреннего и внешнего проектирования. Отладка и тестирование программ. Модели надежности.

Тема 6. Основы программирования на языке Паскаль (20).

Общая характеристика Языка. Основные конструкции языка. Понятие блочной структуры. Синтаксические диаграммы. Структура программы. Константы и переменные. Типы данных. Ввод-вывод данных. Операторы. Процедуры и функции. Рекурсия. Структурированные типы данных: строки, массивы, множества, записи. Файлы в Паскале. Особенности различных реализаций Паскаля.

Тема 7. Работа с графикой в среде Турбо Паскаль (4).

Аппаратная и программная поддержка графики, инициализация графики. Базовые процедуры и функции: построения графических образов, работы с цветовой палитрой, работы с текстом.

Каждая лекция содержит необходимый объем теоретического материала, изучение которого предусмотрено федеральным государственным образовательным стандартом направления. В дополнение к лекционному материалу, студентам рекомендуется использовать основную и дополнительную литературу согласно перечню, приведенному в п.1.8.

Студенты в рамках аудиторных занятий должны, в целом, владеть понятийным аппаратом, основанном на ранее изученных дисциплинах, воспринимать теоретический материал основного содержания лекции, видеть причинно-логические связи в лекции, понимать схему решения примеров, приводимых в лекции. Для освоения темы каждой лекции на более глубоком уровне требуется дополнительная работа с теоретическим материалом в форме прочтения и изучения основной и дополнительной литературы, самостоятельной работы с лекцией.

3.2 Методические указания к практическим и лабораторным занятиям

Практические и лабораторные работы направлены на закрепление теоретического материала на практическом уровне. Допускается работа в подгруппах, состоящих из 2 студентов, с выполнением одного варианта. Отчет в этом случае оформляется каждым студентом отдельно. Опрос проводится независимо от личного вклада в результат выполнения работы. Для выполнения лабораторной работы необходимо освоить теоретические основы соответствующего раздела, составить блок-схему реализации задачи, выполнить программную реализацию, протестировать задачу на примере, для которого известно аналитическое решение, оценить погрешность результата, оформить отчет по работе. При возникновении проблемных ситуаций в ходе решения практических задач (неясен алгоритм, непонятна ошибка программной среды при реализации метода, появились затруднения, связанные с тестированием алгоритма и пр.) или освоения теоретического материала преподавателем приветствуется любой диалог или дискуссия (возможно, с участием других студентов), направленные на решение проблемы, при необходимости отведения дополнительного и/или индивидуального времени – в рамках консультаций во внеаудиторное время.

Практический курс предусматривает практические и лабораторные занятия по следующим темам.

На практических занятиях отрабатываются приемы разработки алгоритмов и программирования.

Приведено планирование по тем темам, по которым предусмотрены практические занятия.

№ п/п	Наименование практических работ	Трудоемкость в часах
1	Информация, ее виды и свойства. Способы представления и преобразования информации. Системы счисления	2
2	Алгоритмы линейной, разветвленной и циклической структуры	4
3	Итерационные алгоритмы	2
4	Обработка массивов данных	2
5	Функции и процедуры	2
6	Обработка символьных данных	2
7	Работа с записями	2
8	Работа с внешними файлами	2
Итого		18

Наименование лабораторных работ приведено в таблице:

№ п/п	Наименование лабораторных работ	Трудоемкость в часах
1	Изучение среды программирования Turbo Pascal. Программирование алгоритмов линейной структуры.	2
2	Программы разветвляющей структуры. Табулирование функций.	2
3	Программирование алгоритмов циклической структуры. Нахождение корней уравнений.	2
4	Нахождение суммы ряда с точностью. Вычисление интеграла.	2
5	Работа с одномерными массивами	2
6	Работа с двумерными массивами	2
7	Работа со строковыми переменными. Обработка символьных данных. Множества.	2
8	Записи. Файлы.	2
9	Инициализация графики. Построение графиков функций.	2
Итого		18

Практическая часть курса методически поддержана учебным пособием, указанным в п.1.8.2 перечня основной литературы в пункте 1.8. Кроме методического пособия, студентам рекомендуется использовать также основную и дополнительную литературу согласно перечню, приведенному в п.1.8, при этом обращая внимание на практические аспекты использования алгоритмов и реализацию методов.

Индивидуальное задание (типовой расчет или лабораторная работа) выполняется строго в соответствии с выданным преподавателем заданием и вариантом. Оформлять работу следует четко и аккуратно, придерживаясь основных правил оформления отчетных работ: титульный лист (содержит: ФИО, №группы, курс, дисциплина, тема расчета и т. д.), лист задания (содержит перечень предложенных заданий), раздел, содержащий теоретические основы соответствующего раздела курса (включая подробный алгоритм основного метода), раздел, содержащий описание программной реализации.

Типовой расчет считается выполненным с дифференцированной оценкой, если:

1) работа выполнена полностью и в соответствии с заданием;

2) студент отвечает на основные теоретические вопросы по соответствующему разделу;

3) работа оформлена в соответствии с указанными требованиями.

Лабораторная работа считается выполненной с отметкой «зачтено», если:

1. Программная реализация соответствует заданию.
2. Студент отвечает на основные теоретические вопросы по разделу.

3. Работа оформлена в соответствии с указанными требованиями.

Допускается упрощенный вариант сдачи лабораторных работ в форме расчетных (с заменой блоков программ на однократно исполняемые модули с записью результатов). В этом случае, оценка, на которую может претендовать студент при итоговой аттестации – не выше «удовлетворительно» и, соответственно, число выставляемых баллов согласно балльно-рейтинговой оценки знаний не может превышать 70% от максимального числа баллов.

Сроки сдачи работ ограничены отведенным на выполнение практикума аудиторным временем практических и лабораторных занятий. Рекомендуется выполнять и сдавать на проверку отчеты по лабораторным и практическим работам по мере изложения лекционного материала и выдачи заданий преподавателем. Необходимым условием допуска студента на экзамен является сдача всех практических и лабораторных работ.

3.3 Методические указания по самостоятельной работе студентов

Важным компонентом изучения материала дисциплины является самостоятельная работа студентов. Это и проработка материала лекций, изучение основной и дополнительной литературы, практикумов, решение на компьютере практических задач, выданных на самостоятельную подготовку, подготовка к плановым занятиям, контрольной работе и экзамену.

В ходе самостоятельной работы студентам предлагается, используя полученные знания и навыки, рекомендованную литературу и возможности специализированных программных систем и комплексов, решить ряд практических заданий.

На самостоятельную работу студента по дисциплине «Информатика» отводится 81 час.

Схема самостоятельной работы студентов, перечень тем, рекомендации по работе с литературой, рекомендации по подготовке к аттестации:

№ п/п	№ раздела дисциплины	Форма (вид) самостоятельной работы	Трудоемкость в часах
1	1	Подготовка индивидуального задания №1 – «Выполнение арифметических действий в различных системах счисления» Изучение теоретических основ рекомендуется с использованием лекций по этой теме и литературных источников 1.8.1-1.8.4, 1.8.11, 1.8.13, указанных в перечне основной и дополнительной литературы. Вопросы практической реализации методов рекомендуется рассматривать с помощью литературного источника 1.8.4 в перечне дополнительной литературы.	4
2	2	Подготовка индивидуального задания №2 – «Численные методы решения уравнений: метод половинного деления, метод хорд, метод Ньютона» Изучение теоретических основ рекомендуется с использованием лекций по этой теме и литературных источников 1.8.1-1.8.2, 1.8.5, 1.8.7, 1.8.10, 1.8.18 указанных в перечне основной и дополнительной литературы. Вопросы практической реализации методов рекомендуется рассматривать с помощью литературного источника 1.8.4 в перечне дополнительной литературы.	4
3	3	Подготовка к устному опросу по теме лекционного занятия. Изучение теоретических основ рекомендуется с использованием лекций по этой теме и литературных источников 1.8.4, 1.8.9, 1.8.12, 1.8.17 указанных в перечне основной и дополнительной литературы.	2

4	4	Подготовка к устному опросу по темам практической работы и лекционного занятия Изучение теоретических основ рекомендуется с использованием лекций по этой теме и литературных источников 1.8.3, 1.8.10, 1.8.12, 1.8.16-1.8.19, указанных в перечне основной и дополнительной литературы. Вопросы практической реализации методов рекомендуется рассматривать с помощью литературного источника 1.8.4 в перечне дополнительной литературы.	4
5	5	Подготовка к устному опросу по темам практической работы и лекционного занятия. Подготовка индивидуального задания №3 – «Экстремум функции» Изучение теоретических основ рекомендуется с использованием лекций по этой теме и литературных источников 1.8.5, 1.8.6, 1.8.8, указанных в перечне основной и дополнительной литературы. Вопросы практической реализации методов рекомендуется рассматривать с помощью литературного источника 1.8.4 в перечне дополнительной литературы.	4
6	6	Подготовка к устному опросу по темам практической и лабораторной работ, подготовка отчета по работе Изучение теоретических основ рекомендуется с использованием лекций по этой теме и литературных источников 1.8.11, 1.8.12, 1.8.15-1.8.19, указанных в перечне основной и дополнительной литературы. Вопросы практической реализации методов рекомендуется рассматривать с помощью литературного источника 1.8.4 в перечне дополнительной литературы.	20
7	7	Подготовка к устному опросу по темам практической и лабораторной работ, подготовка отчета по работе. Подготовка индивидуального задания №4 – «Использование нетипизированных файлов для создания эффекта движения графических примитивов по экрану» Изучение теоретических основ рекомендуется с использованием лекций по этой теме и литературных источников 1.8.5, 1.8.7, 1.8.9-1.8.12, указанных в перечне основной и дополнительной литературы. Вопросы практической реализации методов рекомендуется рассматривать с помощью литературного источника 1.8.4 в перечне дополнительной литературы.	12
9	Подготовка к итоговому тестированию по темам лекционных и практических работ		10
10	Подготовка к экзамену		21
Итого			81

4 Контроль знаний

4.1 Текущий контроль знаний

Оценочные средства для текущего контроля успеваемости и для промежуточной аттестации: рейтинговая система оценки знаний учащихся.

Текущий контроль за аудиторной и самостоятельной работой обучаемых осуществляется во время проведения практических и лабораторных занятий посредством устного опроса

по контрольным вопросам соответствующего раздела, а также проверки отчетов по практическим и лабораторным работам. Промежуточный контроль осуществляется два раза в семестр в виде анализа итоговых отчетов на аттестационные вопросы. Для заключительной аттестации студентов в конце семестра обучения проводится контрольное тестирование по вариантам (которое является составной частью зачета по практической части курса).

Задачи для текущего контроля знаний:

1. Дан одномерный массив. Составить программу для выяснения, образуют ли элементы массива геометрическую прогрессию.
2. Дано слово. Составить программу для определения его симметричности.
3. Составить программу для нахождения номера строки и номера столбца данной двумерного массива, на пересечении которых находится минимальный элемент.
4. На плоскости заданы координаты N точек. Составить программу для определения числа точек, попавших внутрь окружности с центром в точке (x0,y0) радиуса R.
5. Составить программу для вычисления суммы максимальных значений двумерных массивов A, B и C. Для вычисления максимального значения массива использовать подпрограмму-функцию.
6. Характеристикой строки данной целочисленной матрицы назовем сумму ее положительных четных элементов. Составить программу для размещения строк данной таблицы в соответствие с ростом их характеристик.
7. Дан текстовый файл g.txt. Составить программу для формирования нового файла f.txt, включив в него лишь те строки исходного файла, которые не содержат цифр.
8. Дана матрица. Составить программу для определения всех номеров элементов этой матрицы, имеющих минимальное значение.
9. Составить программу для вычисления суммы минимальных элементов трех двумерных массивов A, B и C. Для вычисления минимального элемента линейной таблицы использовать подпрограмму-функцию.
10. Составить программу для вычисления суммы S = S1 + S2 + S3 , где Si – сумма абсолютных значений i- ого данного одномерного массива. Для вычисления Si использовать подпрограмму-функцию.
11. Составить программу для сортировки данного набора строк по символу с номером k>0. Значение k не превосходит длины самой короткой строки из данного набора.
12. Дан текстовый файл g.txt. Составить программу для подсчета в этом файле количества повторений данного слова.
13. Составить программу для вычисления суммы S = S1 + S2 + S3 , где Si – сумма квадратов значений i-ого данного одномерного массива. Для вычисления Si использовать подпрограмму-функцию.
14. Дана двумерная матрица размером N x N. Составить программу для вычисления суммы абсолютных значений элементов этой таблицы, расположенных справа от главной диагонали.
15. Дан текстовый файл g.txt. Составить программу для подсчета числа строк этого файла, состоящих из одинаковых символов.
16. Характеристикой строки данной целочисленной двумерного массива назовем сумму модулей ее отрицательных нечетных элементов. Составить программу для размещения столбцов данной таблицы в соответствие с ростом их характеристик.
17. Дана матрица N x M. Составить программу для поиска строк с минимальным и максимальным последним элементом. Найденные строки поменять местами.
18. Дан текстовый файл g.txt. Составить программу для определения максимальной длины строки этого файла. Все строки максимальной длины вывести на экран.
19. Дан текстовый файл g.txt. Составить программу для подсчета в этом файле количества строк максимальной длины.
20. Дана матрица размером N x N. Составить программу для вычисления суммы абсолютных значений элементов этой таблицы, расположенных слева от главной диагонали.

Пример компьютерных тестовых заданий.

1. Какой тип данных не относится к целочисленным типам?

а) Integer

б) Real

в) Byte

2. Какая из операций является операцией целочисленного деления?

а) \

б) mod

в) div

3. Определите приоритеты операций: +, *, ^?

а) 1,2,3

б) 2,3,1

в) 3,2,1

4. Какая из операций является операцией целочисленного деления?

а) \

б) mod

в) div

5. s:=’экзамен’;

d:= Length(s);

WriteLn(d);

Какое значение получит переменная d?

а) «э»

б) 7

в) 5

6. b:=Pos(‘ab’,’BCABF’);

Какое значение получит переменная b?

а) 3

б) 5

в) 0

7. s1:=’картография’;

s2:=Copy(s1,6,4);

Какое значение получит переменная s2?

а) «карт»

б) «граф»

в) «картограф»

8. s:=’Спорт’;

Delete(s,5,1);

WriteLn(s);

Какое значение получит переменная s?

а) «орт»

б) «порт»

в) «спор»

9. For i:=1 to 10 do WriteLn(‘тест’);

Сколько раз будет напечатано слово «тест»?

а) 9

б) 10

в) 0

10. For i:=1 downto 10 do WriteLn(‘тест’);

Сколько раз будет напечатано слово «тест»?

а) 9

б) 10

в) 0

11. s:=-1;
```
Repeat
  s:=s+1
Until s > 0;
WriteLn (‘s=’, s);
```
Что будет выведено на экран?
а) s=0
б) s=1
в) s=-1
12. s:=-2;
```
Repeat
  s:=s+1
Until s < 0;
WriteLn (‘s=’, s);
```
Что будет выведено на экран?
а) s=0
б) s=-2
в) s=-1
13. s:=-1;
```
While s > 0 do  s:=s+1;
WriteLn (‘s=’, s);
```
Что будет выведено на экран?
а) s=0
б) s=-1
в) s=1
14. s:=-1;
```
While s < 0 do  s:=s+1;
WriteLn (‘s=’, s);
```
Что будет выведено на экран?
а) s=0
б) s=-1
в) s=1
15. b:=56.125;
```
Write(‘b=’,b:5:2);
```
Что будет выведено на экран (символом «_» обозначен знак пробела)?
а) b=56.12
б) b=56.13
в) b=_ _ _56.12
16. На экран выведено:
b=56.1250
В каком формате был задан вывод числа b=56.125?
а) b:5:3
б) b:2:4
в) b:4:2
17. Что из перечисленного не может быть принято в качестве идентификатора?
а) a12
б) 12a
в) _12a
18. Var a: byte; b: real; c:integer;
Какой из вариантов содержит ошибку?
а) a:=10; b:=5.2; c:=100;
б) a:=-10; b:=-5.2; c:=100;

в) a:=10; b:=-5.2; c:=-100;

19. Const

A:array[1..3,1..4] of integer = ((1, 2, 3, 4),
(5, 6, 7, 8),
(9, -1, 0, -3));

Какое значение имеет элемент a[2,3]?

а) 5

б) -1

в) 7

20. $z = \dfrac{\cos(a^2) + \sin^2(x)}{a + b}$

Какой из вариантов записи функции является правильным?

а) z:=Cos(Sqr(a))+Sqr(Sin(x))/(a+b);

б) z:=Cos(Sqr(a))+Sqr(Sin(x))/a+b;

в) z:= Sqr(Cos(a))+Sin(Sqr(x))/(a+b);

4.2 Итоговый контроль знаний

Итоговый контроль осуществляется после успешного прохождения студентами текущего и промежуточного контроля в виде экзамена.

Перечень теоретических вопросов к экзамену по курсу «Информатика»:

Примерные вопросы к экзамену:

1. Алгоритмы, способы задания и описания.
2. Основные конструкции алгоритмов. Алгоритмы линейной структуры.
3. Основные конструкции алгоритмов. Алгоритмы разветвляющейся структуры.
4. Основные конструкции алгоритмов. Алгоритмы циклической структуры «До».
5. Основные конструкции алгоритмов. Алгоритмы циклической структуры «Пока».
6. Языки программирования. Алфавит языка Паскаль. Структура программы на языке Паскаль.
7. Правила записи арифметических выражений.
8. Переменные, их описание в программе. Скалярные типы данных. Оператор присваивания. Приоритет операции.
9. Операторы WRITE, WRITELN. Процедура ввода READ, READLN.
10. Оператор безусловного перехода GOTO.
11. Оператор условного перехода IF.
12. Оператор CASE. Операции сравнения.
13. Операторы повторений. Оператор цикла FOR.
14. Операторы повторений. Оператор цикла WHILE.
15. Операторы повторений. Оператор цикла REPEAT.
16. Массивы. Работа с одномерными массивами.
17. Сортировка массивов.
18. Массивы. Работа с двумерными массивами.
19. Работа со строковыми переменными.
20. Множества. Операции над множествами.
21. Подпрограммы общего вида (PROCEDURE).
22. Подпрограммы-функции (FUNCTION).
23. Рекурсия. Рекурсивные программы.
24. Локальные и глобальные переменные.
25. Формальные и фактические параметры в процедуре.
26. Понятие о файлах. Процедуры открытия и закрытия файлов.
27. Процедуры и функции работы с файлами.
28. Текстовые файлы.

29. Записи.
30. Записи с вариантами.

Экзамен сдается в конце первого семестра. Форма сдачи экзамена – устная. Необходимым условием допуска на экзамен является сдача всех лабораторных и практических работ. Экзаменационный билет содержит два теоретических вопроса и три тест-задачи различной степени сложности. Экзамен проходит в письменной форме с последующей индивидуальной беседой преподавателя с экзаменующимся. На письменную работу над билетом отводится 1 час. Каждый пункт оценен определенным количеством баллов, билет содержит шкалу перевода баллов в традиционную пятибальную оценку.

Пример экзаменационного билета:

АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Утверждено на заседании кафедры	Кафедра МАиМ
23 декабря 2011 г. протокол №4	Факультет МиИ
Заведующий кафедрой	Курс I
Утверждаю:_____	Дисциплина Информатика

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Алгоритмы, способы задания и описания.
2. Операторы повторений. Оператор цикла REPEAT.
3. Вычислить $S = \sum_{i=1}^{n} \frac{1}{i+5}$.
4. Определить функцию r.

$$r = \sqrt{S^2 + t}; \quad S = \frac{5x^2 + 2e^y}{3x + \sin^2 y^3}; \quad t = \frac{x^2 + 2y - 3}{2\sin^2 x + \frac{1}{x+y}}$$

Итоговая оценка выставляется студенту с учетом общего рейтинга по дисциплине и набранных за семестр баллов, включая баллы за тестирование.

Шкала перевода баллов в итоговую оценку по дисциплине:

Итоговый рейтинг по дисциплине, Кол-во баллов за семестр	Оценка по пятибалльной шкале
91-100	5
75-90	4
51-74	3
менее 50	2

5 Интерактивные технологии и инновационные методы, используемые в образовательном процессе

При преподавании дисциплины «Информатика» используются следующие инновационные технологии и методы: чтение проблемных лекций, применение мультимедийного проектора при чтении лекций, «мозговой штурм», использование ресурсов сети Internet и электронных учебников при самостоятельной и аудиторной работе студентов, дискуссии в обсуждении проблемных ситуаций. Детальная схема занятий, проводимых с использованием интерактивных методов обучения представлена в п. 1.6.