

Федеральное агентство по образованию
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГОУВПО "АмГУ"
Факультет математики и информатики

УТВЕРЖДАЮ

Зав. кафедрой МАиМ

_____ Т.В. Труфанова

« ___ » _____ 2007 г.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ
СИСТЕМ И СЕТЕЙ

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ

для специальности 010501 – "Прикладная математика и информатика"

Составители: Т.Г Решетнёва

Благовещенск

2007 г.

ББК

*Печатается по решению
редакционно-издательского
совета
факультета математики и
информатики
Амурского государственного
университета*

Решетнёва Т.Г.

Программное обеспечение вычислительных систем и сетей:

Учебно-методический комплекс по дисциплине для студентов АмГУ очной формы обучения специальности 010501 "Прикладная математика и информатика". – Благовещенск: Амурский гос. ун-т, 2007. – 68 с.

Учебно-методический комплекс по дисциплине "Программное обеспечение вычислительных систем и сетей" предназначен для студентов специальности 010501 – "Прикладная математика и информатика" очной формы обучения, призван помочь ведущим преподавателям и студентам в организации процесса изучения дисциплины. Комплекс содержит рабочую программу дисциплины, материалы для проведения практических занятий, справочный материал и библиографический список.

© Амурский государственный университет, 2007

© Кафедра математического анализа и моделирования, 2007

I. РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Рабочая программа по дисциплине "Программное обеспечение вычислительных систем и сетей" для специальности 010501 – "Прикладная математика и информатика".

Курс 4. Семестр 5–6. Лекции 32 час. Экзамен 6 семестр. Практические (семинарские) занятия 32 час. Зачет 5 семестр. Лабораторные занятия (нет). Самостоятельная работа час. Всего часов.

Составитель Т.Г. Решетнёва, доцент. Факультет математики и информатики. Кафедра математического анализа и моделирования. Благовещенск, 2007 г.

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ, ЕЕ МЕСТО В УЧЕБНОМ ПРОЦЕССЕ

1.1 Цель преподавания дисциплины.

Основной целью дисциплины является обучение студентов теоретическим основам и практическим навыкам работы с программными комплексами вычислительных систем и сетей, ориентированными на решение различного рода задач; овладение основными приемами и методами программного управления средствами вычислительной техники; ознакомление с методами и стандартами разработки программного обеспечения.

1.2 Задачи изучения дисциплины.

В результате изучения дисциплины студент должен иметь представление:

- об архитектуре современных ЭВМ и базовых принципах построения средств вычислительной техники;
- о методах и стандартах разработок программного обеспечения;
- о современных методах и средствах программирования;

- о существующих на сегодняшний день пакетах программ для решения различных прикладных и системных задач.

знать и уметь использовать:

- базовые устройства современных вычислительных систем;
- методы и стандарты разработок программного обеспечения;
- решать задачи, возникающие в процессе сопровождения и эксплуатации программных средств;
- современные методы и средства программирования;
- существующие на сегодняшний день пакеты программ для решения прикладных задач пользователя;
- решать задачи, возникающие в процессе сопровождения и эксплуатации программных средств;
- современные численные методы решения математических задач на ЭВМ;
- приемы и методы разработки сетевых приложений;
- инструментальные средства (ИС) создания серверной части сетевых приложений;
- инструментальные средства создания клиентской части сетевых приложений;

должен приобрести навыки:

- разработки, отладки, тестирования и документирования программного обеспечения;
- работы с пакетами программ и системами программирования; программировать в их среде, использовать встроенные функции;
- разработка информационных баз данных, принципы работы с базами данных;
- работы с графическими редакторами (векторная и растровая графика);
- решать в среде данных пакетов различного рода задачи и представлять результаты исследований;

- формирование знаний, умений и навыков в области теории информационного моделирования и проектирования баз данных;
- разработки серверной и клиентской часть сетевых приложений;
- сопровождения сетевых приложений.

1.3 Перечень дисциплин с указанием разделов (тем), усвоение которых студентами необходимо при изучении данной дисциплины.

Дисциплина базируется на знаниях, полученных в рамках курсов «Информатика», «Языки программирования и методы трансляции», «Практикум на ЭВМ», «Системное и прикладное программное обеспечение», «Базы данных и экспертные системы», «Операционные системы и сетевые технологии».

1.4 Структура дисциплины

Дисциплина рассчитана на два семестра. В пятом семестре рассматриваются современные тенденции развития программного обеспечения ЭВМ и сетей, вопросы инженерии программного обеспечения, основы программирования прикладных математических задач с помощью пакета MATLAB и программирование баз данных в среде Delphi. В шестом семестре изучаются вопросы компьютерной графики, студенты знакомятся с популярными графическими редакторами, их назначением и использованием, рассматриваются программно-технические комплексы создания и ведения ГИС. Также акцент делается на изучение средств анализа и управления компьютерными сетями, серверного и клиентского программного обеспечения.

2. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

2.1 Федеральный компонент.

Нет.

2.2 Наименование тем, их содержание, объем в лекционных часах.

ТЕМАТИЧЕСКИЙ ПЛАН ЛЕКЦИОННЫХ ЗАНЯТИЙ

Наименование темы		Количество часов
1		2
1	5 семестр Современные тенденции развития программного обеспечения ЭВМ и сетей.	4
2	Прикладное программное обеспечение ЭВМ.	2
3	Инженерия программного обеспечения.	12
4	Основы программирования в среде MATLAB	6
5	Построение баз данных в Delphi	8
ИТОГО ЗА 5 СЕМЕСТР		32
6	6 семестр Компьютерная графика. Графические редакторы.	6
7	САПР	6
8	Геоинформационные системы, их назначение создание и использование. Программное обеспечение ГИС.	6
9	Средства анализа и управления сетями.	4
10	Основы технологии «клиент-сервер». Языки гипертекстовой разметки. Серверное программное обеспечение. Клиентская часть приложений.	10
ИТОГО ЗА 6 СЕМЕСТР		32

2.3 Практические и семинарские занятия, их содержание и объем в часах.

ТЕМАТИЧЕСКИЙ ПЛАН ПРАКТИЧЕСКИХ ЗАНЯТИЙ

Наименование темы		Количество часов
1		2
1	5 семестр Операционная система MS DOS. Файлы конфигурации и командные файлы config.sys, autoexec.bat. Основные команды.	2
2	Операционная система Windows. Реестр.	2
3	Знакомство с пакетом MATLAB. Встроенные функции, их назначение и использование для решения различных задач	2

4	Основы программирования в среде MATLAB	4
5	Построение баз данных в Delphi	6
ИТОГО за 5 семестр		16
6	Графические редакторы (Corel, Photoshop и др.)	4
7	САПР (AutoCAD)	4
8	ГИС (ArcView, ArcGIS, MapInfo, Панорама)	4
9	Форматирование текста и списков. Форматирование таблиц. Работа с формами. Работа с объектами. Создание динамической Web-страницы. Разработка Web-приложения.	4
ИТОГО за 6 семестр		16

При выполнении практических работ по данному курсу студенты должны ознакомиться с основными принципами построения, функционирования и использования современных программных комплексов, овладеть основными приемами и методами программного управления средствами вычислительной техники.

2.4 Лабораторные занятия, их наименование и объем в часах.

Лабораторные занятия не предусмотрены.

2.5 Самостоятельная работа студентов.

Самостоятельная (внеаудиторная) работа студентов состоит в проработке лекционного материала, подготовке к практическим и контрольным работам.

В качестве самостоятельной работы по дисциплине «Программное обеспечение вычислительных систем и сетей» студентам предлагается рассмотреть и изучить следующие вопросы:

1.Связь программного обеспечения с аппаратным, виды аппаратных компонент. Управление на физическом и логическом уровне, проверка готовности, обработка прерываний.

2.Системное и прикладное программное обеспечение – классификация, характеристика, использование.

3.Назначение, функции и состав системного программного обеспечения, вопросы надежности и устойчивости.

4. Векторная и растровая графика.
5. Защита информации. Методы защиты информации.
6. Интегрированные среды разработки ПО. CASE-технологии.
7. Программное обеспечение разработки АИС.
8. Этапы жизненного цикла разработки программного обеспечения.
9. Функции управления программной инженерией.
10. История развития инструментальных средств поддержки разработки ПО.
11. Знакомство с публикациями в периодических изданиях: «Мир ПК», «Компьютерпресс», «Домашний компьютер» и др.
12. Работа с ресурсами Internet и с дополнительной литературой.
13. Автоматизированная картографическая система (АКС), цифровая модель рельефа (ЦМР), растровая модель данных, векторная модель данных, топологическая и нетопологические модели.
14. Вычислительная геометрия, оверлейные операции, масштаб.
15. Проекция, ГИС, ГИС-технология, геоизображение, геоид, анаморфоза, примитивы 1, 2 и 3-го порядков, пиксел, растровое представление, алгоритм, конвертация данных.
16. Алгоритмы сжатия пространственной информации.
17. Структура файлов Arc View (векторных, растровых, атрибутивных баз данных).
18. Форматы растровых и векторных файлов.

Контроль над выполнением самостоятельной работы осуществляется путём проведения контрольных работ, опросов на практических занятиях и написании расчетных работ, а также данные вопросы включены в перечень вопросов к зачету и экзамену.

2.6 Вопросы к зачету.

5 семестр

1. Аппаратные и программные средства ВТ.
2. Определение и назначение программного обеспечения (ПО). Этапы развития ПО.

3. Классификация ПО для различных классов: общее (системное) и специальное ПО ЭВМ общего назначения.
4. Определение операционной системы (ОС). Уровни ОС и распределение функций по уровням. Организация управления памятью. Управление процессом выполнения программы.
5. Обзор архитектур современных операционных систем. Сетевые и распределенные ОС. Функциональные компоненты сетевой ОС. Концепции распределенной обработки в сетевых ОС.
6. Особенности организации вычислительных процессов. Сравнительная характеристика, условия применения.
7. Структура пакетов программ (ПП). Этапы разработки ПП.
8. Классификация качеств ПО, требования к качеству в различных прикладных областях.
9. Принципы инженерии программного обеспечения.
10. Проектирование ПО и его цели.
11. Метод пошаговой детализации и метод структурного программирования. Параллельное ПО. Объектно-ориентированное проектирование.
12. Спецификация, использование, качества, классификация.
13. Цели и требования верификации. Тестирование ПП, методы тестирования. Отладка программ.
14. Методы проектирования тестовых наборов данных. Сборка программ при тестировании.
15. Критерии надежности ПП. Модели надежности, их классификация.
16. Инструментальные средства поддержки разработки ПО.
17. MATLAB - назначение, основные функциональные возможности.
18. Особенности программирования в среде MATLAB.
19. Организации циклов в MATLAB.
20. Внешние расширения системы (m-файлы).
21. Встроенные операторы и функции.

22. Графические возможности. Оператор plot.
23. Что такое база данных и СУБД. Структура базы данных. Типы данных.
24. Создание базы данных в среде Delphi (создание и изменение структуры базы, открытие и закрытие базы данных, заполнение базы данных информацией, уничтожение записей в базе данных, поиск данных).
25. Компоненты для работы с базами данных: DataSource, Table, Query, DBGrid, DBNavigator, DBComboBox, DBLookupListBox, DBLookupComboBox.
26. Интегрированные среды разработки ПО. CASE-технологии.

2.7 Вопросы к экзамену

6 семестр

1. Программное обеспечение ПК. Классификация ПО. Назначение, функции и состав программного обеспечения.
2. Понятие информационной системы. Виды информационных систем.
3. Признаки классификации информационных систем.
- 4.
5. Интегрированные среды разработки ПО. CASE-технологии.
6. Назначение и основные функции Delphi. Основные понятия Delphi. Структура среды программирования Delphi. Интегрированная среда разработки программ.
7. Понятие базы данных. Классификация баз данных. Структура систем баз данных.
8. Логическое проектирование как этап разработки базы данных; особенности концептуального проектирования.
9. Компонент DBNavigator, основные свойства, принцип работы.
10. Использование Database Desktop для создания новой таблицы.
11. Доступ к файлу данных (таблице). Компоненты Table и Data. Компонент DataSource, основные свойства, примеры.
12. Компонента DBGrid, основные свойства, примеры работы.
13. Структуры хранения и методы доступа к информации: файлы, страницы, индексы, хэши.

14. Технологии сжатия информации.
15. Языки программирования баз данных.
16. Работа с внешними данными с помощью технологии ODBC (BDE).
17. Объектно-ориентированное программирование в среде баз данных.
Использование SQL-запросов. SQL сервер. Использование SQL для выборки данных из таблицы, создание SQL-запросов.
18. Компьютерная графика, ее использование в различных отраслях человеческой деятельности.
19. Понятие графической информации, способы ее представления и обработки в ЭВМ. Растровая, векторная, фрактальная и 3D-графика.
20. Графические редакторы. Форматы графических файлов.
21. Системы инженерной графики. САПР. AutoCAD, назначение, характеристика основных функциональных возможностей.
22. Построение основных графических примитивов. Способы привязки при построении графических образов. Возможности редактирования графических образов в среде AutoCAD.
23. Работа с текстом в AutoCAD. Штриховка и заливка цветом графических объектов. Нанесение размеров и допусков.
24. Геоинформационные системы. Составные части ГИС. Базовые модели данных в ГИС.
25. Особенности организации данных в ГИС. Типы данных и их источники. Проблемы ввода данных. Технические средства ввода данных.
26. Технологии ввода данных (растровых и векторных). Технологии векторизации растровых данных.
27. ГИС настольного типа. Общая характеристика ArcView. Понятие проекта, вида, их создание, состав и редактирование.
28. Понятие темы. Виды тем, их создание, расположение, редактирование в ArcView. Таблицы свойств атрибутов, их привязка и использование в ArcView.

29. ГИС ArcGIS, MapInfo, Панорама, назначение, принципы работы.
30. Основные понятия компьютерных систем. Эволюция вычислительных систем. Предпосылки создания компьютерных сетей.
31. Основные программные и аппаратные компоненты сети. Проблемы построения сетей. Физическая и логическая структуризация сети. Сетевые службы.
32. Средства анализа и управления сетями. Функции и архитектура систем управления сетями. Архитектура систем управления сетями.
33. Стандарты систем управления. Стандарты систем управления SNMP, OSI.
34. Мониторинг и анализ сетей.
35. Основы технологии «клиент-сервер». Основы технологии «клиент-сервер». Схема взаимодействия клиента и сервера. Серверы приложений: типы, назначение, функции.
36. Серверы Интернет. Web-сервер, его функции и предъявляемые к нему требования. Microsoft Internet Information Services (IIS). Web-сервер Apache.
37. Основы Web-программирования: основные понятия и термины. Web-дизайн и Web-программирование.
38. Принципы гипертекстовой разметки. Структура гипертекстовых документов.
39. Основы разработки сетевых приложений. Принципы построения серверной части программного обеспечения.
40. Основные задачи, выполняемые серверными программами.
41. Средства создания программ, выполняемых на стороне сервера. Их характеристика и назначение.
42. Характеристика типовых задач, решаемых клиентской частью приложений. Функциональные возможности клиентской части.
43. Обзор инструментальных средств разработки программ, выполняющихся на стороне клиента. Их назначение и возможности.

3. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

3.1 Перечень обязательной (основной) литературы

1. Арапчор Т.А., Далаа С.М., Монгуш М.М. Основы программирования в Delphi. – Кызыл: Изд-во ТывГУ, 2004.

2. Барановская Т.П., Лойко В.И., Семенов М.И., Трубилин А.И. Архитектура компьютерных систем и сетей: Учебное пособие/ Под ред. В.И. Лойко.-М.: Финансы и статистика, 2003.-256с.

3. Бугаевский Л.М., Цветков М.Я. Геоинформационные системы: Учебное пособие для ВУЗов.-М.: 2000.-222с.

4. Гецци К., Джазайери М., Мандриоли Д. Основы инженерии программного обеспечения. 2-е изд.: Пер. с англ. – СПб.: БХВ-Петербург, 2005. – 832 с.

5. Джеймс Ли, Brent Уэр. Использование Linux, Apache, MySQL и PHP для разработки Web-приложений. – М.: Вильямс, 2004. – 432 с.

6. Иванников А.Д., Кулагин В.П., Тихонов А.Н., Цветков В.Я. Геоинформатика.-М.: МАКС Пресс, 2001.-349с.

7. Компьютерные сети. Учебный курс, 2-е изд. – Microsoft Press, Русская редакция, 1998.

8. Королев Л.Н., Миков А.И. Информатика. Введение в компьютерные науки: Учеб.-М.: Высш.шк., 2003.-341с.

9. Иртегов Д. Введение в операционные системы. – СПб.: БХВ-Петербург, 2002.

10. Максимов Н.В., Попов И.И. Компьютерные сети.-М.: ФОРУМ:ИНФРА-М, 2003.-336с.

11. Матросов А.В. HTML 4.0 / А.В. Матросов, А.О. Сергеев, М.П. Чаунин. – СПб.: БХВ-Петербург, 2003. – 672 с.: ил.

12. Монахов С.В., Цветков В.Я., Савиных В.П. Общая геоинформатика.-М.: МАКС Пресс, 2004.-336с.

13. Петюшкин А.В. HTML в Web-дизайне. – СПб.: БХВ-Петербург, 2004

14.Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. – Спб.: Питер, 2002.

15.Петров В.Н. Информационные системы.-СПб.: Питер, 2002.-688с.

16.Фаронов В.В., Система программирования Delphi. – Спб.: БХВ-Петербург, 2005. – 912с.

17.Юров В.И. Assembler.-СПб.: Питер, 2003.-624с.

3.2 Перечень дополнительной литературы

1. Алексахина Л.П., Поляков А.К. Программирование на языке Ассемблера КР580 / Под ред. А.А.Дерюгина.-М.: Моск. энерг. ин-т, 1986.-96 с.

2. Бродин В.Б., Шагурин И.И. Микропроцессор i486. Архитектура, программирование, интерфейс. -М.: "Диалог-МИФИ", 1993, -240с.

3. Вычислительные машины и системы: Учебник для вузов/В.Д.Ефремов и др. - М.: Высш.шк., 1993. - 292с.

4. Глушаков С.В., Жакин И.А., Хачиров Т.С. Программирование Web-страниц: Учебный курс. – Харьков: Фолио, 2002.

5. Глушаков С.В., Клевцов А.Л., Терехилов С.А. Программирование на Delphi 5.0 – Харьков: Фолио, 2002.

6. Иртегов Д.В. Введение в операционные системы.-СПб.: БХВ-Петербург, 2002.-624с.

7. Каган Б.М. Электронные вычислительные машины и системы. - М.: Энергоатомиздат, 1991. – 592 с.

8. Норенков И.П., Маничев В.Б. Основы теории и проектирования САПР.: Уч. для втузов по спец. «Вычислительные машины, компл., сист. и сети.». – М.: Высш. шк., 1990. – 335с.

9. Разработка САПР. В 10 кн. – М.: Высш. шк., 1990.

10.Сван Т. Освоение Turbo Assembler: Второе издание. – Киев: Диалектика, 1996, 544 с.

11.Соколов А.П. Системы программирования: теории, методы, алгоритмы: Учеб.пособие.-М.: Финансы и статистика, 2004.-320с.

12.Том Армстронг. Active X: создание Web-приложений. – Киев:

Издательская группа BHV, 1998.

13. <http://www.php.spb.ru>

14. <http://www.javaportal.ru>

3.3 Перечень методических пособий

1. Методические указания к лабораторным работам по курсу «Разработка САПР» для студентов специальности 220300 «Системы автоматизированного проектирования» дневной формы обучения /Воронеж. гос. техн. ун-т; Сост. М.В. Питолин, А.В. Питолин, Д.А. Севостьянов. Воронеж, 1996 - 28с.

2. Семочкин А.Н. Вычислительная техника.-Благовещенск,БГПУ.-2001.-67с.

3.4 Перечень наглядных и иных пособий.

1. Карточки с заданиями к практическим работам / *Т.Г. Решетнёва*.

3.5 Средства обеспечения освоения дисциплины.

1. Среда программирования Borland Delphi.

2. Пакет MatLab.

3. Программное обеспечение ГИС: ArcView, ArcGIS, MapInfo, Панорама.

4. Графические редакторы: Corel, Fotoshop и др.

5. САПР AutoCAD.

4. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Компьютерный класс кафедры МАиМ.

5. КРИТЕРИИ ОЦЕНКИ ЗНАНИЙ

5.1 Текущий контроль знаний

Текущий контроль за аудиторной и самостоятельной работой обучаемых осуществляется посредством устного опроса по изучаемым разделам и проверки отчетности по практическим работам и индивидуальным заданиям. Предполагает систематическое проведение контрольных работ, а также

выполнение типового расчета в 5 и 6 семестрах, которые включают в себя задания по всему семестру.

5.2 Требования к знаниям студентов, предъявляемые на зачете.

Зачёт сдаётся в конце 5 семестра. Форма сдачи – устная. Необходимым допуском на зачет является сдача всех заданий по практическим работам, положительные оценки за промежуточные контрольные работы. На зачете студенту предлагается ответить на два вопроса из предлагаемого списка и ответить на дополнительные вопросы по теме.

Знания студента оцениваются на «зачтено» при полном ответе на вопрос и удовлетворительном ответе на дополнительные вопросы преподавателя.

Оценка «не зачтено» ставится при незнании вопроса, предлагаемого студенту на зачетном занятии.

5.3 Требования к знаниям студентов, предъявляемые на экзамене.

Экзамен сдаётся в конце 6 семестра. Форма сдачи – устная. Необходимым условием допуска на экзамен является выполнение всех заданий по практическим работам по дисциплине, сдача расчётных работ. В экзаменационный билет входят три вопроса из различных разделов курса, на которые студент должен дать развернутый ответ. Показать знание теории по данной части курса, продемонстрировать свободную ориентацию в материале, знание понятий и терминологии, ответить на дополнительные вопросы.

Знания студента оцениваются на «отлично» при полном изложении теоретического материала экзаменационного билета, ответах на дополнительные вопросы со свободной ориентацией в материале и других литературных источниках.

Оценка «хорошо» ставится при твердых знаниях студентом всех разделов курса (в пределах конспекта лекций) и при преимущественно правильных ответах на дополнительные вопросы части (допускаются нетвердое знание одного – двух вопросов билета).

Оценку «удовлетворительно» студент получает, если дает неполные ответы на теоретические вопросы билета, показывая поверхностное знание учебного материала, владение основными понятиями и терминологией; при неверном ответе на билет или на дополнительные вопросы. Допускается полное незнание одного из вопросов билета.

Оценка «неудовлетворительно» выставляется за незнание студентом одного из разделов курса, если студент не дает ответы на теоретические вопросы билета, показывая лишь фрагментарное знание учебного материала, незнание основных понятий и терминологии, при полном незнании двух вопросов из трех предлагаемых в билете.

II. ГРАФИК САМОСТОЯТЕЛЬНОЙ УЧЕБНОЙ РАБОТЫ СТУДЕНТОВ ПО ДИСЦИПЛИНЕ НА КАЖДЫЙ СЕМЕСТР С УКАЗАНИЕМ ЕЕ СОДЕРЖАНИЯ, ОБЪЕМА В ЧАСАХ, СРОКОВ И ФОРМ КОНТРОЛЯ

График самостоятельной учебной работы студентов по дисциплине (с указанием ее содержания, объема в часах, сроков и форм контроля) приведен в рабочей программе дисциплины.

III. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ СЕМИНАРОВ, ДЕЛОВЫХ ИГР, РАЗБОРУ СИТУАЦИЙ И Т. П. СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ (ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ)

Проведение деловых игр, разбор ситуаций и т. п. рабочей программой дисциплины не предусмотрены.

Форма проведения практического семинара: а) приветствие студентов, 1 мин.; б) определение личного состава студенческой группы, 4 мин.; в) объявление тематики и вопросов практического семинара, 1 мин.; г) выполнение заданий, 70 мин.; д) подведение итогов семинара, 13 мин.; е) прощание со студентами, 1 мин.

Список рекомендуемой литературы (основной и дополнительной) приведен в рабочей программе.

IV. КРАТКИЙ КОНСПЕКТ ЛЕКЦИЙ (ПО КАЖДОЙ ТЕМЕ) ИЛИ ПЛАН-КОНСПЕКТ

Тема 1. Современные тенденции развития программного обеспечения ЭВМ и сетей.

Введение. Общие сведения о предмете. Связь дисциплины с другими курсами учебного плана. Цели и задачи дисциплины.

Современные тенденции развития программного обеспечения ЭВМ и сетей. Перспективы развития технологических средств разработки ПО. Тенденции развития инструментальных средств разработки.

Тема 2. Прикладное программное обеспечение ЭВМ

Классификация прикладных программ, назначение и функциональные возможности. Пакеты прикладных программ (ППП), их назначение, организация и использование. Разработка ППП.

Задача проектирования программных систем. Организация процесса проектирования программного обеспечения (ПО). Цели проектирования ПО.

Принципы проектирования ПО. Использование декомпозиции и абстракции при проектировании ПО. Процедурная абстракция. Абстракция данных. Специфики процедур и данных. Декомпозиция системы.

Тема 3. Инженерия программного обеспечения

Роль и принципы инженерии программного обеспечения. Жизненный цикл программного обеспечения. Определение. Проектирование. Реализация. Внедрение. Эксплуатация.

Классификация, требования к качеству в различных прикладных областях. Показатели качества программного обеспечения. Надёжность программного обеспечения. Факторы, снижающие надёжность работы программ. Программные методы повышения надёжности работы программ.

Испытания и сопровождение ПО. Методы защиты программ и данных.

Проектирование и архитектура программного обеспечения.
Спецификация. Верификация.

Процесс производства программного обеспечения.

Подходы к программированию. Процедурное программирование.
Функциональное программирование. Логическое программирование.
Параллельное программирование. Объектно-ориентированное
программирование.

Основные методы разработки программ. Нисходящее и восходящее
проектирование. Модульное программирование.

Программирование с защитой от ошибок. Краткий обзор основных
технологий программирования. НИРО технология. Структурное
программирование. R - технология программирования. Методика Джексона.
Коллективные методы разработки программ. Метод главного программиста.

Отладка. Типичные источники ошибок. Комплексная отладка
программ. Определение пропускной способности программы.

Тестирование. Этапы тестирования. Нисходящее и восходящее
тестирование. Автоматизация тестирования.

Документирование и оценка качества программных продуктов. Состав
документации на программный продукт. Государственные стандарты
документации на программный продукт.

Инструментальные средства поддержки разработки и программные
среды.

Технологические средства разработки программного обеспечения.
Инструментальные среды разработки. Назначение, возможности, состав
инструментальной среды. Проектирование интерфейса с пользователем.
Структуры диалога. Поддержка пользователя. Библиотеки визуальных
компонентов. События. Потоки. Динамически связываемые библиотеки.

Современные средства разработки, CASE-системы. Что такое CASE-
система. Классификация CASE-систем. Характерные программные средства

типичной верхней CASE-системы. Методы спецификации программ в CASE-системах. Методика IDEF0 на примере Logic Works' BPWin.

Современные технологии. Развитие современных технологий интегрирования программных средств OLE. Основы OLE. Связь с библиотеками OLE. COM-интерфейс. Интерфейс модели составного объекта. Услуги интерфейса IUnknown. Назначение реестра Windows '95 и Windows NT применительно к OLE. Компонент «Фабрика классов».

Технология COM. Технология ActiveX. Технология MTS. Технология CORBA.

Проектирование интерфейса с пользователем. Цели проектирования интерфейса с пользователем. Процессы ввода–вывода. Сообщения. Подсказки. Помощь.

Диалог. Структуры диалога (командный язык, запрос–ответ, меню, экранные формы). Понятие дружественного интерфейса. Многооконные интерфейсы. Окна, пиктограммы, прямое манипулирование объектами.

Реализации интерфейсов с пользователем с использованием графических пакетов.

Тема 4. Основы программирования в среде MATLAB

Основные возможности MATLAB, назначение, использование. Средства программирования. Встроенные функции пакета. Графические возможности MATLAB. Тулбоксы, их функции.

Общие принципы работы с пакетом Matlab

Matlab - интерпретатор , что означает последовательное выполнение вводимых операторов без предварительной компиляции. Matlab имеет собственную оболочку и язык для разработки и выполнения программ. Общий вид среды разработки показан на рис. 1.1.

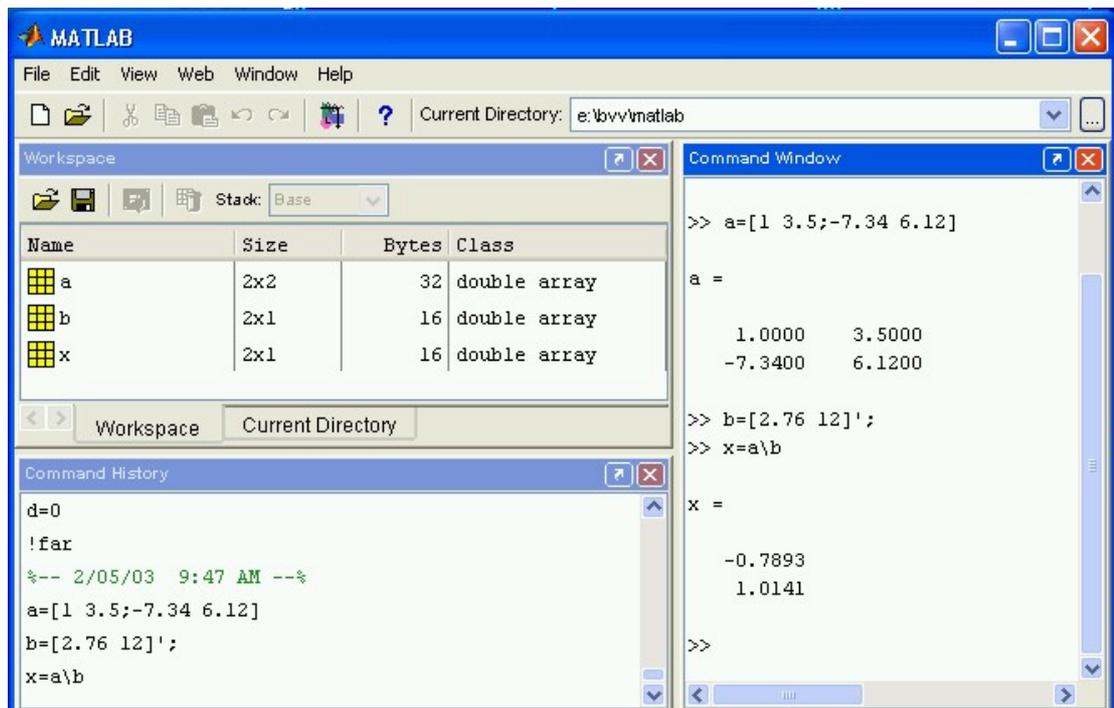


Рис. 1.1 Среда разработки Matlab

По умолчанию открыто три окна. Окно Command Window предназначено для ввода команд, в окне Workspace отображаются переменные, хранящиеся в памяти. Окно Command History протоколирует вводимые команды. Операторы Matlab могут вводиться в командном окне или из файлов с именами *.m.

Например, если в директории e:\bvv\matlab имеется файл name.m с набором операторов Matlab, то для его выполнения необходимо выполнить следующие действия.

1. Ввести в командном окне Matlab команду `path(path,'e:\bvv\matlab');`. Эта команда добавляет к пути поиска m-файлов путь e:\bvv\matlab.

2. После добавления пути к переменной path можно запускать программу, набрав в командном окне name (без .m).

При вводе команды name Matlab выполняет следующие действия.

1. Просматривает список переменных в памяти и, если в памяти хранится переменная с именем name, ее значение выводится на экран.

2. Если переменная с именем name не найдена, далее просматриваются все файлы в текущей директории. Текущая директория определена в окне Current Directory среды разработки. Сменить текущую директорию можно командой cd. Например, `cd e:\bvv\matlab`. Вывести текущую директорию на экран можно командой pwd. Если файл name.m найден, он запускается на выполнение.

3. Если в текущей директории файл name.m не обнаружен, тогда просматриваются все пути из переменной path и первый найденный файл name.m запускается на выполнение.

Следует обратить особое внимание на имена создаваемых m-файлов. Дело в том, что в Matlab в директории %MATLAB\toolbox (%MATLAB - путь

к директории, где установлен Matlab) содержится очень большое количество файлов с именами *.m, а пути на все поддиректории директории toolbox содержатся в переменной path. Команда path(path,'e:\bvv\matlab') добавляет путь e:\bvv\matlab в конец переменной path, поэтому возникает опасность дублирования имен, а Matlab запускает на исполнение первый найденный файл. Можно добавлять свой путь поиска в начало или в конец переменной path командой addpath. Например, чтобы добавить путь e:\bvv\matlab в начало переменной path нужно ввести команду addpath e:\bvv\matlab -begin, а команда addpath e:\bvv\matlab -end поставит этот путь в конец переменной path. Удалить путь из переменной path можно командой rmpath e:\bvv\matlab.

В любой момент можно получить справку о любой функции matlab с помощью команды help. Например, чтобы получить информацию о синтаксисе команды addpath, необходимо ввести команду help addpath.

Получить список переменных, хранящихся в памяти, позволяет команда whos (или ее короткая форма who). Удалить переменную var из памяти можно командой clear var, а удалить все переменные из памяти clear all. Команда format управляет форматом вывода переменных на консоль.

Переменные и знаки операций

Переменные в Matlab не объявляются, как в других языках программирования, а вводятся по ходу выполнения программы. По умолчанию все переменные являются матрицами из вещественных чисел типа double (аналог типа double в Си).

Например, ввести матрицу 2x2 можно командой a=[1 3.5;-7.34 6.12], как показано на рис. 1.1. Если после введенной команды не поставить точку с запятой, то результат выводится на консоль. На рис. 1.1 введены две матрицы a - размером 2x2 и b - 2x1. С помощью знака операции \ (обратная косая) решить систему линейных уравнений, определенных матрицей a и вектором правых частей b. Решение помещается в переменную x и выводится на консоль. Можно проверить, что x действительно решение командой a*x-b - на консоли увидите переменную ans 2x1 с нулевыми элементами. По умолчанию результат помещается в переменную с именем ans.

Обращаются к матричным элементам с помощью круглых скобок, а индексы нумеруются с единицы. Например a(2,1) - матричный элемент из второй строки и первого столбца. Вводимые матрицы могут быть комплексными, для обозначения мнимой единицы зарезервированы две переменные - i и j. Однако по ходу выполнения программы эти переменные могут быть переопределены.

Например ввода комплексной матрицы ac=[1+2*i 4*i;3.6 -7.35+3.67*i].

Некоторые функций для генерирования матриц:

a1=ones(5,5) - матрица 5x5 с единичными матричными элементами,

a2=zeros(4,6) - матрица 4x6 с нулевыми матричными элементами,

a3=eye(4,4) - единичная матрица 4x4,

`a4=rand(7,8)` - 7x8 матрица со случайными матричными элементами из интервала [0-1].

Размерность массивов в `matlab` произвольная, т.е. можно заводить массивы 3-х мерные, 4-х мерные и т.д. Очень часто при оперировании с массивами используется символ `:`.

Например, `c=a(:,1)`; - в переменную `c` занесены все строки первого столбца матрицы `a`, или `c1=a(2:4,2)`; - в `c1` занесены строки со второй по четвертую второго столбца матрицы `a`.

С помощью оператора `x=0:0.1:2*pi`; (`pi` - предопределенная переменная) заводится новая переменная-строка, элементами которой являются все значения от нуля до 2π с шагом 0.1.

В `matlab` определены следующие знаки арифметических операций:

- + сложение,
- вычитание,
- * матричное умножение,
- / матричное деление,
- \ обратное деление,
- ^ возведение в степень,
- ' транспонирование
- .* поэлементное умножение,
- ./ поэлементное деление,
- .\ обратное поэлементное деление,
- .^ поэлементное возведение в степень
- .' поэлементное транспонирование.

Описание знаков операций.

+ Сложение или унарный плюс. При сложении матрицы должны иметь одинаковые размеры. К матрице любого размера можно прибавлять скаляр.

- Вычитание или унарный минус. При вычитании матрицы должны иметь одинаковые размеры. Из матрицы любого размера можно вычитать скаляр.

* Матричное умножение. Сомножители должны иметь правильные размеры.

/ Деление матриц. A/B эквивалентно $A*inv(B)$, `inv` - функция для вычисления обратной матрицы.

^ В выражении C^p `p` должно быть скаляром. Если `p` целое, тогда `C` перемножается `p` раз. Если `p` отрицательное, то от `C` берется обратная матрица. Если `p` вещественное, тогда сначала вычисляются собственные значения и собственные вектора матрицы $C[V,D]=eig(C)$; где `V` - матрица, содержащая в столбцах собственные вектора матрицы `C`, а `D` - диагональная матрица с собственными значениями `C` на диагонали. Результат возведения в степень вычисляется так $C^p=V*D.^p/V$. Описание операции `.^`

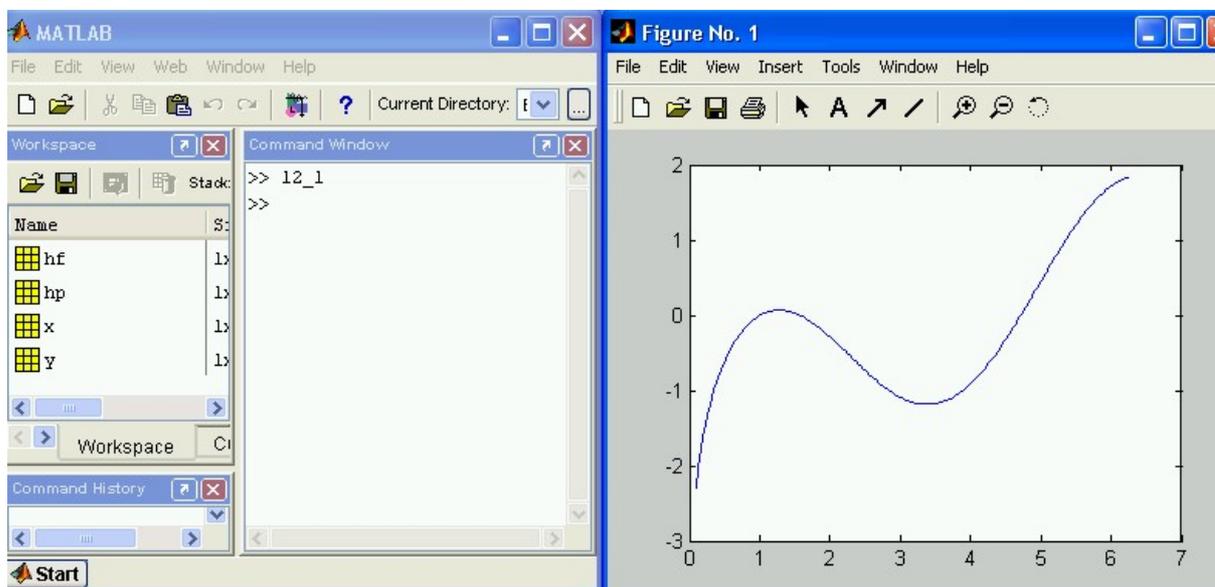
Графические объекты

Функция `figure` создает объект графическое окно. Например, команда `figure(1)` создает в памяти объект "графическое окно" и отображает объект на экране с подписью в верхней части окна "Figure No. 1". Если окно с таким

именем уже существовало, то оно получает фокус ввода и становится активным. Вообще говоря, вызов функции `figure` перед началом процесса рисования необязателен. Любые функции для рисования, такие как `plot` или `plot3`, автоматически открывают графическое окно перед началом рисования. Если окно уже было открыто, то рисование выполняется в последнем открытом окне. Выполним программу `l2_1.m`.

```
clear all           %чистим память
hf=figure(1);      %открываем графическое окно.
x=0.1:pi/100:2*pi; %заполняем массивы x и y
y=cos(x).*log(x);
hp=plot(x,y);      %рисуем график y(x)
```

На рис. 2.1 показана оболочка Matlab и результат работы программы `l2_1.m`.



Функция `uicontrol` для разработки интерфейсных программ

Функция `uicontrol` является основной функцией для разработки интерфейсных программ. Она размещает в памяти и высвечивает на указанном графическом окне интерфейсные элементы. Следующие две строки открывают графическое окно и размещают на нем интерфейсный элемент.

```
h1=figure;
hu=uicontrol(h1);
```

Тип интерфейсного элемента, его размеры, расположение в графическом окне, цвет и т.д. определяются значениями полей интерфейсного объекта. Посмотреть все возможные поля объекта и их значения по умолчанию можно командой `set(hu)`. В следующей таблице приводятся некоторые поля интерфейсных объектов и их возможные значения. Значения по умолчанию выделены в фигурные скобки.

Поле	Назначение
BackgroundColor	Определяет цвет объекта.
FontUnits	Единицы измерения размера шрифта. Возможные значения [inches centimeters normalized {points} pixels]
FontName	Имя шрифта.
FontSize	Размер шрифта.
Units	Единицы измерения для положения элемента в графическом окне и его размеров. Начало системы координат всегда в левом нижнем углу. Возможные значения [inches centimeters normalized points {pixels} characters]. Обычно для поля Units используют значение normalized. При таком выборе в левом нижнем углу графического окна всегда расположена точка (0,0), а в правом верхнем (1,1). Это позволяет создавать интерфейсы, независимые от разрешения монитора и фактического размера графического окна на дисплее.
Position	Определяет положение левого нижнего угла элемента и его ширину и высоту. Задается вектором из четырех элементов. Например, строка set(hu,'position',[0.1 0.1 0.4 0.5]); задает левый нижний угол в точке (0.1,0.1) , ширину 0.4 и высоту 0.5.
Style	Определяет тип интерфейсного элемента. Возможные значения [{pushbutton} togglebutton radiobutton checkbox edit text slider frame listbox popupmenu].
String	Задаёт надпись на интерфейсном элементе.
UiContextmenu	Ссылка на всплывающее меню, связанное с объектом.
Callback	Определяет имя функции, которая вызывается , когда с объектом происходит событие (например, щелчок мыши на объекте).

Например, чтобы создать интерфейсный объект типа togglebutton (кнопка с двумя положениями - утопленная и отжатая) и написать функцию для обработки нажатий нужно написать в матлабе

```
global ht
```

```
ht=uicontrol(h1,'style','togglebutton','callback','togl');
```

Для обработки нажатий на кнопку нужно написать функцию togl, которая может выглядеть примерно так

```
function togl()
```

```
global ht
```

```
val=get(ht,'value');
```

```
if val==0 % кнопка отжата
```

```
.
```

```
.
```

```
.
```

```
else %кнопка утоплена
```

```
.
```

```
.  
.end
```

Когда происходит нажатие на кнопку `tog1`, вызывается функция `tog1`. Эта функция получает указатель на объект `ht` через оператор `global`. Функция `get` считывает значение поля `'value'` объекта `ht`. Это поле определяет состояние объекта `togglebutton`, 0 - кнопка отжата, 1 - кнопка утоплена.

Следующий пример размещает интерфейсный объект типа `'popup'` и иписывает функцию `mycall` для обработки событий.

```
global hpop  
hpop=icontrol('style','popup','units','normalized','position',[0.1 0.1 0.4 0.05],...  
'string','hsv|hot|cool|jet','callback',mycall);
```

Функция `mycall` выглядит примерно так:

```
function mycall()  
global hpop  
val=get(hpop,'value');  
switch val % выбираем высвеченную строку  
case 1 % высвечена первая строка  
. . .  
case 2 % высвечена вторая строка  
. . .  
otherwise  
. . .  
end
```

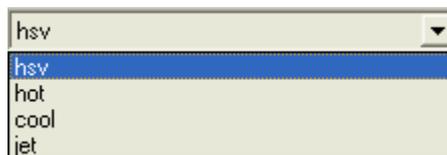


Рис. 3.1

В качестве полного рассмотрим программу для отрисовки графика заданной функции. Все необходимые интерфейсные элементы отрисовываются функцией `graphic.m`.

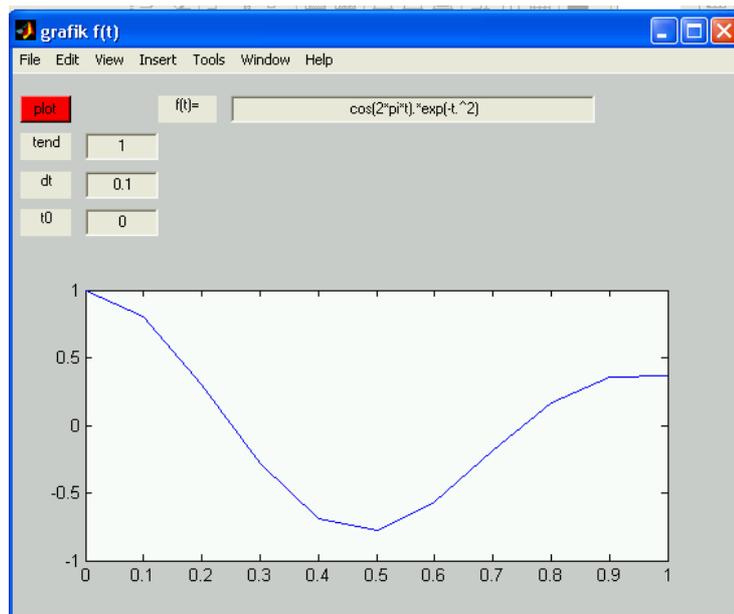
```
function graphic()  
%#function myplot  
global x0 dx xend fun  
h1=figure;
```

```

a1=axes('units','normalized','position',[0.1 0.1 0.8 0.5]);
set(h1,'currentaxes',a1,'numbertitle','off','name','grafik f(t)');
xlabel('t');
ylabel('f');
pos=[0.1 0.7 0.1 0.05];
x0=uicontrol('style','edit','units','normalized','position',pos,'string','0');
pos1=[0.1 0.77 0.1 0.05];
dx=uicontrol('style','edit','units','normalized','position',pos1,'string','0.1');
pos2=[0.1 0.84 0.1 0.05];
xend=uicontrol('style','edit','units','normalized','position',pos2,'string','1');
pos3=[0.01 0.7 0.07 0.05];
uicontrol('style','text','units','normalized','position',pos3,'string','t0');
pos4=[0.01 0.77 0.07 0.05];
uicontrol('style','text','units','normalized','position',pos4,'string','dt');
pos5=[0.01 0.84 0.07 0.05];
uicontrol('style','text','units','normalized','position',pos5,'string','tend');
pos6=[0.01 0.91 0.07 0.05];
uicontrol('style','pushbutton','units','normalized','position',pos6,...
'string','plot','backgroundcolor',[1 0 0],'callback','myplot');
pos7=[0.3 0.91 0.5 0.05];
fun=uicontrol('style','edit','units','normalized','position',pos7,'string',...
'cos(2*pi*t).*exp(-t.^2)');
pos8=[0.2 0.91 0.08 0.05];
uicontrol('style','text','units','normalized','position',pos8,'string','f(t)=');

```

Вот как выглядит результат работы этой функции на экране.



При нажатии кнопки `plot` вызывается функция `myplot` из файла `myplot.m`, которая и отрисовывает график.

```

function myplot()
global x0 dx xend fun
s1=get(x0,'string');
s2=get(dx,'string');
s3=get(xend,'string');
s4=get(fun,'string');
sx=strcat(['',s1,':',s2,':',s3,']);
t=eval(sx); % выполняем строку sx, как команду матлаба и результат заносим
в t.
%next string for compilation to the standalone application only
%you can to comment these string for work in matlab only
str=strrep(s4,'t',sx); % замена в строке s4 символа t на строку sx
f=eval(str); % строка str выполняется как команда матлаба, а результат
заносится в f
plot(t,f);

```

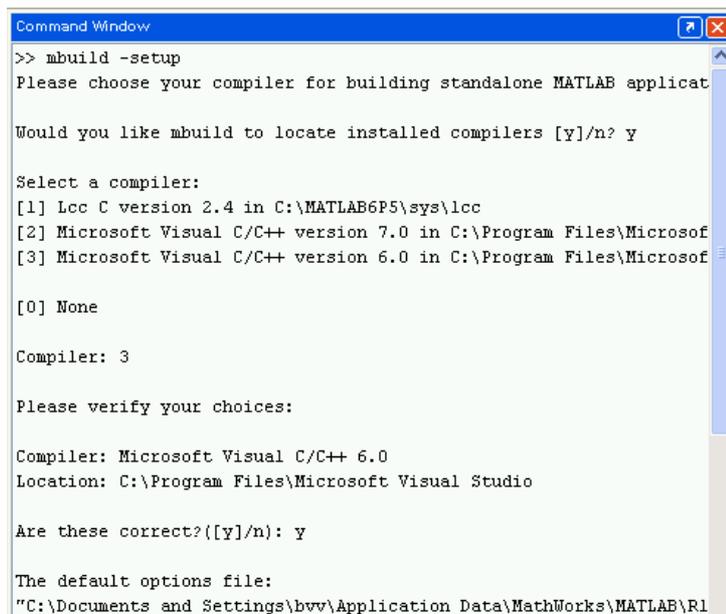
Замена символа `t` в строке `s4` на строку `sx` связана с тем, что в дальнейшем функция `myplot` будет компилироваться в Си код. В матлаб запрещено компилировать в Си функции содержащие вызов функции `eval` со строкой, содержащей имена переменных из рабочего пространства матлаба.

Как из функции на языке `matlab` приготовить код на языке Си и из него приготовить `exe`-файл.

Как производится компиляция программ, написанных на языке `matlab`, в код на языке Си и затем готовится независимое `exe`-приложение можно рассмотреть на примере функции `graphic.m`. Прежде всего необходимо заметить, что компилируются только функции матлаба.

Компиляция осуществляется с помощью компилятора `mcc`. Перед компиляцией необходимо настроить компилятор с помощью команды `mbuild-setup`, которую можно вводить либо из командного окна `matlab`, либо из командной строки Windows.

Вот как выглядит диалог в командном окне.



```
Command Window
>> mbuild -setup
Please choose your compiler for building standalone MATLAB applicat

Would you like mbuild to locate installed compilers [y]/n? y

Select a compiler:
[1] Lcc C version 2.4 in C:\MATLAB6P5\sys\lcc
[2] Microsoft Visual C/C++ version 7.0 in C:\Program Files\Microsof
[3] Microsoft Visual C/C++ version 6.0 in C:\Program Files\Microsof
[0] None

Compiler: 3

Please verify your choices:

Compiler: Microsoft Visual C/C++ 6.0
Location: C:\Program Files\Microsoft Visual Studio

Are these correct?([y]/n): y

The default options file:
"C:\Documents and Settings\bvv\Application Data\MathWorks\MATLAB\RL
```

После настройки компилятора можно приступить к компиляции с помощью команды `mcc -m -B sgl graphic.m`

После компиляции в рабочем каталоге образуются файлы `graphic.c` и `graphic.exe`. Файл `graphic.exe` является независимым `exe`-файлом, который может выполняться независимо от матлаба. Однако, если не установлен `matlab`, то для выполнения `exe`-файлов необходимо установить библиотеки матлаба. Они хранятся в специальном самораспаковывающемся архиве `mglinstaller.exe`, который обычно хранится в `%MATLAB%\extern\lib\win32`.

Необходимо обратить внимание на вторую строку файла `graphic.m`.
`%#function myplot`

Все `callback` функции должны быть объявлены подобным образом, если функция будет компилироваться в Си.

Тема 5. Построение баз данных в Delphi

Представление о процессе разработки программного обеспечения для реляционных баз данных. Методика взаимодействия программ, созданных в среде Delphi с базами данных различного типа.

Средства построения приложений, рассчитанных на работу с электронными архивами (базами данных). Возможность выбора способа доступа к данным: стандартный доступ с помощью машины баз данных BDE (Borland Data base Engine), технология ADO ActiveX Data Objects), прямое управление сервером InterBase. Основные принципы работы с SQL-сервером InterBase.

Тема 6. Компьютерная графика. Графические редакторы

Понятие графической информации, способы ее представления и обработки в ЭВМ. Пакеты программ, ориентированные на работу с графической информацией, назначение и основные возможности.

Растровое представление изображения. Понятие растра. Виды растров. Алгоритмы растровой графики. Геометрические преобразования. Рисование 3D сцен. Текстуры. Моделирование освещения. Представление пространственных форм. Цвет в компьютерной графике. Обработка изображений.

Тема 7. САПР

Структура, принципы создания и классификация САПР. Состав программно-технических комплексов САПР. Обзор современных САПР.

Система автоматизированного проектирования (AutoCAD).

Системы инженерной графики. САПР. AutoCAD, назначение, характеристика основных функциональных возможностей. Построение основных графических примитивов. Способы привязки при построении графических образов. Возможности редактирования графических образов в среде AutoCAD. Работа с текстом в AutoCAD. Штриховка и заливка цветом графических объектов. Нанесение размеров и допусков.

Проектирование – процесс создания описания необходимого для построения в заданных условиях еще не существующего объекта на основе его первичного описания. Проектирование разделяют на неавтоматизированное, автоматизированное и автоматическое. Мы будем рассматривать автоматизированное, особенность которого состоит в том, что в ходе проектирования происходит постоянный диалог человека с ЭВМ в процессе создания описания объекта.

Выделяют две основные задачи в САПР:

1. Задача анализа. Связана с определением функции объекта или системы по заданному описанию и оценкой возможных проектных решений.
2. Задача синтеза. Связана с созданием самого объекта и проектной документацией.

Синтез бывает:

1. Структурный – получение структурной схемы объекта, формирование сведений о составе элементов и способах их соединения между собой.

2. Параметрический – определение числовых значений параметров элементов или систем. Синтез называют оптимизацией, если определяются наилучшие в заданном смысле структуры (структурная оптимизация) или параметры (параметрическая оптимизация). Проведение оптимизации требует задание спектра критериев оптимизации и соответствующих ограничений. Если критерий один, то скалярная оптимизация, если несколько – векторная.

Критерии бывают:

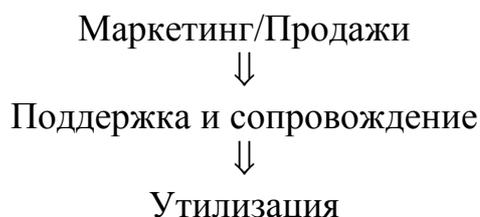
1. Частные.
2. Аддитивные.
3. Мультипликативные.

4. Минимаксные. Для проведения задач анализа применяют математические модели и языки моделирования (имитационное моделирование), позволяющие проводить анализ объекта в динамике (VHDL и VERILOG).

Моделирование - это работа по прогнозированию характеристик жизненного цикла продукта до его производства. Характеристики продукта включают затраты на его разработку, эксплуатацию, расходные материалы (топливо), продолжительность жизни продукта, ударопрочность, прочность, безопасность, шум, надежность, комфортность, простоту изготовления и обслуживания, затраты на гарантийный ремонт, время выведения на рынок, прибыльность и многое другое. Моделирование выполняется при помощи программного обеспечения, способного прогнозировать все эти характеристики жизненного цикла продукта. Моделирование выполняется для прогнозирования будущего, а особенности продукта систематически варьируются с целью улучшения характеристик продукта на раннем этапе разработки, при этом решения о технологическом и материальном оснащении принимаются заранее. Моделирование помогает проектировщикам понять, какие компромиссы и решения в области проекта они должны сделать для оптимизации своей продукции. Моделирование сокращает и заменяет дорогостоящий длительный процесс создания физических прототипов и их тестирования. Моделирование может воздействовать на каждый из пунктов, обозначенных выше, приводя Вас к увеличению продаж и сокращению затрат на разработку продукта.

Немного о жизненном цикле продукта.





Моделирование может использоваться для сокращения времени на каждом этапе процесса и для создания правильного проекта изделия - такого, который было бы легко воплотить в производство, легко продавать и обслуживать – в Самом начале процесса, до планирования основных затрат (оборудование и материалы).

О значении моделирования на этапе Концептуального Проектирования. На этапе разработки концепции, основное решение на высшем уровне принимается исходя из рыночных потребностей в данном изделии, требований функциональности продукта и нужд бизнеса. Моделирование на этом этапе позволяет разработчикам концепции проверить ее, чтобы убедиться, что продукт можно изготовить в соответствии с заданными требованиями и потребительскими свойствами. Моделирование облегчает первоначальную оценку различных концепций проекта, предоставляя возможность удовлетворения всех требований к функциональности в установленных рамках времени и затрат средств на выполнение работ по данному проекту.

О значении моделирования на этапе Детального Проектирования. На этом этапе проект в целом уже определен, включая его отдельные узлы и компоненты с заданными потребительскими свойствами. Определены геометрия, материалы и стадии производственного процесса. Моделирование на этом этапе позволяет удостовериться, что данный проект реален, его можно будет выполнить и поставить на рельсы серийного производства. Весь проект может быть смоделирован, от системы в целом до каждого отдельного компонента. На этом этапе моделирование выполняется инженерами-разработчиками и инженерами-технологами, планирующими производство.

О значении моделирования на этапе Испытаний. Это один из важных этапов в жизни проекта, т.к. он определяет дальнейшую судьбу изделия. Этап испытаний начинается тогда, когда доступен образец. Большинство компаний создают несколько натуральных прототипов и подвергают их тщательному тестированию. Если экспериментальные образцы не проходят испытаний, проект изменяется и образец проходит новые испытания, до тех пор, пока не будет получен положительный результат. Такой цикл создания, испытания и устранения дефектов требует значительных временных, производственных и финансовых затрат. Моделирование может использоваться на этой стадии для уменьшения числа прототипов и физических циклов восстановления и повторного испытания, путем моделирование тестируемого образца на компьютере. Если тестируемый образец не проходит модельных испытаний, повторное изменение проекта очень недорого. Используя ком-

пьютерное моделирование на данном этапе удастся сократить затраты на разработку прототипов более чем на 50%, а это значительная экономия времени и средств. Кроме этого, моделирование может быть использовано для сокращения времени и объема испытаний натурального образца. Например, при помощи моделирования можно «пробежаться» по образцу, сообщив «данные», которые бы измерялись при физическом испытании. Это позволяет инженерам, производящим испытание, заранее и более качественно определять критерии измерений и нагрузки/динамического нагружения, экономя время и исключая «работу вслепую» как часто бывает при лабораторных исследованиях.

О значении моделирования на этапе Производства. Всем понятно, что на этом этапе продукция изготавливается. В данном случае моделирование используется для оптимизации процессов производства с целью минимизации отходов и этапов обработки. Моделирование улучшает процедуру изготовления продукта и снижает затраты на гарантийный ремонт, выявляя и исключая те параметры проекта, которые могут вызвать дефекты во время процесса производства. Более того, некоторые из новых производственных процессов, такие как гидроформование и суперпластическое формование, требуют моделирования для идентификации параметров производственного процесса, таких как температура, давление и скорость.

О значении моделирования на этапе Маркетинга/Продаж. Моделирование может использоваться не только для разработки и производства Вашей продукции. Оно также может применяться для маркетинга и продаж Вашей продукции. Моделирование можно использовать в коммерческих предложениях, чтобы показать, как Ваш продукт будет выглядеть в конкретных условиях и какими интересными свойствами он может быть интересен покупателю. Результаты моделирования могут использоваться в рекламе для придания Вашему продукту образа «высокотехнологичного». Ваши продажи увеличатся благодаря тому, что моделирование улучшает дизайн, а также саму продукцию, т.е. ее потребительские свойства.

О значении моделирования на этапе Поддержки и Сопровождения. В дальнейшем, после реализации, Ваша продукция будет нуждаться в поддержке, техническом обслуживании, капитальном и текущем ремонте. Моделирование может использоваться на этом этапе для проведения ремонта и для модификаций изделия, обеспечивая корректировку проблемы, при этом одновременно сохраняя функциональность первоначального дизайна и набор полезных свойств. Задачи этого этапа очень четко просматриваются в аэрокосмической промышленности, где моделирование применяется для продления ресурса жизненного цикла изделий.

О значении моделирования на этапе Утилизации. Это последний и тоже достаточно важный этап жизненного цикла изделий и технологий, особенно актуальный в настоящее время. Ничто не вечно в этом мире. Когда срок полезного использования изделия закончен, оно утилизируется или

перерабатывается. Моделирование здесь применяется для выбора таких производственных процессов и упаковочных материалов, при которых возможна экономичная переработка изделия, включая различные типы материалов, которые были использованы для его изготовления. Во многих отраслях промышленности число «контактирующих частей» во время производства на порядок превышает число фактических частей в продукте. Применение моделирования позволяет спланировать их повторное использование или эффективную утилизацию еще на этапе эскизного проектирования самого изделия, значительно задолго до начала его эксплуатации.

А где же возврат инвестиций, потраченных на моделирование? Когда Вы покупаете программное обеспечение для моделирования, включая гарантийное обслуживание и компьютеры, Ваша компания делает инвестиции. Инвестиции - это затраты, которые, по идее, должны возвращаться. Таким образом, инвестиция дает доход (финансовую прибыль), который по размеру должен обычно превосходить саму инвестицию. Поэтому в первую очередь Вас будет интересовать «доходность инвестиций», существует и такой термин. Давайте теперь обратимся к рассмотрению и анализу доходности инвестиций в моделирующие компьютерные технологии. Существует несколько аспектов «доходности инвестиций» при вложении средств в моделирующие технологии:

- Вид инвестиции (разовая или постоянная);
- Валовая прибыль;
- Доход от инвестиции (финансовая прибыль);
- Внутренняя норма прибыли.

Срок окупаемости инвестиций.

Математическое моделирование объектов применяется, как правило, для расчета и выбора оптимальных параметров объекта или системы, при условии, что построение математической модели возможно, и для условий статистики. Если построение математической модели невозможно, то применяют статистические методы и методы нейро-сетевых подходов. Также для проведения задачи анализа в настоящее время получили широкое распространение СППР (Системы Поддержки Принятия Решений), СПР и экспертные системы, которые строятся на принципах искусственного интеллекта, в основе лежит продукционный подход, семантические сети и фреймы, и языки искусственного интеллекта (Prolog и Lisp). Фреймы – совокупности взаимосвязанных данных, позволяющих точно определить характер объекта. СППР и экспертные системы могут включать в себя одновременно весь спектр вышеуказанных моделей и средств.

Существует множество определений СППР, отражающих точки зрения представителей различных дисциплин и научных школ. Так СППР может определяться, как "основанная на использовании моделей совокупность процедур по обработке данных и суждений, помогающих руководителю в принятии решений".

В некоторых случаях предлагается рассматривать СППР в качестве "интерактивных автоматизированных систем, которые помогают лицам, принимающим решения, использовать данные и модели, чтобы решать неструктурированные проблемы". СППР может выглядеть как "компьютерная информационная система, используемая для поддержки различных видов деятельности при принятии решений в ситуациях, где невозможно или нежелательно иметь автоматическую систему, которая полностью выполняет весь процесс решения". Большинство исследователей согласны, что СППР предназначены для решения слабоструктурированных проблем.

В процессе принятия решений возникла необходимость в субъективных, экспертных моделях, которые могут быть крайне полезны для ЛПР. Возникла также необходимость в учете знаний многих экспертов, в анализе принятых ранее решений. В структуре СППР появился блок "база знаний" (БЗ), и такие системы получили название "интеллектуальных".

Развитие технических программных средств, позволяющих "индустриализировать" технологию создания новых систем, привело к формированию еще одной точки зрения на СППР, которая получила название "адаптивного проектирования". Сторонники этого подхода считают, что термин СППР имеет право на существование только в тех случаях, когда "конечная система" возникает в ходе адаптивного процесса проектирования и внедрения.

Процесс принятия решений.

ТПР, а следовательно, и СППР подразумевает под собой использование процесса, представленного на рис.1.

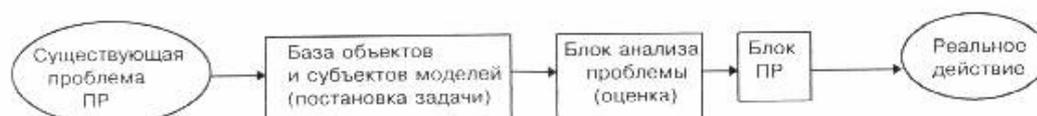


Рис.1. Процесс принятия решения.

Мы начинаем с некоей существующей проблемы решения, с которой столкнулся человек, которому необходимо принять это решение. Наше внимание направлено на то, чтобы применить последовательность видимых шагов для внесения ясности, применительно к ситуации, когда человек, которому необходимо принять решение, смог предпринять рекомендованные действия.

Первый шаг (постановка задачи) – формальная модель той реальной ситуации, с которой столкнется ЛПР. Мы назовем это формальное представление проблемы "базой объективных и субъективных моделей". В основе решения лежит процесс, направленный на выделение альтернативных подходов к решению проблемы, которые логически согласуются с базой знаний (БЗ) и отсюда рекомендуются для дальнейшего анализа. Затем нам предстоит усовершенствовать оценку анализов с тем, чтобы достичь интуиции, почему предложенный альтернативный подход не только является

логически правильным, но и выглядит убедительным, что человек будет действовать в полном соответствии с ним. Оценка может содержать некоторые варианты анализов с тем, что оценка действительно соответствует содержанию проблемы. В какой-то момент, оценка покажет, что предложенный альтернативный подход настолько верен, что для ЛПР нет смысла продолжать анализ.

Целью процесса является синтез основы решения. Рисунок 2 более детально демонстрирует роль "базы объективных и субъективных моделей". Основа включает в себя три части: альтернативный подход, который выбрал ЛПР; доступная информация и предпочтения, отдаваемые ЛПР.



Рис.2. Выявление и оценка БД.

Концептуальные модели.

Анализ существующих точек зрения на специфические отличия СППР от других типов автоматизированных систем в системах проектирования позволяет выделить в качестве оснований классификации СППР следующие наиболее существенные признаки: концептуальные модели; решаемые задачи; обеспечивающие средства; области применения.

В рамках информационного подхода СППР в промышленных САПР и САИТ относят к классу автоматизированных информационных систем, основное назначение которых – улучшить деятельность работников умственного труда в организациях путем применения информационной технологии.

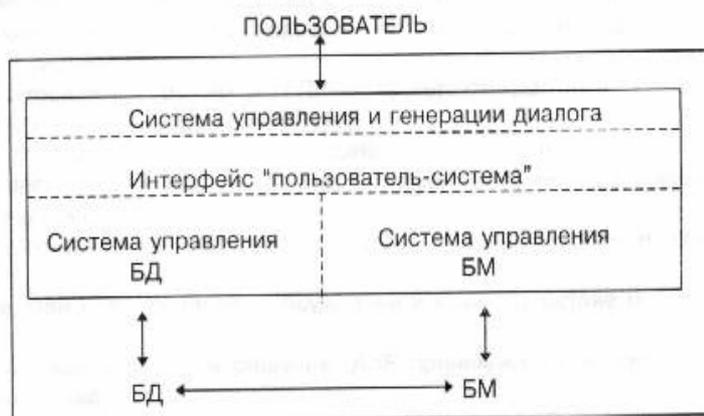


Рис.3. Концептуальная модель СППР (информационный подход).

Важной особенностью СППР в промышленных САПР и САИТ является их способность формировать модели для принятия решений.

Предполагается, что в БМ следует встраивать не локальные модели, а модели объединенные с БД. Процедуры моделирования должны обеспечивать гибкость построения моделей и легкость управления ими, а система управления – возможность обслуживать широкий спектр моделей, быстро и легко создавать новые модели, управлять БД с помощью функций управления.

Отличительной особенностью СППР в промышленных САПР и САИТ, которые основаны на знаниях, является, по мнению их создателей, явное выделение отсутствовавшего ранее аспекта поддержки решений: способности к "пониманию" проблемы, извлечь информацию и подготовить ответ. Степень участия программных средств человеко-машинной системы в этом процессе предлагается рассматривать в качестве грубой меры ИИ СППР.

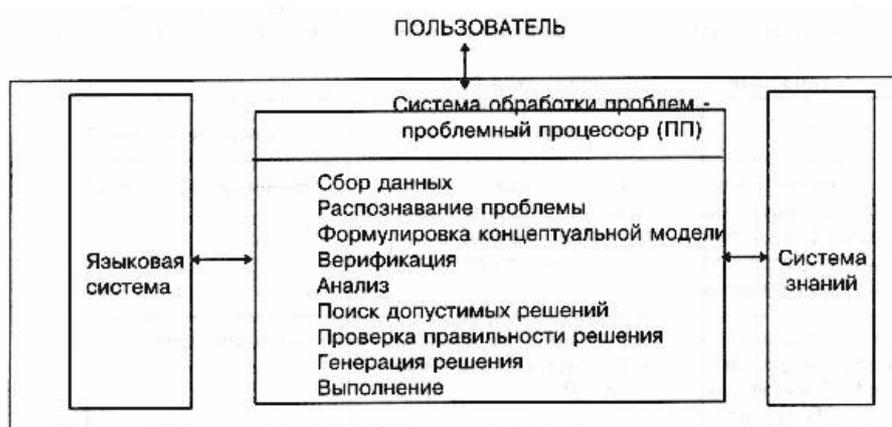


Рис.4. КП СППР (подход, основанный на знаниях).

Возросшее внимание к методам разработки и внедрения СППР в промышленных САПР и САИТ обусловило появление инструментального подхода в концептуальных моделях СППР.

Можно выделить три условия систем:

- Специализированные САПР (работают конечные пользователи; поддержка решения отдельных прикладных программ (задач) в конкретных ситуациях);
- СППР-генераторы (пакеты связанных программных средств поиска и выдачи данных);
- СППР-"инструментарий" (высший уровень технологичности – языки, совершенные ОС, средства ВВ и прочее).

Виды обеспечения САПР

1. Математическое обеспечение (МО) – совокупность математических методов, моделей и алгоритмов проектирования, представленных в заданной форме.
2. Техническое обеспечение (ТО) – совокупность связанных и взаимодействующих технических средств (средств вычислительной техники).

3. Программное обеспечение (ПО) – совокупность машинных программ, необходимых для осуществления процесса проектирования, включающее системное и прикладное ПО.
4. Информационное обеспечение (ИО) – совокупность сведений, необходимых для выполнения проектирования. Включает СУБД (Система Управления Базами Данных), саму базу данных и базу знаний.
5. Лингвистическое обеспечение (ЛО) – совокупность языков проектирования, включая термины, определения, правила формализации естественного языка, методы сжатия и развертывания текстов.
6. Методическое обеспечение (МО) – совокупность документов, устанавливающих состав, правила отбора и эксплуатации средств обеспечения системы.
7. Организационное обеспечение (ОО) – совокупность документов, определяющих состав проектной организации, связь между подразделениями, а также форму представления результатов проектирования и порядок рассмотрения проектных документов.

Классификация САПР

САПР классифицируют по:

- *разновидности и сложности объектов проектирования:*

- а) САПР низкосложных объектов (количество составных частей до 100);
- б) САПР среднесложных объектов (100-10 000);
- в) САПР высокосложных объектов (выше 10 000).

- *уровню автоматизации:*

- а) низкоавтоматизированные (до 25% проектных процедур автоматизировано);
- б) среднеавтоматизированные (25%-50%);
- в) высокоавтоматизированные (50%-75%).

- *уровню комплексности:*

- а) одноэтапные (один этап проектирования);
- б) многоэтапные (несколько этапов);
- в) комплексные (весь процесс создания изделия).

- *характеру и числу выпускаемых проектом документов:*

- а) САПР низкой производительности (100-10 000 проектных документов в пересчете на формат А4 за год);
- б) САПР средней производительности (10 000- 100 000);
- в) САПР высокой производительности (100 000 и выше).

По ГОСТу информация должна храниться как в твердом виде (на бумаге) так и на магнитном носителе (магнитная лента); срок хранения – 50 лет.

- *числу уровней технического обеспечения:*

- а) одноуровневый (на основе ЭВМ среднего и высокого класса со штатным периферийным оборудованием);
- б) двухуровневый (на основе ЭВМ среднего и высокого класса, в качестве интеллектуальных терминалов – персональные ЭВМ);

в) комплексный (на основе ЭВМ среднего и высокого класса, объединенных в сеть и каждая из ЭВМ имеет сеть интеллектуальных терминалов на основе персональных ЭВМ).

ОРГАНИЗАЦИЯ САПР

Составными структурными частями САПР являются подсистемы, обладающие всеми свойствами системы и создаваемые как самостоятельные системы.

Подсистемой САПР называют выделенную по некоторым признакам часть САПР, позволяющую получать законченные проектные решения.

САПР разделяют на:

Проектирующие подсистемы

Обслуживающие подсистемы

К проектирующим подсистемам относят:

1. Подсистему функционально-логического проектирования.

На выходе этой системы мы получаем функциональную схему, за ней логическую схему, на выходе принципиально - электрическую схему.

2. Подсистему конструкторского проектирования

На выходе получаем конструкцию устройства и конструкторскую документацию, включающую схему расположения элементов на поверхности модуля и топологию печатных соединений между элементами.

3. Подсистему технологической подготовки производства

На выходе получаем маршрутную карту производственного процесса и программы для управления станков с числовым программным управлением (для управления технологическим оборудованием).

К обслуживающим подсистемам относят:

Систему информационного поиска

Систему документирования

Систему графического отображения объектов проектирования

В состав как проектирующих, так и обслуживающих систем современных САПР могут входить:

Экспертные системы

Это системы, в основе которых лежит база знаний, представленная либо в виде системы продукции, либо в виде фреймов (FRAME). Экспертная система позволяет формализовать знания эксперта в определенной предметной области с целью принятия рациональных проектных решений.

Системы принятия решений

Это системы, позволяющие производить выбор эффективных проектных решений в условиях определенности и неопределенности исходной информации на основе формальных методов и процедур. Для оценки проектных решений могут также применяться нейросетевые технологии.

Системы поддержки принятия решений.

ПРИНЦИПЫ СОЗДАНИЯ САПР

Принцип включения. Обеспечивает разработку систем на основе требований, позволяющих включать эти системы в САПР более высокого уровня.

Принцип системного единства. При создании, функционировании и развитии САПР связь между подсистемами должна обеспечивать целостность всей системы.

Принцип развития. САПР должна создаваться и функционировать с учетом появления, совершенствования и обновления ее подсистем и компонентов.

Принцип комплексности. Обеспечивает связность процесса проектирования элементов и объектов в целом на всех уровнях проектирования, позволяя осуществлять согласование и контроль характеристик элементов и объектов в целом.

Принцип информационного единства. Состоит в использовании в подсистемах, компонентах и средствах обеспечения САПР единых условных обозначений, терминов, символов, проблемно-ориентированных языков и способов представления данных в соответствии с принятыми нормативными документами.

Принцип совместимости. Языки, символы, коды, информационные и технические характеристики, связи между подсистемами, средствами обеспечения САПР и компонентами должны обеспечивать эффективное функционирование подсистем и сохранять открытую структуру системы в целом.

Открытой называют систему, в которой интерфейсы взаимодействия с внешней средой стандартизированы.

Принцип стандартизации. Состоит в проведении унификации, типизации, стандартизации подсистем и компонентов, инвариантных к проектным объектам и отраслевой специфике, а также установление правил с целью упорядочивания деятельности по созданию и развитию САПР.

СТАДИИ СОЗДАНИЯ САПР

При создании САПР различают внешнее и внутреннее проектирование.

К внешнему проектированию относят:

Стадию предпроектных исследований

Разработку технического задания на проект

Внутреннее проектирование включает:

разработку технических предложений

рабочий проект

эскизный проект

технический проект

изготовление, отладку, испытание и ввод системы в действие

Выше сказанное по существу может быть применено при создании большинства технических объектов и систем.

СИСТЕМНЫЙ ПОДХОД К ПРОЕКТИРОВАНИЮ

Исследование объектов проектирования с помощью их математических моделей составляет суть системного подхода. Выделяют следующие принципы системного подхода:

Иерархичность. Каждая система или элемент может рассматриваться как отдельная система.

Структурность. Состоит в возможности описания системы через описание коммутационных связей между ее элементами.

Взаимозависимость. Заключается в проявлении свойств системы только при взаимодействии с внешней средой.

Множественность описания. Заключается в описании системы на основе множества взаимодействующих математических моделей.

Целостность. Свойства всей системы определяются на основе анализа свойств ее частей.

Суть системного подхода – это проектирование части с учетом целого.

БАЗОВЫЕ ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ В САПР

Наиболее перспективными на сегодняшний день являются технологии:

- Сквозного проектирования
- Параллельного проектирования
- Нисходящего проектирования
- CALLS технологии

Сквозное проектирование.

Смысл сквозной технологии состоит в эффективной передаче данных и результатов конкретного текущего этапа проектирования сразу на все последующие этапы. Это нужно для того, чтобы на необходимом уровне качества выполнить какой-либо из этапов проектирования. Разработчику часто не хватает регламентированной информации от предыдущего этапа и необходима более полная и разнообразная информация, которая могла быть сформулирована на одном из ранних этапов проектирования (не обязательно на соседнем). У разработчиков, выполняющих различные этапы проектирования, может быть одновременно с первым этапом проектирования получено техническое задание и таким образом, все разработчики могут одновременно начать продумывать как более успешно реализовать свой этап. Данная технология базируется на модульном построении САПР, на использовании общих баз данных и баз знаний, и характеризуется широкими возможностями моделирования и контроля на всех этапах проектирования. Сквозные САПР как правило являются интегрированными, т.е. имеют альтернативные алгоритмы реализации отдельных проектных процедур.

Параллельное проектирование.

При параллельном проектировании информации относительно каких-либо промежуточных или окончательных характеристик разрабатываемого

изделия формируется и предоставляется всем участникам работ, начиная с самых ранних этапов проектирования. В этом случае информация носит прогностический характер. Ее получение базируется на математических моделях и методах прогностической оценки критериев качества проектного решения. Оценка может производиться на основе аналитической модели, статистических методов и методах экспортных систем. Технология параллельного проектирования реализуется на основе интегрированных инструментальных средств прогностической оценки и анализа альтернативных проектных решений с последующим выбором базового проектного решения. Предполагается, что инженер начинает работать над проектом на высоком уровне абстракции с последующей детализацией проекта.

Принципиальным отличием параллельного проектирования от сквозного проектирования (хотя параллельное проектирование получило развитие на основе сквозного) в том, что информация не просто поступает на все последующие этапы проектирования, но и по существу эти этапы начинают выполняться одновременно.

Фирма MENTOR GRAPHICS впервые создала среду параллельного проектирования на основе принципа объединения всех инструментальных средств проектирования и данных в одном непрерывном и гибком процессе создания изделия.

В состав этой инфраструктуры входят:

Среда управления проектированием

Система управления данными проекта

Система поддержки принятия решений (СППР)

Нисходящее проектирование.

Предполагается, что инженер начинает работать над проектом на высоком уровне абстракции с последующей детализацией проекта. Основной задачей руководителя или инженера является определение оптимального концептуального решения, выбор функциональных алгоритмов проектирования, а также выбор наиболее эффективных инструментальных средств проектирования.

Другими словами определение правильной стратегии проектирования на основе достаточно общей и зачастую неопределенной информации. Данная программа решается на основе применения предиктивных инструментальных средств, т.е. программ, обеспечивающих связь этапов функционально-логического, конструкторского проектирования и этапа технологической подготовки производства. При этом предиктивный инструментальный используется как на уровне отдельных проектных процедур, так и на уровне проекта в целом. Нисходящее проектирование позволяет получить изделия с другими характеристиками, создать надежное устройство. Все современные производители работают на уровне нисходящего проектирования.

CALLS технология.

Основная идея заключается в создании электронного описания и сопровождения изделия на всех этапах его жизненного цикла. Электронное описание должно соответствовать принятым отечественным и международным стандартам в данной предметной области. Это технология информационного сопровождения создания изделия.

СТРУКТУРА ПРОЦЕССОВ ПРОЕКТИРОВАНИЯ МОДУЛЯ ЭЛЕКТРОННОЙ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

1. Концептуальное проектирование (АВАН) – основополагающий и определяющий этап проектирования и все современные фирмы-разработчики систем проектирования перемещают проектные ресурсы на ранние этапы проектирования. Доказано, что при этом достигается почти двойная экономия средств за счет того, что отсутствует необходимость в повторных операциях последующих этапов проектирования. На этом этапе возможна доработка технического задания с целью обеспечения заданного уровня качества изделия, а иногда просто для обеспечения реализуемости проекта как такового.

2. Функционально- логическое проектирование.

А) Проектирование функциональных схем.

Б) Проектирование программ испытаний и тестов.

3. Конструктивное (техническое) проектирование.

А) Конструктивное АВАН проектирование.

а) Формирование множества рациональных вариантов (конструктивные решения на основе методов принятия решений).

б) Анализ альтернативных программных модулей реализации последующих проектных процедур и выбор из них наиболее приемлемых - адаптация САПР к объекту проектирования.

в) Выбор базового варианта конструктивного решения (выбор метрических и топологических параметров объекта).

Б) Компоновка конструктивных модулей .

В) Этап размещения элементов на поверхности модуля.

Г) Трассировка сигнальных соединений.

Д) Технологическая подготовка производства (это создание маршрутных карт производственного процесса).

Е) Подготовка технической документации.

СТРАТЕГИЯ АВТОМАТИЗИРОВАННОГО ИНТЕГРИРОВАННОГО ПРОИЗВОДСТВА (АИП)

Целью АИП является применение эффективных методов проектирования производства в единой системе с целью обеспечения экономического благополучия предприятия. Технология АИП базируется на полном электронном описании изделий.

Стратегия АИП включает:

Маркетинг

Концептуальное и рабочее проектирование
Технологическую подготовку производства
Эксплуатацию объекта
Его сопровождение

Примером современной системы, реализующей стратегию АИП, является система CADD'S 5. Интегрально-инструментальная среда системы включает 85 отдельных продуктов функционально охватывающих:

Эскизное и рабочее проектирование
Синтез геометрических моделей
Инженерный анализ
Разработку документации
Подготовку производства

Работая в среде параметрического конструирования системы CADD'S 5, пользователь определяет целевую функцию, указывает изменяемые параметры и задает связывающие условия. Далее осуществляется настройка системы и запускается подсистема «Design optimizer», которая последовательно просматривает различные варианты конструкции, приближаясь к оптимальному решению.

Выделяют шесть основных функций АИП:

Исследование, разработка и проектирование

- а) Базисное проектирование (составная часть – этап АВАН проектирования и предпроектные исследования);
- б) Прогностическую оценку и оптимизацию проектных параметров;
- в) Контроль и администрирование процесса проектирования в терминах информационных систем.

Эта функция управляет данными:

- данные геометрического моделирования;
- цифровой контроль данных;
- анализ корректности данных;
- инспекция технического задания;
- администрирование данных в соответствии с наиболее эффективными маршрутами проектирования).

Эта функция реализуется через систему автоматизации инженерного труда (САИТ), а также непосредственно через модули САПР.

Наиболее успешно эта функция реализована в системе PROENGINEER, которая является CAD/CAM/CAE системы третьего поколения. Система обеспечивает работу проектировщика, с единой математической моделью объекта, а не с набором разрозненных математических моделей.

Обобщенная математическая модель объекта может быть представлена в виде многоуровневой, конвейерной, сетевой математической моделью, где уровни модели отражают уровни проектирования (этапы) изделия, конвейерность модели отражает механизмы передачи данных и их спектр между этапами (не обязательно соседними), и сетевая структура модели

отражает параметрические связи между отдельными характеристиками проектируемого изделия. Многоуровневость модели отражает также процесс последовательной детализации объекта в ходе проектирования.

Ядром всех современных технологий проектирования (нисходящее, сквозное, параллельное, CALS) является многоуровневая математическая модель проектирования (конвейерная, сетевая, обобщенная).

Планирование производства

Включает в себя все уровни планирования, в том числе:

- а) Долгосрочное планирование
- б) Динамическое планирование, проектирование и производство
- в) Планирование проектирования
- г) Планирование производства
- д) Планирование сбыта изделия

Организация производственных процедур

Включает процедуры по производству, контролю и хранению готовой продукции

Продажа и маркетинг

Включает в себя:

- а) Установление планов по производству и продаже
- б) Разработку программ по маркетингу
- в) Обеспечение работы вспомогательных служб

Финансовое управление процессами проектирования и производства, распределение и учет затрат

Обеспечивает постоянную опережающую оценку затрат при процессах проектирования и производства

Система администрирования

Включает в себя все аспекты управления

- а) Персоналом
- б) Офисом (управление предприятием, собственностью, отношениями с общественностью)

Все вышеперечисленные функции строятся на принципе опережающего планирования процессов проектирования и производства изделия.

Управление общим потоком информации в компании, реализующей технологию АИП, невозможно без элементов поддержки, к которым относят:

- Аппаратное обеспечение и программный контроль
- Обеспечение информационной связи компьютеров
- Управление данными
- Общее управление системой
- Прикладное программное обеспечение
- Систему поддержки принятия решений
- Систему управления производственными функциями

Структуру САПР можно представить в виде следующих модулей:

- Модуль структурного проектирования – разработка структуры объекта.

- Модуль схемотехнического проектирования – выбор и расчет элементов, расчет параметров.

- Модуль конструкторского проектирования – разработка конструкции изделия, топология.

Если речь идет о разработке изделий электроники, то представленные модули имеют следующие названия:

- Подсистема концептуального проектирования.

- Подсистема функционально-логического проектирования.

- Подсистема конструкторского проектирования.

- Потом идет этап технологической подготовки производства, само производство и этап испытаний.

Если речь идет о программном обеспечении – подсистемы алгоритмов, программ, автономное комплексное тестирование внешних спецификаций.

Для выполнения указанной последовательности этапов необходимы следующие подсистемные составные части:

- Модуль групповой обработки данных – производит прогностические оценки и совместный анализ параметров различных этапов разработки изделия.

- Модуль генерации проектных решений – встроен во все подсистемы.

- Модуль тестирования проектных решений.

- Модуль согласования решений и принятия групповых решений.

- Модуль ответственности за построение статических, аналитических и динамических моделей.

- Модуль графики.

Групповая обработка данных

Строится на основе распределенных систем поддержки принятия решений. Основное назначение – преодоление трудностей связанных с последовательной технологией проектирования и принятия решений и переходом к параллельной технологии проектирования. Технология, базирующаяся на вычислительных сетях, обеспечивает каждого участника группы возможностью использования ЭВМ для участия в выработке решения, хотя в некоторых случаях может применяться один общий большой экран. Большинство существующих систем поддержки принятия решений базируются на двух технологиях – телекоммуникации и базы данных. Сейчас все больше появляются мультимедиа-технологии, которые позволяют использовать методы виртуальной реальности.

Генерация проектных решений

Можно подразделить на:

- Генерация решений, основанная на типовых сценариях, по аналогии или на основе комбинаций известных частных решений (выполняется ЭВМ).

- Генерация неожиданных решений – компьютер проводит оценку предлагаемых решений.
- Модуль может быть построен на основе программной реализации аналитических моделей; с использованием экспертных систем; путем комбинации различных процедур, находящихся в базе данных.

Оценка проектов

Производится на основе либо аналитических моделей, либо динамических, либо экспертных систем (содержат лингвистический процессор для общения с пользователем, базу знаний, содержащую эвристические знания и интерпретатор, который на основе входных данных формирует решение задачи). Знания в экспертных системах могут быть представлены в виде семантических сетей, на основе логических подходов, на основе фреймов и продукционных правил. При разработке сложного технического объекта на первый план выступает задача комплексного проектирования, и система поддержки принятия решений берет на себя функции по стыковке между собой различных устройств и подсистем объекта. С учетом технологии параллельного проектирования, точками синхронизации процесса параллельной работы являются модули согласования проектных решений по проектам стыкуемых объектов. Если речь идет о разработке изделий электроники, то генерация возможных решений может основываться на переборе различных вариантов компоновки, заложенных в экспертную систему, варьированием значений коэффициентов уравнений и на основе комбинирования различных элементов конструкций. А моделирующие системы строятся на основе численных методов решения систем уравнений, на статистических методах обработки предыдущих проектов и на методах ситуационного моделирования.

Модуль графики

Визуализация требует достаточно серьезных графических приложений и соответственно резко увеличивает установочные требования к системе, поэтому на сегодняшний день они получили распространение в системах разработки изделий механики, в машиностроении и строительстве.

Тема 8. Геоинформационные системы, их назначение создание и использование. Программное обеспечение ГИС

1. Место геоинформатики в системе наук. Взаимосвязи с картографией, информатикой и математическими методами, применяемыми в различных сферах геоэкологии, картографии, дистанционного зондирования. Основные термины геоинформатики.

Геоинформационные системы. Классификации ГИС. Структура ГИС. История развития ГИС.

2. Общие принципы построения моделей данных в ГИС. Основные понятия моделей данных. Базовые модели данных в ГИС. Данные, их типы. Источники данных. Технические средства ввода и вывода данных. Технологии ввода данных (растровых и векторных).

Структурирование пространственных данных. Растровое представление данных и его разновидности. Алгоритмы сжатия растровых данных. Векторное представление данных и его разновидности (бесструктурные, топологические, решетчатые модели). Преобразования типа "растр-вектор" и "вектор-растр".

Особенности организации данных в ГИС. Определение положения точек на поверхности Земли. Координатные данные. Основные картографические проекции.

3. Особенности создания картографических баз данных (КБД). Анализ данных и моделирование. Оверлейные операции. Операции вычислительной геометрии. Операции с трехмерными объектами. Цифровые модели местности, методы их построения. Анализ данных и моделирование с использованием статистических методов. ГИС как средство принятия решений.

4. Организация работы с ГИС. Классические ГИС профессионального уровня. Классические ГИС настольного типа (Arc View, MapInfo, WinGis и др.).

5. Структура ГИС MapInfo. Файловая организация хранения картографической и тематической информации в рамках ГИС MapInfo. Возможности импорта и экспорта графической и тематической информации. Векторизация растровой информации методом цифровое по "подложке". Создание картографических и атрибутивных баз данных. Язык программирования MapBasic. Модули MapInfo.

6. Структура ГИС Arc View. Файловая организация хранения картографической и тематической информации. Язык программирования Arc Avenu. Скрипты в составе Arc View. Модули Arc View.

7. Примеры реализации ГИС. Глобальные проекты (GRID). Международные проекты. Национальные программы. Региональные ГИС. Локальные ГИС.

Тема 9. Средства анализа и управления сетями

Функции и архитектура систем управления сетями. Архитектура систем управления сетями. Стандарты систем управления. Стандарты систем управления SNMP, OSI. Мониторинг и анализ сетей.

Тема 10. Основы технологии «клиент-сервер». Языки гипертекстовой разметки. Серверное программное обеспечение. Клиентская часть приложений

Основы технологии «клиент-сервер». Схема взаимодействия клиента и сервера. Серверы приложений: типы, назначение, функции. Серверы Интернет. Web-сервер, его функции и предъявляемые к нему требования. Microsoft Internet Information Services (IIS). Web-сервер Apache. Основы Web-программирования: основные понятия и термины. Web-дизайн и Web-программирование.

Принципы гипертекстовой разметки. Структура гипертекстовых документов. Идентификаторы URI. Коды языков. Понятие о стандартном обобщенном языке разметки SGML. Консорциум W3C. Версии языка гипертекстовой разметки HTML. Понятие о расширяемом языке разметки XML.

Основы разработки сетевых приложений. Принципы построения серверной части программного обеспечения. Основные задачи, выполняемые серверными программами. Средства создания программ, выполняемых на стороне сервера. Их характеристика и назначение.

Характеристика типовых задач, решаемых клиентской частью приложений. Функциональные возможности клиентской части. Обзор

инструментальных средств разработки программ, выполняющихся на стороне клиента. Их назначение и возможности.

V. МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРАКТИЧЕСКИМ СЕМИНАРСКИМ ЗАНЯТИЯМ

Для выполнения практических (семинарских) занятий студенту необходимо иметь конспект лекций. Студент знакомится с заданием и выполняет его, опираясь на конспект лекций. При выполнении задания связанного с программированием, студент разрабатывает алгоритм решения предложенной задачи, набирает текст программы (реализующей алгоритм), отлаживает и тестирует программу.

VI. ПРАКТИЧЕСКИЙ СЕМИНАР (ПЕРЕЧЕНЬ ОСНОВНЫХ ТЕМ)

5 семестр

1. Знакомство с пакетом MATLAB. Встроенные функции, их назначение и использование для решения различных задач.
2. Основы программирования в среде MATLAB.
3. Построение баз данных в Delphi.

Разработка приложений баз данных на основе современных технологий в среде Delphi. Создание приложений обмена данными в среде Delphi. Разработка CORBA приложений в среде Delphi.

6 семестр

4. Графические редакторы (Corel, Photoshop)
5. САПР. AutoCAD.

Построение основных графических примитивов. Способы привязки при построении графических образов. Возможности редактирования графических образов в среде AutoCAD. Работа с текстом в AutoCAD. Штриховка и заливка цветом графических объектов. Нанесение размеров и допусков.

6. ГИС (ArcView, ArcGIS, MapInfo, Панорама)

Знакомство с ГИС ArcView (ArcGIS). Основные функциональные

возможности пакета. Понятие проекта. Создание, открытие, состав и редактирование проектов в ArcView. Понятие вида. Особенности создания вида, его назначение, редактирование в ArcView. Понятие темы. Виды тем, их создание, расположение, редактирование в ArcView. Понятие атрибутивной информации. Таблицы свойств атрибутов, их создание, привязка и использование в ArcView. Создание собственного проекта в ArcView.

Знакомство с ГИС MapInfo. Основные функциональные возможности пакета. Создание собственного проекта в MapInfo.

Знакомство с ГИС Панорама (Карта 2005). Основные функциональные возможности пакета. Создание собственного проекта.

7. Форматирование текста и списков. Форматирование таблиц. Работа с формами. Работа с объектами. Создание динамической Web-страницы. Разработка Web-приложения.

VII. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРИМЕНЕНИЮ СОВРЕМЕННЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ДЛЯ ПРЕПОДАВАНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ (В Т. Ч. РАЗРАБОТАННЫЕ ВЕДУЩИМИ ПРЕПОДАВАТЕЛЯМИ)

При преподавании данной дисциплины можно использовать электронные тестирующие и учебные материалы. Методические указания прилагаются к этим материалам.

VIII. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПРОФЕССОРСКО- ПРЕПОДАВАТЕЛЬСКОМУ СОСТАВУ ПО ОРГАНИЗАЦИИ МЕЖСЕССИОННОГО И ЭКЗАМЕНАЦИОННОГО КОНТРОЛЯ ЗНАНИЙ СТУДЕНТОВ (МАТЕРИАЛЫ ПО КОНТРОЛЮ КАЧЕСТВА ОБРАЗОВАНИЯ)

Преподаватель готовит контролирующие материалы в виде тестов, задач и в другой форме. Во время проведения контроля знаний студентов преподаватель объясняет студентам правила работы с контролируемыми материалами и выдаёт эти материалы студентам. После истечения установленного времени контролирующие материалы собираются и обрабатываются.

IX. КОМПЛЕКТЫ ЗАДАНИЙ ДЛЯ СЕМИНАРСКИХ ЗАНЯТИЙ, КОНТРОЛЬНЫХ РАБОТ, ДОМАШНИХ ЗАДАНИЙ

Задания для практических семинаров, контрольных работ и домашних заданий берутся из книг, реквизиты которых приведены в рабочей программе.

X. ФОНД ТЕСТОВЫХ И КОНТРОЛЬНЫХ ЗАДАНИЙ ДЛЯ ОЦЕНКИ КАЧЕСТВА ЗНАНИЙ ПО ДИСЦИПЛИНЕ

Задание 1

1. Изобразить на плоскости кривую, заданную параметрически, касательную и нормаль к ней в указанной точке $x(t)=t^2$, $y(t)=t^3$, $(x_0, y_0)=(1, 1)$.
2. По таблице значений проинтерполировать табличную функцию тремя способами, построить графики интерполирующих функций.

i	1	2	3	4	5	6	7
x_i	1.20	1.50	1.75	2.15	2.55	2.75	3.00
y_i	33.11	34.81	36.59	38.47	40.44	37.52	34.70

3. Вычислить интеграл по формулам Гаусса, оценить погрешность

$$\int_0^{2\pi} \sqrt{1 - \left(\frac{1}{4}\right) \cdot \sin^2(x)} dx$$

, при $n=8$.

4. Решить уравнение $y' = +2x - 3$, $y(0)=2$, шаг 0.1, на отрезке $[0, 5]$. Решить программным путем и с помощью функции **rkfixed**. Сравнить результаты.
5. Найти корни уравнения $(x-1)^2 - 0.5e^x = 0$ методом хорд с точностью 10^{-4} .
6. Построить график функции $f(x) = \sqrt[3]{(x+3)^2 x}$ и подтвердите изображение аналитическим исследованием.
7. Вычислить производную функции $f(x) = x^2 - 9$, $x_0 = 3$. Найти значение производной в указанной точке. Вычислить односторонние производные функции $g(x) = |f(x)|$ в этой точке. Построить графики обеих функций.

Задание 2

1. Найти корень уравнения $x^3 - 4x^2 + 10x - 10 = 0$ методом секущей с точностью 10^{-4} .
2. Проверить совпадения интерполяционных полиномов $Q(z)$ и $L(z)$ для всех точек заданного диапазона.
3. Составить программу решения системы дифференциальных уравнений методом Рунге – Кутты с автоматическим выбором длины шага. Построить график приближенного решения на отрезке $[0, 20]$, допустимая погрешность $\varepsilon = 10^{-6}$.
$$\begin{cases} y_1' = 2 + y_1 y_2 - 9.533 y_1 & y_1(0) = 1 \\ y_0' = 8.533 y_1 - y_1^2 y_2 & y_2(0) = 4.266 \end{cases}$$
4. Изобразите на графике заданной функции и подтвердите построение аналитическим исследованием: найти координаты точек пересечения с координатными осями, найти и построить наклонные асимптоты. Записать уравнение вертикальных асимптот

$$f(x) = \frac{4x^3 + 3x^2 - 8x - 2}{2 - 3x^2}$$

5. Вычислить интеграл по формуле трапеций Симпсона с точностью до $0,5 \cdot 10^{-3}$, определяя шаг интерполирования с помощью двойного перечета

$$\int_0^1 \frac{1}{1-x+x^2} dx$$

6. Найти аналитически и графически \min и \max функции $f(x)$ на отрезке $[-2,4]$;

$$1 + \sqrt[3]{2x^2(x-6)} = 0$$

7. Вычислить произведение

$$P = \prod_{k=0}^4 \left[1 - \frac{4x^2}{(2k+1)^2 \pi^2} \right]$$

Задание 3

1. Найти корень уравнения $\sin(x) - x \cdot \cos(x) = 0$ методом Ньютона с точностью $\varepsilon = 10^{-4}$, на отрезке $\pi < x < 3\pi/2$.

2. Вычислить произведение

$$P = x \cdot \prod_{k=1}^n \left(1 - \frac{x^2}{k^2 \pi^2} \right)$$

3. Построить графики плотности распределения и функции распределения Фишера для значений n и m :

1) $n=6, m=4$;

2) $n=8, m=5$.

4. Вычислить интеграл от неограниченной функции по заданному отрезку непосредственно и через предел. Построить график подынтегральной функции и график функции, заданной интегралом с переменным пределом.

5. Решить систему дифференциальных уравнений на отрезке $[0,20]$, допустимая погрешность $\varepsilon = 10^{-6}$

$$\begin{cases} y_1' = y = y_3 & y_1(0) = 0 \\ y_2' = -y_1 y_3 & y_2(0) = 1 \\ y_3' = -0.51 y_1 y_2 & y_3(0) = 1 \end{cases}$$

6. Вычислить интеграл, проверив правильность вычислений, взяв производную, и построить графики семейства первообразных

$$\int \frac{1}{\sin^2(x) \cdot (1 - \cos(x))} dx$$

7. Найти асимптоты и построить график функции

$$y = \frac{4x^2 + 9}{4x + 8}$$

Задание 4

1. Найти по определению производную функции $f(x) = \operatorname{tg}(x^3 + x^2 \cdot \sin(2/x))$. Вычислить значение производной в точке $x=0$, имея в виду, что $f(0)=0$.

- Решить на отрезке [1,3] задачу Коши $x^2y' + xy + 1=0$ с постоянным шагом. Получить графики решений, вычисленных с шагом h и $6h$. Сравнить сточным решением $xy = 1 - \ln|x|$.
- Найти корень уравнения $\sin(x) + \cos(x) - 1.78e^{x-3} = 0$ методом половинного деления на отрезке [1,2] с точностью 10^{-5} . Определить число итераций.
- Проанализировать при помощи гистограммы случайную величину, заданную 20 значениями генератора случайных чисел – $\text{rnd}(200)$ (использовать функцию $\text{floor}(x)$ и $\text{ceil}(x)$). Построить график случайной величины и её гистограммы.
- Вычислить интеграл с точностью 10^{-6} , используя квадратные формулы Гаусса

$$\int_0^{\infty} \frac{e^{-x}}{1 + \sqrt{x}} dx$$

- Исследовать ряд на сходимость

$$\sum_{n=1}^{\infty} \frac{1}{\sqrt{n}} \cdot \sin\left(\frac{2}{2n^2 + 1}\right)$$

- Найти общее решение $y'' + 2y' + 5y = 0$.

Задание 5

- Изобразить график функции

$$f(x) = \frac{x^3 - 27x + 54}{x^3}$$

и подтвердить построение аналитическим исследованием: построить график функции второй производной, найти нули второй производной. Найти координаты точек перегиба, если они есть.

- Решить задачу Коши $y' = xy/2$, $y(0) = 1$ с шагом 0.1 на отрезке [0,5]. Решение представить графически.
- Рассчитать числовые характеристики статистического распределения случайной величины, заданной 20 значениями генератора случайных чисел – $\text{rnd}(200)$. Построить график случайной величины, на котором показать среднее, а также линейное и стандартное отклонение.
- Определить область начальных приближений x_0 , для которых итерационный процесс $x_{n+1} = (x_n^3 + 1)/20$ сходится.
- Методом хорд решить уравнение $x^5 + ax - 1 = 0$, $a = 1(1)5$.

- Вычислить интеграл по формуле Симпсона, оценить погрешность

$$\int_0^1 \frac{\sin(x)}{x} dx, n=10.$$

- Изобразить график поверхности, линии уровня функции в квадрате $-4 < y < 4$, $-4 < x < 4$. Указать приближенно точки локальных экстремумов, если они есть.

$$z = xy \exp\left(\frac{-x^2}{3} - \frac{y^2}{4}\right)$$

Задание 6

- По заданной таблице значений найти регрессионные коэффициенты, построить график линейной регрессии и определить точку максимального отклонения.

i	1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	---	----

x_i	-2.0	-1.5	-1.0	-0.5	0.0	0.5	1.0	1.5	2.0	2.5
y_i	11.3	13.2	14.1	14.1	13.0	10.9	7.9	3.8	-1.3	-7.4

- Найти корень уравнения $(x+1)^{0.5} \cdot x^{-1} = 0$ с точностью 10^{-4} методом Ньютона и секущей, сравнить решение.
- Найти аналитически и графически \min и \max функции на отрезке $[-4, 2]$.

$$f(x) = \frac{4x}{x^2 + 4}$$

- Найти точки разрыва функции и определить их тип:

$$g(x) = \frac{1}{1 + \exp\left(\frac{1}{x}\right)}, \quad f(x) = (e^x - 1) \cdot \sin\left(\frac{1}{x}\right), \quad h(x) = \frac{x + 1}{x^2 + 2x + 1}$$

- Решить систему дифференциальных уравнений методом Рунге-Кутты. Построить график решения с шагом 0.01 на отрезке $[0, 10]$

$$\begin{cases} y_1' = -xy_2 & y_1(0) = 0 \\ y_2' = y_1/x & y_2(0) = 1 \end{cases}$$

- Вычислить интеграл по методу трапеции, оценить погрешность при $n=50$

$$\int_0^1 \frac{1}{\sqrt{x}(4-x)} dx$$

- Сравнить бесконечно малые в нуле функции. Построить графики функций в окрестности предельной точки

$$\mu(x) = \left(\sin\left(\frac{x+1}{x}\right)\right)^4 \cdot \sqrt[5]{x}, \quad \gamma(x) = 5\sqrt[5]{x}, \quad \beta(x) = \operatorname{arctg}\left(\sqrt[4]{x^{1.5}}\right), \quad \alpha(x) = \exp(-\sqrt{x}) - 1$$

Задание 7

- Найти с точностью 10^{-2} длины дуги лемнискаты $r^2 = 2\cos(\theta)$, $\theta = (\pi/36)^2 * (\pi/4)$.
- Методом спуска с точность 10^{-3} найти решение системы:

$$\begin{cases} e^{x+y} - x^2 + y = 2; \\ (x+0.5)^2 + y^2 = 1. \end{cases}$$
- Решить систему дифференциальных уравнений с шагом 0.1 на отрезке $[0, 10]$

$$\begin{cases} y_1' = y_2, & y_1(0) = 0.3, \\ y_2' = -0.2y_2 - 10 \sin(y_1); & y_2(0) = 0. \end{cases}$$
- Написать программу для расчета полуширины нормального распределения: $P(x, 0.1)$. Границы ширины находятся из условия: $P(x_p, 0.1)/P(0, 0.1) = 0.5$
- Сравните бесконечно малые в нуле функции. Постройте графики функций в окрестности предельной точки.

$$\mu(x) = \sqrt[3]{x} \cdot \sin\left(\frac{x+1}{\sqrt{x}}\right), \quad \gamma(x) = \sqrt[3]{x}, \quad \beta(x) = \ln(1 + \sqrt{x} \sin(\sqrt{x})), \quad \alpha(x) = \sqrt{x} \sin(x)$$

- Вычислить интеграл по формуле Симпсона, оценить погрешность.

$$\int_{-1}^1 \frac{1}{3+x} dx, \quad \text{при } n=5.$$

- Методом простой итерации найти корни уравнений: $1.4^x - x = 0$ и $e^x - 6x - 3 = 0$.

Задание 8

1. Составить алгоритм локальной линейной интерполяции исходных данных и рассчитать меру отклонения линейной интерполяции от глобальной по формуле:

$$\rho = \sum_P (L(x_p) - l(x_p))^2$$

где $l(x)$ – линейная интерполяция.

2. Найти общее решение уравнения $y'' + a_1 y' + a_2 y = 0$. Решить задачу Коши $y'' + a_1 y' + a_2 y = 0$, $y(a) = y_0$, $y'(a) = y_1$. Проверить правильность решения встроенной функции `rkfixed`. Изобразить график. $a_1 = -4$, $a_2 = 4$, $y(a) = 0$, $y'(a) = 1$, $a = 0$.
3. Для функции $f(x) = x^2 - 2e^{x-1}$, где $x_0 = 1$, $n = 5$ записать формулу Тейлора указанного порядка в окрестности точки x_0 . Изобразить график функции и её многочлена Тейлора.
4. Найти точки разрыва функций и определить их тип

$$g(x) = \operatorname{arctg}\left(\frac{1}{x - \pi}\right), \quad f(x) = \sqrt{|x|} \cdot \sin\left(\frac{1}{x}\right), \quad h(x) = 3^{\frac{1}{x}}$$

5. Найти плотность распределения вероятности суммы двух независимых непрерывных случайных величин, заданных своими плотностями вероятностей

$$P_1 = \begin{cases} 1/2, & x \in [0, 2]; \\ 0, & x \notin [0, 2]; \end{cases} \quad \text{и} \quad P_2 = \begin{cases} 1/2, & x \in [0, 2]; \\ 0, & x \notin [0, 2]. \end{cases}$$

6. Найти корень уравнения $e^x - x^2 + 12 = 0$ на отрезке $[-5, -3]$. Построить график

7. Вычислить интеграл методом прямоугольников, разбив участок $[1, 2.5]$

$$\int \frac{1}{x^2} \cdot \sin\left(\frac{1}{x}\right) dx$$

Задание 9

1. Вычислить сумму ряда $S = \frac{\pi}{6} - \frac{(\frac{\pi}{6})^3}{3!} + \frac{(\frac{\pi}{6})^5}{5!} - \dots + (-1)^n \frac{(\frac{\pi}{6})^{2n+1}}{(2n+1)!}$ с точностью 10^{-4} . Вывести значение суммы и число членов ряда.

2. Решить систему уравнений $\begin{cases} y_1' = \cos(y_1 + 2y_2) + 2 \\ y_2' = \frac{2}{x + 2y_1^2} + x + 1 \end{cases}$ $y_1(0) = 1$, $y_2(0) = 0.005$, шаг 0.1, на

отрезке $[0, 10]$.

3. Вычислить интеграл $\int \sqrt{6x - 5} dx$ на отрезке $[0, 1]$ методом трапеций.
4. Построить выборку двух пар величин при помощи генератора случайных чисел **rnd(150)** и **rnd(250)**, а также графики уравнений линейной регрессии.
5. Найти методом простых итераций приближенное решение линейной системы $Ax = b$.

$$A = \begin{pmatrix} 100 & -14 & 13 \\ 0.5 & 200 & 9.5 \\ -9 & 9 & 300 \end{pmatrix}, \quad B = \begin{pmatrix} -1232 \\ 326 \\ 4335 \end{pmatrix}.$$

6. Найти корень уравнения методом простой итерации $x^3 + 4 \sin(x) = 0$.
7. Найти точки разрыва заданных функций и определить их тип, построить график:

$$f(x) = \sqrt[3]{x-1} \operatorname{arctg} \frac{1}{x-1}, \quad g(x) = \operatorname{arctg} \frac{1}{x-1}, \quad h(x) = \frac{x-1}{x^2 - 2x + 1}.$$

Задание 10

1. Вычислить интеграл $\int \frac{\cos(x) - \sin(x)}{(1 + \sin(x))^2}$, проверить правильность вычислений взяв производную, и построить графики семейства первообразных.
2. Найти корни уравнения $x^5 - 2.2x^3 + 0.5x^2 - 7x = 3.4$ с помощью функции polyroots.
3. Составить алгоритм локальной линейной интерполяции исходных данных и рассчитать меру отклонения линейной интерполяции от глобальной по формуле: $S = \sum_p (L(xp) - l(xp))^2$, где $l(x)$ – линейная интерполяция.
4. Найти корни уравнения методом Ньютона: $e^x + 1 + 6x^3 = 0$, $b = 1 + 0.5k$, $k = 0(1)4$.

5. Решить систему уравнений на отрезке $[0, 20]$ с точностью 10^{-6}

$$\begin{cases} y_1' = y^3 \\ y_2' = y^4 \\ y_3' = \frac{y_1}{(y_1^2 + y_2^2)^{3/2}} \\ y_4' = \frac{-y_2}{(y_1^2 + y_2^2)^{3/2}} \end{cases} \quad \text{при}$$

$$y_1(0) = 0.5, y_2(0) = 0, y_3(0) = 0, y_4(0) = \sqrt{3}.$$

6. Найти аналитически и графически точки, в которых достигаются \min и \max значения на отрезке $[-4, -1]$ непрерывной функции $f(x) = 8 + \frac{8}{x} - \frac{x^2}{2}$. Найти нуль функции на заданном отрезке.
7. Найти точки разрыва следующих функций, определить их тип. Построить графики. $f(x) = \sqrt{|x| - 1} \sin \frac{1}{x - 1}$, $g(x) = \operatorname{arccctg} \frac{1}{x + 1}$, $h(x) = 3^{-\sqrt{x^3}}$

Задание 11

1. Вычислить таблицу значений функции, её первой и второй производных для набора значений аргумента. Построить графики. $f(x) = x^4 - 2.9x^3 + 6.5x^2 - 7x - 5.4$
2. Решить систему уравнений на отрезке $[0, 10]$ с шагом 0.1
$$\begin{cases} y_1' = e^{-(y_1' + y_2')} + 2x \\ y_2' = 2y_1^2 + y_2 \end{cases}$$
3. Написать программу для расчета параметров a и d нормального распределения $P(x, a, d)$, удовлетворяющих условиям: $P(-2, a, d) = 0.2$ и $P(-3, a, d) = 0.1$.
4. Решить уравнение методом Эйлера $y' = 2y$, $y(0) = 1$, шаг 0.1 на отрезке $[0, 5]$.
5. Методом парабол стандартным и модифицированным с точностью 10^{-4} найти корни уравнения: $x^3 - \sin(x) = a$, $a = 1(0.1)3$.
6. Методом хорд вычислить корень уравнения с точностью $\varepsilon = 2 * 10^{-3}$ $f(x) = x^3 - 0.2x^2 - 0.2x - 1.2 = 0$.
7. Вычислить интеграл по формуле трапеций с точностью до $\frac{1}{2} * 10^{-4}$, определяя шаг

$$\text{интегрирования } h \text{ по оценке остаточного члена. } \int_0^{2\pi} x \sin(x) dx.$$

Задание 12

1. Найти асимптоты и построить график функции $y = (x + 16) / \sqrt{9x^2 - 8}$.
2. Решить систему дифференциальных уравнений методом Рунге – Кутты

$$\begin{cases} y_1' = \sqrt{x^2 + 2y_1^2} + y_2 \\ y_2' = \cos(2y_2) + y_1 \end{cases}$$
 при $y_1(0)=0.5, y_2(0)=1$ с шагом 0.1 на отрезке $[0,10]$.
3. Найти по заданному распределению $P_\xi(x)$ аргумента функцию случайной величины $\eta=f(\xi)$ и её распределение $P_\eta(x)$.

$P_\xi(x)$	$\eta=f(\xi)$
$\frac{1}{\pi(1+x^2)}$	$\ln(\xi)$

4. Исследуйте функцию $f(x) = \frac{x}{4} \exp(-\frac{x^2}{8})$, заданную интегралом с переменным верхним пределом $\int_a^x f(x) dx$, $a=0$.
5. Разложить $x/\sqrt{1+x}$ в степенной ряд по степеням $x/(1+x)$.
6. Найти точку пересечения прямой и плоскости, построить график. $\frac{x-1}{2} = \frac{y-1}{-1} = \frac{z-8}{2}$ и

$$\begin{aligned} 4x+2y-z &= 0 \\ x-1 &= 2-2y=z-8 \\ z &= x-1-2+2y+8 \\ z &= 2y+x+5 \end{aligned}$$
7. Найти корень уравнения методом половинного деления на отрезке $[0,0.85]$ с точностью 10^{-5} . Определить число итераций. $x - \frac{1}{3 + \sin(3.6x)} = 0$

Задание 13

1. Написать программу для расчета параметров a и d нормального распределения $P(x,a,d)$, удовлетворяющих условиям: $P(-2,a,d)=0.2$ и $P(-3,a,d)=0.1$.
2. По таблице значений проинтерполировать табличную функцию тремя способами, построить графики интерполирующих функций.

i	1	2	3	4	5	6	7
x_i	1.20	1.50	1.75	2.15	2.55	2.75	3.00
y_i	1.261	1.280	1.291	1.306	1.321	1.336	1.352

3. Решить задачу Коши $\begin{cases} y_1' = \sin(y_1) \cos^2(y_2) \\ y_2' = \cos(y_1) \cos(y_2) \end{cases}$, $y_1(a)=-0.6, y_2(a)=2$ на отрезке $[2,5]$ с постоянным шагом $h=0.1$. Изобразить графики решений вычисленных с шагом h и $2h$.
4. Найти корень уравнения $x^2 + 10x - 10 = 0$ методом половинного деления с точностью 10^{-5} на отрезке $[0,1]$.
5. Записать для данной функции разложение в ряд Тейлора в окрестности точки x_0 . Изобразить график функции $f(x) = \cos\left(\frac{\pi}{4}\right)x, x_0=2$.
6. Исследовать на сходимость знакопеременный ряд $\sum_{n=1}^{\infty} (-1)^n \sqrt{n} \left(2^{\frac{1}{\sqrt{n}}} - 1 \right)$.

7. Изобразить на плоскости кривую, заданную параметрически, касательную и нормаль к ней в указанной точке: $x(t)=\cos^2(t)+\cos(t)$, $y(t)=\cos(t)\sin(t)+\sin(t)$, $(x_0, y_0) = \left(\frac{1+\sqrt{2}}{2}, \frac{1+\sqrt{2}}{2}\right)$.

Задание 14

1. Найти корень уравнения $x - \frac{1}{3 + \sin(3.6x)} = 0$ на отрезке $[0, 0.085]$ методом Ньютона, методом секущей и с помощью встроенной функции root. Сравнить результаты.

2. Вычислить $\int \frac{\exp(\sqrt{\frac{3-x}{3+x}})}{(3+x)\sqrt{9-x^2}} dx$ на отрезке $[0, 2]$ методом трапеции и Симпсона, при $n=50$.

3. По заданной таблице значений найти регрессионные коэффициенты, построить график линейной регрессии и определить точку максимального отклонения.

i	1	2	3	4	5	6	7
x_i	1.20	1.50	1.75	2.15	2.55	2.75	3.00
y_i	-0.670	-0.065	0.177	0.606	1.154	1.221	2.308

4. Решить систему дифференциальных уравнений методом Рунге – Кутты

$$\begin{cases} y_1' = \sqrt{x^2 + 2y_1^2} + y_2 \\ y_2' = \cos(2y_2) + y_1 \end{cases} \text{ при } y_1(0)=0.5, y_2(0)=1 \text{ с шагом } 0.1 \text{ на отрезке } [0, 10] \text{ с шагом } h \text{ и } 2h.$$

5. Сравнить бесконечно малые в нуле функции. Построить графики функций в окрестности предельной точки.

$$\alpha(x) = 2^{10x} - 1, \beta(x) = \sin(\sqrt[4]{x^3}), \gamma(x) = x, \mu(x) = x \sin\left(\frac{x^2 + 1}{x}\right).$$

6. Найти точки разрыва функции, построить графики. $f(x) = \sin(\sqrt[4]{x^3})$, $h(x) = \text{cth}(x-1)^2$, $g(x) = \text{th}^{-1}\left(\frac{1}{x+1}\right)$.

7. Решить задачу Коши на отрезке $[1, 4]$ с шагом $h=0.1$ $\begin{cases} y_1' = y_2 \ln x \\ y_2' = y_1 + y_2^2 \end{cases}$ $y_1(a)=-2, y_2(a)=-1$.

Задание 15

1. Незвестная переменная задана с шагом 0.15 для 20 значений, начиная с $x = -0.8$. Зависимая переменная задана при помощи генератора случайных чисел – rnd(10). Провести глобальную интерполяцию полученных данных и построить график интерполяции.

2. Методом Ньютона найти корни уравнения $x^2 - \cos^2(\pi x) = 0$.

3. Решить систему уравнений $\begin{cases} y_1' = (y_2 - y_1)x \\ y_2' = (y_2 + y_1)x \end{cases}$, при $y_1(0)=1, y_2(0)=1$ с шагом 0.1 на отрезке $[0, 10]$. Построить график приближенного решения. Проверить сходимость.

4. Вычислить приближенно интеграл $\int_0^{\frac{\pi}{2}} dx \int_0^{\frac{\pi}{2}} \sin(x+y) dy$.

5. Найти точки разрыва функции и определить их тип. $f(x)=\sin(x)\cos\left(\frac{1}{x}\right)$,
 $g(x)=\arctg^3\left(\frac{1}{x+3}\right)$, $h(x)=\exp\left(x-\frac{1}{x}\right)$.
6. Методом хорд и методом секущих найти корни уравнения с точностью 10^{-4} $5.3x - e^{0.5x} = 0$.
7. Найти min и max значение функции $f(x)=\sqrt[3]{2(x-2)^2(8-x)}-1$ на отрезке $[0,6]$. Построить график.

Задание 16

1. Найти корень уравнения методом Ньютона с точностью $\varepsilon = 10^{-5}$ на отрезке $[0.5,1.5]$
 $\ln(|x|+1) \cdot \cos(x) - x^3 = 0$.
2. Рассчитать числовые характеристики статистической модели появления редких случайных событий (распределение Пуассона). Вероятность наступления редкого события К раз равна: $P(k, a) = \frac{a^k}{k!} e^{-a}$, где а – среднее число событий.
3. Решить систему уравнений $\begin{cases} \sqrt{y-x+5} = 3 \\ \sqrt{x+y-5} = 11-2y \end{cases}$. Результат представить графически.
4. Изобразить линии, заданные в декартовых координатах явно уравнением $y = \frac{|x|(x-2)}{x+2}$ и не явно уравнением $\frac{y^2}{3} - \frac{x^2}{4} - 1 = 0$, $x_0 = -3$, $(x_0, y_0) = \left(1, \frac{\sqrt{15}}{2}\right)$. Записать уравнение касательной и нормали к каждой кривой в указанных точках и изобразить их на графике.
5. Вычислить интеграл $\int x^2 \sqrt{1-x^2}$ на отрезке $[0,1]$ методом трапеций.
6. Найти все корни уравнения $x^5 - 3.2x^3 + 9.5x^2 - 7x = 7.5$ (используя встроенные функции).
7. Вычислит сумму ряда с точностью 10^{-4} . $S = \frac{1}{x} - \frac{1}{3x^3} + \frac{1}{5x^5} - \dots + (-1)^n \frac{1}{(2n+1)x^{2n+1}}$

Задание 17

1. Сравните бесконечно малые в нуле функции. Построить графики функций в окрестности предельной точки $\alpha(x) = x \operatorname{tg}(\sqrt{x})$, $\beta(x) = \sqrt{1+x} - 1$, $\gamma(x) = \sqrt{x^3}$, $\mu(x) = \sqrt{x^3} \sin\left(\frac{1}{x}\right)$.
2. Составить алгоритм сплайн – интерполяции для таблицы экспериментальных данных, в которой x меняется с шагом 0.2 от 0 до 1, а вектор y создается генератором случайных чисел rnd(1).
3. Исследуйте неоднородную систему линейных алгебраических уравнений $Ax = b$, для двух различных правых частей $b=b_1$ и $b=b_2$.

$$A = \begin{pmatrix} -11 & -6 & -1 & 4 & 9 \\ 2 & 1 & 9 & -1 & -11 \\ -11 & -6 & 5 & 6 & 1 \\ -9 & -5 & 8 & 3 & -2 \\ -20 & -11 & 13 & 9 & -1 \end{pmatrix}, \quad b_1 = \begin{pmatrix} 10.8 \\ -36.2 \\ -17.2 \\ -25.4 \\ -42.6 \end{pmatrix}, \quad b_2 = \begin{pmatrix} 11.8 \\ -35.8 \\ -16.2 \\ -26.73 \\ -44.6 \end{pmatrix}.$$

4. Вычислить интеграл по формуле Симпсона с точностью 10^{-3} $\int_{10}^{100} \frac{\ln(1+x)}{x} dx$.
5. Найти точки разрыва функции и определить их тип: $f(x) = \sqrt[3]{x-1} \arctg\left(\frac{1}{x-1}\right)$,
- $$g(x) = \frac{1}{1 + \exp\left(\frac{-1}{x}\right)}$$
6. Решить на отрезке $[0,1.5]$ задачу Коши $y' + ytg(x) = \sec(x)$ с постоянным шагом. Получить графики решений, вычисленных с шагами h и $4h$. Сравнить с точным решением: $y = \sin(x) - \cos(x)$.
7. Изобразить кривую, заданную в полярных координатах $p(x) = 2 \cos(\phi) + 3$ и
- $$p(x) = \frac{2}{\sin(\phi)} + 1.$$

Задание 18

1. Решить систему уравнений $\begin{cases} \sqrt{2x+y+2} = 3 \\ \sqrt{x+2y+5} = y-x \end{cases}$ программным путем и с помощью блока Giken...Find.
2. Независимая переменная задана с шагом $h=0.15$ для $n=20$ значений. Зависимая переменная задана при помощи генератора случайных чисел по формулу $y=h*i^2 + \text{rnd}(10)$. Составить алгоритм сглаживания данных.
3. Вычислить интеграл с точностью 10^{-5} , используя квадратные формы Гаусса $\int_0^1 \frac{dx}{\sqrt{x}(e^{0.4x} + 1.5)}$.
4. Определить корни уравнения $x^6 - 24x^5 - 375x^4 + 375x^2 + 24x - 1 = 0$ методом Ньютона и хорд, сравнить результаты.
5. Решить задачу Коши $\begin{cases} y'_1 = 2\sqrt{3x^2 + y_1^2 + y_1} \\ y'_2 = \sqrt{x^2 + y_1^2 + y_1} \end{cases}$ на отрезке $[-1,3]$ с постоянным шагом $h=0.1$. Изобразить графики решений, вычисленных с шагом h и $2h$.
6. Исследуйте и, если решение существует, найдите по формулам Крамера решение системы $Ax = b$
- $$A = \begin{pmatrix} 0.040 & 0.032 & 0.500 & 0.350 \\ -0.020 & 0.200 & 0.053 & 0.200 \\ 0.850 & 0.267 & 0.089 & 0.208 \\ 0.023 & 0.450 & -0.067 & 0.200 \end{pmatrix}, \quad B = \begin{pmatrix} 2.481 \\ 1.182 \\ 8.520 \\ 2.205 \end{pmatrix}.$$
7. Вычислить производную функции $f(x) = x^2 - 9$, $x_0=3$. Найти значение производной в указанной точке. Вычислить односторонние производные функции $g(x) = |f(x)|$ в этой точке. Построить графики обеих функций.

XI. ТЕМАТИКА РАСЧЕТНО-ГРАФИЧЕСКИХ РАБОТ

Базы данных:

1. Разработать приложение справочной службы аэропорта. Программа должна выдавать справки об авиарейсах (например, Благовещенск-Москва). По требованию сообщать следующую информацию: количество свободных мест на заданное число, проданные места на заданное число, проданные места на весь месяц.
2. Разработать помощник экзаменатора. Экзаменационные вопросы и ответы хранятся в файлах на диске. Каждый вопрос имеет бал сложности. Необходимо подобрать 5 вопросов из разных разделов курса, имеющих в сумме бал сложности N и вывести их на экран. Предусмотреть тренировочный режим работы, когда возможен вывод ответов на предложенные вопросы. Доступ к тренировочному режиму предоставляется по паролю.
3. Разработать справочную систему по стандартным функциям языка Турбо Паскаль.
4. Разработать справочную систему по операторам языка Турбо Паскаль.
5. Разработать справочную систему по визуальным компонентам Delphi.
6. Разработать справочную систему по стандартным процедурам Object Pascal.
7. В альпинистском клубе ведется хроника восхождений. Записываются даты начала и завершения каждого восхождения, имена и адреса участвующих в нем альпинистов, название и высота горы, район или страна, где она расположена. Создать программу ведения БД, которая включает в себя:

- ввод и корректировку исходных данных;
- вывод информации по запросам: а) наименование вершины, потребовавшей самого длительного восхождения и страны, где она расположена; б) ФИО и адрес альпиниста, совершившего наибольшее число восхождений.

Для хранения информации использовать два файла.

8. Разработать программу ведения БД данных риэлторской фирмы. Данные о проданных квартирах хранятся в двух файлах – основном и справочном. Основной файл содержит сведения о сделках и имеет следующие поля: дата сделки, покупатель, идентификационный номер продаваемой квартиры, цена. Файл-справочник содержит информацию о проданных квартирах: общая площадь квартиры, число комнат. Номер записи в файле-справочнике соответствует идентификационному номеру квартиры. Предусмотреть ввод и корректирование исходной информации и вывод ведомости о сумме продаж, совершенных в каком-либо году, месяце. Ведомость имеет вид:

Месяц	Сумма продаж	По числу комнат в квартире		
		1	2	3
Январь				
...				
Итого	х			

9. Создать программу ведения БД сотрудники. Информация хранится в двух файлах. Файл 1 – список сотрудников: ФИО, код должности и подразделения. Файл 2 – справочник: наименование должности, зарплата за 1 час работы. Количество отработанных часов для каждого сотрудника вводится с клавиатуры ПК. Ведение БД включает в себя: а) ввод информации о сотрудниках и запись ее на диск; б) удаление ненужной информации; в) корректировка записей БД; г) вывод расчетной ведомости для каждого подразделения, которая имеет вид:

№	ФИО	Начислено	Подходный налог	Отчисления в пенсионный фонд	К выплате

Динамические структуры

10. Построить имитационную модель бензоколонки. На бензоколонке k стоек (1 стойка обслуживает один автомобиль). Каждый автомобиль обслуживается s секунд, интервал между прибытием на бензоколонку автомобилей является случайной величиной, распределенной по закону $P(x)$. Если все стойки заняты, автомобиль становится в очередь. Для заданных $P(x)$ и s определить возможно меньшее значение k для того, чтобы очередь не удлинялась.
11. Составить программу решения «задачи коммивояжера». Необходимо определить минимальную стоимость проезда коммивояжера по N городам с возвращением в исходную точку. Каждый город входит в маршрут только один раз. Предположить, что стоимость проезда из города i в город j такая же, как и из j в i .

Игры

12. *Шахматы*. Написать программу, которая генерирует или считывает шахматную позицию и определяет не находится ли один из королей под шахом и не является ли шах матом. Предусмотреть два варианта ввода исходных данных: а) шахматная позиция генерируется с помощью датчика случайных чисел; б) шахматная позиция вводится с клавиатуры.
13. *Кроссворд*. Написать программу составления кроссвордов. Исходными данными является конфигурация 6×6 (некоторое расположение пустых и заполненных квадратов) и список слов, состоящих из шести или менее букв. Результатом должно быть расположение этих слов, образующее общепринятый кроссворд или сообщение о том, что такая конфигурация невозможна.
14. *«Кости»*. Вводится число от 2 до 12 и ставка. Программа с помощью датчика случайных чисел дважды выбирает числа от 1 до 6 (бросание двух кубиков, на гранях которых должны быть числа от 1 до 6). Если сумма выпавших цифр меньше 7 и играющий задумал число меньше 7, он выигрывает ставку. Если сумма выпавших цифр больше 7 и играющий задумал число больше 7, он тоже выигрывает ставку. Если

угадал сумму цифр, то получает в 4 раза больше, чем ставка. Ставка проиграна, если ни одна из ситуаций не учтена. В начальный момент времени у играющего 100 очков. В программе должно присутствовать графическое изображение кубиков при каждом ходе игрока.

15. *Морской бой*. Поле 10x10 позиций. Стоят невидимые вражеские корабли: 4 по 1 клетке, 3 по 2 клетки, 2 по 3 клетки, 1 – 4 клетки. Необходимо поразить каждую из клеток кораблей. Два игрока вводят позиции кораблей в виде цифр (1,2,3,4) – соответствующие позиции матрицы, тем самым определяя конфигурацию положения кораблей. Игроки по очереди наносят удары по кораблям противника. Если позиция корабля указана верно, то она помечается крестиком на поле. Предусмотреть вариант игры, когда один из игроков – ЭВМ.
16. *«Сбей самолет»*. По экрану летят вражеские самолеты. Цель играющего сбить самолеты. Пусковая установка находится в нижней части экрана. Установку можно перемещать по строке вперед и назад.
17. *Обучение работе на клавиатуре*. Составить программу обучения работе с клавиатурой. Программа должна выдавать на экран буквы, цифры, слова и фразы, которые следует набрать на клавиатуре и оценивать правильность и скорость набора. В программе предусмотреть 3 уровня подготовленности обучающегося.
18. *«Скачки»*. В игре участвуют 10 наездников. За каждый тур игры каждый из них продвигается вперед на расстояние от 1 до 5 км случайным образом. Длина дистанции 50 км. Всего производится 5 заездов. Победителю каждого заезда начисляется 5 очков. Победитель - наездник, набравший максимальное количество очков во всех заездах. Перед началом заездов, участник игры выбирает номер наездника, с которым он будет идентифицироваться во время игры. В каждом туре с вероятностью 0,1 каждый наездник может упасть, т.е. передвинуться на 0 км. Передвижение наездников отображать графически на экране.

Строки и текстовые файлы.

19. Текст программы на Паскале хранится в файле на диске. Составить программу обработки текста программы: а) подсчитать какие ключевые слова Паскаля, и в каком количестве использованы в обрабатываемом тексте; б) составить перечень имен простых переменных; в) представить перечень меток программы в алфавитном порядке.
20. Текст программы на Паскале хранится в файле на диске. Составить программу обработки текста программы: а) определить максимальную степень вложенности циклов в программе; б) определить общее количество строк и символов, отличных от пробела; в) удалить из текста программы все комментарии.
21. Текст программы на Паскале хранится в файле на диске. Составить программу обработки текста программы: а) первые буквы служебных слов сделать заглавными; б) текст комментария заменить на номер

- комментария по порядку; в) переписать текст программы в новый файл с минимальным количеством пробелов (сохранить пробелы только там, где они необходимы).
22. Создать программу, анализирующую правильность записи арифметического выражения с точки зрения синтаксиса Паскаля. Арифметическое выражение задается строковой переменной и вводится с клавиатуры.
 23. Текст программы на Паскале хранится в файле на диске. Распечатать на экран текст программы таким образом, чтобы в каждой строке размещался только один оператор. Организовать смещение операторов относительно операторных скобок как это принято в Паскале.
 24. Составить программу, позволяющую сжимать текстовую информацию, а затем преобразовывать сжатую информацию в исходное состояние. В программе предусмотреть два варианта: а) для хранения текста в сжатом виде найти часто повторяющиеся последовательности из двух букв и заменить их кодом. В качестве кода использовать символы, не встречающиеся в тексте. Составить таблицу кодов; б) в заданном тексте найти слова, которые встречаются более 3 раз. Закодировать их и сжать текст, заменив слова кодами. Составить таблицу кодов.
 25. Разработать программу «выравнивания». Исходная информация – текст, записанный в текстовом файле. Программа выводит этот текст с выравниванием по краям. Параметры абзаца задаются в диалоговом режиме.
 26. Текст программы на Паскале хранится в файле на диске. Читать программу с диска и получить ее распечатку. В распечатке программы отметить операторы, изменяющие значения заданных переменных (их имена вводятся). Степень вложенности операторов цикла пометить добавлением слева соответствующего количества пробелов.
 27. Составить программу, определяющую наличие неописанных идентификаторов в тексте программы на Паскале. Текст программы хранится в файле на диске.

ГИС

28. Технические и программные средства ввода и вывода данных в ГИС.
29. Модели данных в ГИС. Основные типы координатных моделей.
30. ГИС-проекты (глобальные, международные, региональные, локальные).
31. История ГИС. Основные черты развития геоинформатики в России.
32. Алгоритмы сжатия пространственной информации.
33. Анаморфированные геоизображения.
34. Автоматизация ввода и ведения картографической и атрибутивных баз данных.
35. Мониторинг технических объектов средствами ГИС-технологий.
36. Создание экологической ГИС.
37. Цифровые модели местности. Логическая и физическая структура

ЦММ.

38. Дистанционное зондирование. Применение данных дистанционного зондирования в ГИС.
39. САПР и ГИС, их отличие и взаимодействие.
40. Применение экспертных систем в ГИС.
41. Особенности моделирования в ГИС.
42. Разработка собственной ГИС.

ХII. КОМПЛЕКТЫ ЭКЗАМЕНАЦИОННЫХ БИЛЕТОВ ПО ДИСЦИПЛИНЕ

Комплекты экзаменационных билетов составляются на основе перечня вопросов, приведенного в рабочей программе, по следующей форме.

ГОУВПО «Амурский государственный университет»	
Утверждено на заседании кафедры	Кафедра <i>математического анализа и моделирования</i>
«___» _____ 200 г.	
Заведующий кафедрой	Факультет <i>математики и информатики</i>
	Курс 4
Утверждаю: _____	Дисциплина <i>"Программное обеспечение вычислительных систем и сетей"</i>
Экзаменационный билет 1	
1. Метод пошаговой детализации и метод структурного программирования.	
2. Использование Database Desktop для создания новой таблицы.	
3. Обзор инструментальных средств разработки программ, выполняющихся на стороне клиента. Их назначение и возможности.	

ХIII. КАРТА ОБЕСПЕЧЕННОСТИ ДИСЦИПЛИНЫ КАДРАМИ ПРОФЕССОРСКО-ПРЕПОДАВАТЕЛЬСКОГО СОСТАВА

Дисциплину в полном объёме ведёт:

1. Фамилия, имя, отчество: Решетнёва Татьяна Геннадьевна
2. Учёное звание: доцент
3. Учёная степень: канд. геол.-мин. наук

СОДЕРЖАНИЕ

I. Рабочая программа дисциплины	3
1. Цели и задачи дисциплины, ее место в учебном процессе	3
1.1 Цель преподавания дисциплины	3
1.2 Задачи изучения дисциплины	3
1.3 Перечень дисциплин с указанием разделов (тем), усвоение которых студентами необходимо при изучении данной дисциплины	5
1.4 Структура дисциплины	5
2. Содержание дисциплины	5
2.1 Федеральный компонент	5
2.2 Наименование тем, их содержание, объем в лекционных часах	6
2.3 Практические и семинарские занятия, их содержание и объем в часах	6
2.4 Лабораторные занятия, их наименование и объем в часах	7
2.5 Самостоятельная работа студентов	7
2.6 Вопросы к зачету	8
2.7 Вопросы к экзамену	10
3. Учебно-методические материалы по дисциплине	13
3.1 Перечень обязательной (основной) литературы	13
3.2 Перечень дополнительной литературы	14
3.3 Перечень методических пособий	15
3.4 Перечень наглядных и иных пособий	15
3.5 Средства обеспечения освоения дисциплины	15
4. Материально-техническое обеспечение дисциплины	15
5. Критерии оценки знаний	15
5.1 Текущий контроль знаний	15
5.2 Требования к знаниям студентов, предъявляемые на зачете	16
5.3 Требования к знаниям студентов, предъявляемые на экзамене	16

II. График самостоятельной учебной работы студентов по дисциплине на каждый семестр с указанием ее содержания, объема в часах, сроков и форм контроля	17
III. Методические рекомендации по проведению практических семинаров, деловых игр, разбору ситуаций и т. п. список рекомендуемой литературы (основной и дополнительной)	17
IV. Краткий конспект лекций (по каждой теме) или план-конспект	18
V. Методические указания к практическим семинарским занятиям	49
VI. Практический семинар (перечень основных тем)	50
VII. Методические указания по применению современных информационных технологий для преподавания учебной дисциплины (в т. ч. разработанные ведущими преподавателями)	51
VIII. Методические указания профессорско-преподавательскому составу по организации межсессионного и экзаменационного контроля знаний студентов (материалы по контролю качества образования)	51
IX. Комплекты заданий для лабораторных работ, контрольных работ, домашних заданий	51
X. Фонд тестовых и контрольных заданий для оценки качества знаний по дисциплине	51
XI. Тематика расчетно-графических работ	61
XII. Комплекты экзаменационных билетов по дисциплине	66
XIII. Карта обеспеченности дисциплины кадрами профессорско-преподавательского состава	66
Содержание	67

