

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

Кафедра Информационных и управляющих систем
(наименование кафедры)

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ

Проектирование автоматизированных систем обработки информации и управления
(наименование дисциплины)

Основной образовательной программы по направлению подготовки (специальности):

230102.65 «Автоматизированные системы обработки информации и управления»
(код и наименование направления (специальности))

Благовещенск 2011

УМКД разработан _____
(степень, звание, фамилия, имя, отчество разработчиков)

Рассмотрен и рекомендован на заседании кафедры

Протокол заседания кафедры от « ____ » _____ 201__ г. № ____

Зав. кафедрой _____ / _____ /
(подпись) (И.О. Фамилия)

УТВЕРЖДЕН

Протокол заседания УМСС _____
(указывается название специальности (направления подготовки))

от « ____ » _____ 201__ г. № ____

Председатель УМСС _____ / _____ /
(подпись) (И.О.Фамилия)

1. Рабочая программа учебной дисциплины

РАБОЧАЯ ПРОГРАММА

ПРОЕКТИРОВАНИЕ АВТОМАТИЗИРОВАННЫХ СИСТЕМ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

Для специальности 230102.65 «Автоматизированные системы обработки информации и управления»

Квалификация (степень) выпускника – инженер

Специальное звание – нет

Курс – 5

Семестр – 9

Лекции – 45 (час.)

Экзамен – 9

Практические (семинарские) занятия – нет

Лабораторные занятия – 45 (час.)

Самостоятельная работа – 86 (час.)

Общая трудоемкость дисциплины – 176 (час.)

Курсовая работа (проект) – 9 сем.

Составитель – А.В.Бушманов, доцент, канд. техн. наук

Факультет математики и информатики

Кафедра информационных и управляющих систем

Рабочая программа составлена на основании Государственного образовательного стандарта высшего профессионального образования по направлению подготовки 654600 – Информатика и вычислительная техника

Рабочая программа обсуждена на заседании кафедры информационных и управляющих систем

«__» _____ 20__ г., протокол № _____

Заведующий кафедрой _____ А.В. Бушманов

Рабочая программа одобрена на заседании учебно-методического совета специальности 230102.65 «Автоматизированные системы обработки информации и управления»

«__» _____ 20__ г., протокол № _____

Председатель _____ В.В. Еремина

Рабочая программа переутверждена на заседании кафедры информационных и управляющих систем

«__» _____ 20__ г., протокол № _____

Заведующий кафедрой _____ А.В. Бушманов

СОГЛАСОВАНО

Учебно-методическое
управление

«__» _____ 20__ г.

СОГЛАСОВАНО

Председатель учебно-методического
совета факультета

_____ С.Г. Самохвалова

«__» _____ 20__ г.

СОГЛАСОВАНО

Заведующий выпускающей кафедрой

_____ А.В. Бушманов

«__» _____ 20__ г.

СОГЛАСОВАНО

Директор научной библиотеки

_____ Л.А. Проказина

«__» _____ 20__ г.

1 ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель дисциплины: Цель преподавания дисциплины заключается в том, чтобы на основе предшествующих курсов учебного плана дать студентам завершающие знания в области современных научных и практических методов и моделей управления сложными автоматизированными, информационными и организационно-административными системами. Дать знания по проектированию и функционированию систем АСОИУ.

Основные разделы дисциплины определенные в Государственном образовательном стандарте высшего профессионального образования: Общая характеристика процесса проектирования АСОИУ; структура информационно-логической модели АСОИУ, разработка функциональной модели; исходные данные для проектирования; разработка модели и защита данных; разработка пользовательского интерфейса; разработка проекта распределенной обработки; структура программных модулей; разработка алгоритмов; логический анализ структур АСОИУ; анализ и оценка производительности АСОИУ; управление проектом АСОИУ; проектная документация; инструментальные средства проектирования АСОИУ; типизация проектных решений; графические средства представления проектных решений.

Задачи дисциплины:

Изучая курс «Проектирование Автоматизированные системы обработки информации и управления», студенты должны получить представление о системах обработки информации и управления, о проектировании систем, получить навыки разработки программ, выяснить какие методики при этом используются. С целью приобретения навыков в решении основных вопросов проектирования АСОИУ, при их качественной формулировке при изучении дисциплины, необходимы практические шаги в виде курсового проектирования.

По окончании изучения курса студенты должны уметь проектировать и эксплуатировать АСОИУ в самых различных сферах человеческой деятельности, владеть соответствующими навыками.

2 МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВПО

Дисциплина относится к федеральному компоненту цикла специальных дисциплин (СД.10), Государственного образовательного стандарта высшего профессионального образования по специальности 230102 «Автоматизированные системы обработки информации и управления» (квалификация (степень) «инженер»).

Для успешного освоения данной дисциплины необходимы знания, умения и навыки, приобретенные в результате освоения дисциплин базовой части математического и естественнонаучного цикла (ЕН) и общепрофессиональных дисциплин (ОПД) Государственного образовательного стандарта высшего профессионального образования по специальности 230102 «АСОИУ» (квалификация (степень) «инженер»): математический анализ (ЕН.Ф.01.02); теория вероятностей, математическая статистика и случайные процессы (ЕН.Ф.01.06); вычислительная математика (ЕН.Ф.01.05); алгоритмические языки и программирование (ОПД.Ф.05), основы теории управления (ОПД.Ф.06); базы данных (ОПД.Ф.09); сети ЭВМ и телекоммуникации (ОПД.Ф.10).

3 ТРЕБОВАНИЯ К УРОВНЮ ОСВОЕНИЯ СОДЕРЖАНИЯ ДИСЦИПЛИНЫ

В результате освоения дисциплины обучающийся должен демонстрировать следующие результаты образования:

- знать: принцип, модели, средства описания информационных систем и их элементов, объектно-ориентированные модели предметных областей, средства спецификации функциональных задач и проектных решений; современные методы и средства разработки АСОИУ; принципы, модели и методы управления информационными системами, тенденции их развития, связь со смежными областями;

- уметь: применять базовые методики их исследования, выполнять сравнительный анализ полученных результатов.
- владеть: методами и средствами проектирования и комплексирования аппаратных и программных средств АСОИУ; современными методами организации разработки АСОИУ и их программного обеспечения; методами оценки качества программного обеспечения, надежности и качества информационных систем, сертификации и аттестации АСОИУ и их компонентов.

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 176 часов.

Общая характеристика процесса проектирования АСОИУ; структура информационно-логической модели АСОИУ, разработка функциональной модели; исходные данные для проектирования; разработка модели и защита данных; разработка пользовательского интерфейса; разработка проекта распределенной обработки; структура программных модулей; разработка алгоритмов; логический анализ структур АСОИУ; анализ и оценка производительности АСОИУ; управление проектом АСОИУ; проектная документация; инструментальные средства проектирования АСОИУ; типизация проектных решений; графические средства представления проектных решений.

№ п/п	Раздел дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость в часах				Формы текущего контроля успе- ваемости Форма промежу- точной аттеста- ции
				Лек	Пр	Лаб	Сам	
1	Общая характеристика процесса проектирования АСОИУ	9	1	3	-	3	5	Защита лаб. ра- боты
2	Структура информаци- онно-логической модели АСОИУ, разработка функциональной модели	9	2	3	-	3	5	Защита лаб. ра- боты
3	Исходные данные для проектирования	9	3	3	-	3	5	Защита лаб. ра- боты
4	Разработка модели и за- щита данных	9	4	3	-	3	5	Защита лаб. ра- боты
5	Разработка пользователь- ского интерфейса	9	5	3	-	3	5	Защита лаб. ра- боты
6	Разработка проекта рас- пределенной обработки	9	6	3	-	3	5	Защита лаб. ра- боты

7	Структура программных модулей	9	7	3	-	3	6	Защита лаб. работы
8	Разработка алгоритмов	9	8	3	-	3	5	Защита лаб. работы
9	Логический анализ структур АСОИУ	9	9	3	-	3	5	Защита лаб. работы
10	Анализ и оценка производительности АСОИУ	9	10	3	-	3	6	Защита лаб. работы
11	Управление проектом АСОИУ	9	11	3	-	3	6	Защита лаб. работы
12	Проектная документация	9	12	3	-	3	5	Защита лаб. работы
13	Инструментальные средства проектирования АСОИУ	9	13	3	-	3	6	Защита лаб. работы
14	Типизация проектных решений	9	14	3	-	3	5	Защита лаб. работы
15	Графические средства представления проектных решений	9	15	3	-	3	6	Защита лаб. работы
16			1-15	45	-	45	86	экзамен

5 СОДЕРЖАНИЕ РАЗДЕЛОВ И ТЕМ ДИСЦИПЛИНЫ

5.1 Лекции

5.1.1 Общая характеристика процесса проектирования АСОИУ

Методологическая основа проектирования. Информационная система. Система управления. Система как объект проектирования АСОИУ. АСУ предприятием. АСУ технологическим процессом. Системы автоматизированного проектирования.

5.1.2 Основные понятия теории проектирования АСОИУ.

Состав и структура АСОИУ. Объект проектирования. Субъект проектирования. Технология проектирования. Анализ и синтез в проектировании систем. Понятие декомпозиции систем. Понятие жизненного цикла. Системный анализ. Системный синтез. Формализация технологии проектирования.

5.1.3 Средства структурного анализа.

Диаграммы потоков данных. Диаграммы «Сущность-Связь», Диаграммы переходов состояний. Основные, вспомогательные и организационные процессы. Международный стандарт ISO/IES 12207. Характеристики основных процессов. Вспомогательные процессы для поддержки выполнения основных процессов, обеспечения качества проекта, организации верификации, проверки и тестирования ПО.

5.1.4 Дополнительные группы процессов ЖЦ по ИС (международный стандарт ISO/IEC 15288).
Договорные процессы. Процессы предприятия. Проектные процессы. Технические процессы. Специальные процессы.

5.1.5 Каноническое проектирование. Стадии и этапы проектирования. Модель «AS-IS». Модель «TO-BE».
Формирование требований. Разработка концепции ИС. Техническое задание. Эскизный проект. Технический проект. Рабочая документация. Ввод в действие. Сопровождение.

5.1.6 Состав и содержание ТЗ.
Задачи решаемые при разработке ТЗ. Требования к составу и содержанию ТЗ.

5.1.7 Понятие технического проекта. Состав технического проекта.
Комплекс научно-исследовательских и экспериментальных работ для выбора основных проектных решений и расчет экономической эффективности системы.

5.1.8 Типовое проектирование ИС. Типовые проектные решения.
Параметрически-ориентированное проектирование. Модельно-ориентированное проектирование. Репозиторий.

5.1.9 Классификация структурных методологий, их характеристика.
Технические структурные карты Константайна и Джексона. Диаграммы потоков данных в нотации Йодана/Де Марко.

5.1.10 Методология SADT.
Концепции методологии. Построение SADT-модели. Декомпозиция модели. Типы связей между функциями.

5.1.11 Методология структурного системного анализа Гейна-Сарсона.
Графические иерархические спецификации описывающие систему с позиции потоков данных. Словари данных. Миниспецификации обработки данных.

5.1.12 Спиральная модель этапов проектирования АСОИУ.
Взаимодействие интерфейсов программных модулей между собой и с базой данных. Реализация алгоритмов. Прототипные технологии (RAD-технологии).

5.1.13 Методология функционального моделирования.
Стандарт IDEF0. Методология IDEF0. Компоненты синтаксиса IDEF0.

5.1.14 Методология потоков данных DFD.
Диаграммы уровней иерархии. Основные компоненты диаграмм потоков данных. Правила детализации.

5.1.15 Методология IDEF3.
Логика взаимодействия информационных потоков. Диаграммы Workflow. Единицы работы – Unit of Work (UOW). Связи в IDEF3. Перекрестки для слияния и разветвления стрелок.

5.1.16 Методология разработки информационной базы IDEF1X.
Сущность в методологии IDEF1X. Связи. Атрибуты.

5.1.17 Методология UML.
Диаграммы вариантов использования. Диаграммы классов. Диаграммы последовательностей. Диаграммы состояний. Диаграммы кооперации. Диаграммы деятельности. Диаграммы компонентов.

5.1.18 Эскизный проект. Рабочий проект.
Содержание эскизного проекта. Состав рабочей документации.

5.1.19 Принципы построения функциональной структуры АСОИУ.
Состав функциональных подсистем. Принципы построения функциональных подсистем. Проблемный принцип формирования подсистем. Предметно-функциональный подход.

5.1.20 Обеспечивающие подсистемы информационных систем.
Подсистема «Организационное обеспечение». Подсистема «Правовое обеспечение». Подсистема «Техническое обеспечение». Подсистема «Математическое обеспечение».

Подсистема «Программное обеспечение». Подсистема «Информационное обеспечение». Подсистема «Лингвистическое обеспечение». Подсистема «Технологическое обеспечение».

5.1.21 Проектирование информационного обеспечения АСОИУ.

Принципы проектирования документооборота. Система документации. Требования к унифицированной системе документооборота. Проектирование систем входных и выходных документов.

5.1.22 Внемашиное информационное обеспечение.

Основные понятия классификации информации. Система классификации и кодирования информации. Понятия и основные требования к системе кодирования информации. Состав и содержание операций проектирования классификаторов.

5.1.23 Внутримашинное информационное обеспечение.

Проектирование форм электронных документов. Информационная база. Требования к организации хранения файлов в информационной базе. Интегрированная информационная база.

5.1.24 Ввод в эксплуатацию АСОИУ.

Подготовка объекта к вводу. Предварительные испытания. Опытная эксплуатация. Приемочные испытания. Сопровождение АСОИУ.

5.2 Лабораторные занятия

5.2.1 Лабораторная работа 1. Создание диаграммы DFD.

5.2.2 Лабораторная работа 2-3. Решение поставленной задачи при помощи технологии IDEF0.

5.2.3 Лабораторная работа 4. Создание диаграммы IDEF3.

5.2.4 Лабораторная работа 5-6. Взаимосвязь моделей IDEF0 и IDEF3.

5.2.5 Лабораторная работа 7-8. Два подхода к построению DFD-моделей.

5.2.6 Лабораторная работа 9-10. Создание модели TO-BE.

5.2.7 Лабораторная работа 11-12. Моделирование управленческого учета на предприятии.

5.2.8 Лабораторная работа 13-14. Введение в CASE-пакет Rational Rose.

5.2.9 Лабораторная работа 15. Диаграммы вариантов использования.

5.2.9 Лабораторная работа 16. Диаграммы классов.

5.2.10 Лабораторная работа 17. Диаграммы взаимодействия.

5.2.11 Лабораторная работа 18. Диаграммы состояний.

5.2.12 Лабораторная работа 19. Диаграммы пакетов, компонентов, размещения.

5.2.13 Лабораторная работа 20-21. Генерация исходных текстов программ.

5.2.14 Лабораторная работа 22. Обратное проектирование.

6 САМОСТОЯТЕЛЬНАЯ РАБОТА

№ п/п	Раздел дисциплины	Форма (вид) самостоятельной работы	Трудоемкость в часах
1	Программное обеспечение IDEF-моделирования.	Выполнение лабораторной работы, оформление отчета.	6
2	Методология функционального моделирования	Выполнение двух лабораторных работ, оформление отчетов.	10

3	Синтаксис и семантика моделей IDEF3.	Выполнение двух лабораторных работ, оформление отчетов.	10
4	Основные этапы проведения проектирования в Rational Rose.	Выполнение двух лабораторных работ, оформление отчетов.	10
5	Изучение средств обратного проектирования Rational Rose.	Выполнение двух лабораторных работ, оформление отчетов, подготовка к сдаче зачета.	10
6	Курсовой проект	Выполнение курсового проекта по индивидуальному заданию.	40
Итого:			86

7 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Образовательный процесс по дисциплине строится на основе комбинации следующих образовательных технологий.

Интегральную модель образовательного процесса по дисциплине формируют технологии методологического уровня: модульно-рейтинговое обучение, технология поэтапного формирования умственных действий, технология развивающего обучения, элементы технологии развития критического мышления.

Реализация данной модели предполагает использование следующих технологий стратегического уровня (задающих организационные формы взаимодействия субъектов образовательного процесса), осуществляемых с использованием определенных тактических процедур:

- лекционные (вводная лекция, информационная лекция, обзорная лекция, лекция-консультация, проблемная лекция);
- лабораторные (углубление знаний, полученных на теоретических занятиях, решение задач);
- тренинговые (формирование определенных умений и навыков, формирование алгоритмического мышления);
- активизации познавательной деятельности (приемы технологии развития критического мышления через чтение и письмо, работа с литературой, подготовка презентаций по темам домашних работ);
- самоуправления (самостоятельная работа студентов, самостоятельное изучение материала).

Информационные технологии используются при организации коммуникации со студентами для представления информации, выдачи рекомендаций и консультирования по оперативным вопросам (электронная почта), использование мультимедиа-средств при проведении лекционных и практических занятий.

В качестве образовательных технологий при изучении дисциплины используются мультимедийные лекции, на лабораторных занятиях используются современные пакеты программных продуктов. С целью текущего контроля знаний студентов на лабораторных работах проводится контроль выполнения работы. Студентам предлагается обсудить полученные результаты и высказать свое мнение по применению возможных приемов для улучшения показателей либо результатов работы.

№ п/п	Раздел дисциплины	Форма (вид) образовательных технологий	Количество часов
1	Общая характеристика процесса проектирования АСОИУ.	Мультимедийная лекция	3
		Лабораторная работа	3
2	Структура информационно-логической модели АСОИУ, разработка функциональной модели	Мультимедийная лекция	3
		Лабораторная работа	3
3	Исходные данные для проектирования	Мультимедийная лекция	3
		Лабораторная работа	3
4	Разработка модели и защита данных	Мультимедийная лекция	3
		Лабораторная работа	3
5	Разработка пользовательского интерфейса	Мультимедийная лекция	3
		Лабораторная работа	3
6	Разработка проекта распределенной обработки	Мультимедийная лекция	3
		Лабораторная работа	3
7	Структура программных модулей	Мультимедийная лекция	3
		Лабораторная работа	3
8	Разработка алгоритмов	Мультимедийная лекция	3
		Лабораторная работа	3
9	Логический анализ структур АСОИУ	Мультимедийная лекция	3
		Лабораторная работа	3
10	Анализ и оценка производительности АСОИУ	Мультимедийная лекция	3
		Лабораторная работа	3
11	Управление проектом АСОИУ	Мультимедийная лекция	3
		Лабораторная работа	3
12	Проектная документация	Мультимедийная лекция	3
		Лабораторная работа	3
13	Инструментальные средства проек-	Мультимедийная лекция	3

	тирования АСОИУ	Лабораторная работа	3
14	Типизация проектных решений	Мультимедийная лекция	3
		Лабораторная работа	3
15	Графические средства представления проектных решений	Мультимедийная лекция	3
		Лабораторная работа	3
16	Всего по разделам		90

8 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

8.1 Оценочные средства для текущего контроля успеваемости

8.1.1 Контрольные вопросы допуска к выполнению лабораторных работ

8.1.2 Отчеты о выполнении индивидуальных вариантов заданий лабораторных работ

8.2 Оценочные средства для промежуточной аттестации

Вопросы к экзамену:

8.2.1 Система как объект проектирования АСОИУ. Системы автоматизированного проектирования.

8.2.2 Состав и структура АСОИУ. Объект проектирования. Субъект проектирования. Технология проектирования.

8.2.3 Диаграммы потоков данных. Диаграммы «Сущность-Связь», Диаграммы переходов состояний.

8.2.4 Основные, вспомогательные и организационные процессы. Международный стандарт ISO/IES 12207. Характеристики основных процессов.

8.2.5 Дополнительные группы процессов ЖЦ по ИС (международный стандарт ISO/IEC 15288). Договорные процессы. Процессы предприятия. Проектные процессы. Технические процессы. Специальные процессы.

8.2.6 Каноническое проектирование. Стадии и этапы проектирования. Модель «AS-IS». Модель «TO-BE».

8.2.7 Состав и содержание ТЗ. Задачи решаемые при разработке ТЗ. Требования к составу и содержанию ТЗ.

8.2.8 Понятие технического проекта. Состав технического проекта.

8.2.9 Комплекс научно-исследовательских и экспериментальных работ для выбора основных проектных решений и расчет экономической эффективности системы.

8.2.10 Типовое проектирование ИС. Типовые проектные решения.

8.2.11 Параметрически-ориентированное проектирование.

8.2.12 Модельно-ориентированное проектирование. Репозиторий.

8.2.13 Классификация структурных методологий, их характеристика.

8.2.14 Методология SADT. Концепции методологии. Построение SADT-модели. Декомпозиция модели. Типы связей между функциями.

8.2.15 Методология структурного системного анализа Гейна-Сарсона. Графические иерархические спецификации описывающие систему с позиции потоков данных. Словари данных. Миниспецификации обработки данных.

8.2.16 Спиральная модель этапов проектирования АСОИУ.

- 8.2.17 Взаимодействие интерфейсов программных модулей между собой и с базой данных. Реализация алгоритмов.
- 8.2.18 Прототипные технологии (RAD-технологии).
- 8.2.19 Методология функционального моделирования. Стандарт IDEF0. Методология IDEF0. Компоненты синтаксиса IDEF0.
- 8.2.20 Методология потоков данных DFD. Диаграммы уровней иерархии. Основные компоненты диаграмм потоков данных. Правила детализации.
- 8.2.21 Методология IDEF3. Логика взаимодействия информационных потоков. Диаграммы Workflow. Единицы работы – Unit of Work (UOW).
- 8.2.22 Связи в IDEF3. Перекрестки для слияния и разветвления стрелок.
- 8.2.23 Методология разработки информационной базы IDEF1X.
- 8.2.24 Сущность в методологии IDEF1X. Связи. Атрибуты.
- 8.2.25 Методология UML. Диаграммы вариантов использования. Диаграммы классов.
- 8.2.26 Диаграммы последовательностей. Диаграммы состояний. Диаграммы кооперации. Диаграммы деятельности. Диаграммы компонентов.
- 8.2.27 Эскизный проект. Рабочий проект. Содержание эскизного проекта. Состав рабочей документации.
- 8.2.28 Принципы построения функциональной структуры АСОИУ. Состав функциональных подсистем. Принципы построения функциональных подсистем.
- 8.2.29 Проблемный принцип формирования подсистем. Предметно-функциональный подход.
- 8.2.30 Обеспечивающие подсистемы информационных систем.
- 8.2.31 Проектирование информационного обеспечения АСОИУ. Принципы проектирования документооборота. Система документации. Требования к унифицированной системе документооборота. Проектирование систем входных и выходных документов.
- 8.2.32 Внемашинное информационное обеспечение. Основные понятия классификации информации. Система классификации и кодирования информации.
- 8.2.33 Понятия и основные требования к системе кодирования информации. Состав и содержание операций проектирования классификаторов.
- 8.2.34 Внутримашинное информационное обеспечение. Проектирование форм электронных документов. Информационная база.
- 8.2.35 Требования к организации хранения файлов в информационной базе. Интегрированная информационная база.
- 8.2.36 Ввод в эксплуатацию АСОИУ. Подготовка объекта к вводу. Предварительные испытания. Опытная эксплуатация. Приемочные испытания. Сопровождение АСОИУ.
- 8.3 Учебно-методическое обеспечение самостоятельной работы
- 8.3.1 Карточки с заданиями и методическими указаниями по выполнению лабораторных работ
- 8.3.2 СТО СМК 4.2.3.05-2011. Стандарт ФГБОУВПО «АмГУ». Оформление выпускных квалификационных и курсовых работ (проектов), 2011. – 95 с.

9 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

а) основная литература:

- 10.1 Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
- 10.2 Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

б) дополнительная литература:

- 10.3 Гвоздева Т.В. Проектирование информационных систем: учеб. пособие: рек. УМО/ Т.В. Гвоздева, Б.А. Баллод. -Ростов н/Д: Феникс, 2009.-509 с.
- 10.4 Бройдо В.Л. Вычислительные системы, сети и телекоммуникации: учеб. пособие: рек. Мин. обр. РФ/ В. Л. Бройдо, О. П. Ильина. -3-е изд.. -СПб.: Питер, 2008.-766 с.
- 10.5 Фуфаев Э.В. Компьютерные технологии в приборостроении: учеб.: рек. УМО/ Э. В. Фуфаев, Л. И. Фуфаева. -М.: Академия, 2009.-334 с.
- 10.6 Бройдо В.Л. Архитектура ЭВМ и систем: учеб.: рек. Мин. обр. РФ/ В. Л. Бройдо, О. П. Ильина. -2-е изд.. -СПб.: Питер, 2009.-720 с.
- 10.7 Управление проектами: учеб.пособие: рек. Мин. обр. РФ/ И.И Мазур [и др.]; под общ. ред. И. И. Мазура, В. Д. Шапиро. -5-е изд., перераб. -М.: Омега-Л, 2009.-960 с.
- 10.8 Клейменов С.А. Администрирование в информационных системах : учеб. пособие: рек. УМО/ С.А.Клейменов, В.П.Мельников, А.М.Петраков. -М.: Академия, 2008.-272 с.
- 10.9 Хетагуров Я.А. Проектирование автоматизированных систем обработки информации и управления (АСОИУ) : учеб.: доп. Мин. обр. РФ/ Я. А. Хетагуров. -М.: Высш. шк., 2006.-224 с.
- в) периодические издания:
- 10.10 Проблемы теории и практики управления
- 10.11 Информатика и системы управления
- 10.12 Информационные системы и технологии
- г) программное обеспечение и Интернет-ресурсы:

Свободно распространяемое программное обеспечение

№ п/п	Наименование ресурса	Характеристика
1	http://www.iqlib.ru	Интернет библиотека образовательных изданий, в которой собраны электронные учебники, справочные и учебные пособия. Удобный поиск по ключевым словам, отдельным темам и отраслям знаний.
2	http://www.intuit.ru	Интернет-университет информационных технологий, в котором собраны электронные и видео-курсы по отраслям знаний
3	http://amursu.ru	Сайт АмГУ, Библиотека – электронная библиотека АмГУ
4	http://www.biblioclub.ru	Электронная библиотечная система «Университетская библиотека – online»: специализируется на учебных материалах для ВУЗов по научно-гуманитарной тематике, а так же содержит материалы по точным и естественным наукам

10 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

- 10.1 Лекционная аудитория, оборудованная мультимедийными средствами.
 10.2 Лаборатории, оборудованные рабочими местами пользователей ЭВМ.

11 РЕЙТИНГОВАЯ ОЦЕНКА ЗНАНИЙ СТУДЕНТОВ ПО ДИСЦИПЛИНЕ

Семестровый модуль дисциплины						
№ п/п	Раздел дисциплины	Виды контроля	выполнения (недели)	Максимальное кол-во баллов	Посещение, активность на занятиях	Максимальное кол-во баллов за модуль
1	Общая характеристика процесса проектирования АСОИУ	ЛР № 1	1	5	1	4
2	Структура информационно-логической модели АСОИУ, разработка функциональной модели	ЛР № 2	2	5	1	5
		ЛР № 3		5	1	
3	Исходные данные для проектирования	ЛР № 4	3	5	1	4
4	Разработка модели и защита данных	ЛР № 5	4	5	1	5
		ЛР № 6		5	1	
5	Разработка пользовательского интерфейса	ЛР № 7	5	5	1	4
6	Разработка проекта распределенной обработки	ЛР № 8 ЛР № 9	6			3
7	Структура программных модулей	ЛР № 10	7			4
8	Разработка алгоритмов	ЛР № 11 ЛР № 12	8			4
9	Логический анализ структур АСОИУ	ЛР № 13	9			4
10	Анализ и оценка производительности АСОИУ	ЛР № 14 ЛР № 15	10			4

11	Управление проектом АСО-ИУ	ЛР № 16	11			4
12	Проектная документация	ЛР № 17 ЛР № 18	12			3
13	Инструментальные средства проектирования АСОИУ	ЛР № 19	13			4
14	Типизация проектных решений	ЛР № 20 ЛР № 21	14			4
15	Графические средства представления проектных решений	ЛР № 22	15			4
16	Экзамен					40
Итого						100

2 Краткое изложение программного материала

Лекция 1. Общая характеристика процесса проектирования АСОИУ.

Методологическая основа проектирования.

Под проектом ИС понимается проектно-конструкторская и технологическая документация, в которой представлено описание проектных решений по созданию и эксплуатации ИС в конкретной программно-технической среде.

Под проектированием ИС понимается процесс преобразования входной информации об объекте проектирования, о методах проектирования и об опыте проектирования объектов аналогичного назначения в проект ИС. С этой точки зрения проектирование ИС сводится к последовательной формализации проектных решений на различных стадиях жизненного цикла ИС: планирования и анализа требований, технического и рабочего проектирования, внедрения и эксплуатации ИС.

Осуществление проектирования ИС предполагает использование определенной технологии проектирования, соответствующей масштабу и особенностям разрабатываемого проекта.

Информационная система.

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

Процесс управления системой как направленное воздействие на элементы системы для достижения цели можно представить в виде информационного процесса, связывающего внешнюю среду, объект и систему управления. При этом внешняя среда и объект управления информируют систему управления о своем состоянии, система управления анализирует эту информацию, вырабатывает управляющее воздействие на объект управления, отвечает на возмущения внешней среды и при необходимости модифицирует цель и структуру всей системы.

В информационной системе объект управления представляет собой подсистему элементов производственной деятельности и хозяйственных процессов.

Система управления.

Система управления – это совокупность взаимодействующих структурных подразделений информационной системы, осуществляющих следующие функции управления:

- *планирование* – функция, определяющая цель функционирования информационной системы на различные периоды времени;
- *учет* – функция, отображающая состояние объекта управления в результате выполнения хозяйственных процессов;
- *контроль* – функция, с помощью которой определяется отклонение учетных данных от плановых целей и нормативов;
- *оперативное управление* – функция, осуществляющая регулирование всех хозяйственных процессов с целью исключения возникающих отклонений в плановых и учетных данных;
- *анализ* – функция, определяющая тенденции в работе информационной системы и резервы, которые учитываются при планировании на следующий временной период.

Система как объект проектирования АСОИУ.

Объектом проектирования является любой объект, для которого необходимо разработать автоматизированную информационную систему. Под системой понимается совокупность элементов, объединенных для выполнения определенных функций или задач. Под совокупностью элементов можно понимать следующее:

- понятия и способы отображения, хранения и передачи информации;
- алгоритмические и программные составляющие системы;
- математическое обеспечение, реализуемое в алгоритмах и программах (алгоритмы поиска, сортировки и т.д.);
- аппаратная часть (компьютеры, линии передачи данных, датчики, регистраторы первичной информации и т.д.);
- системы и приборы отображения и представления информации;
- производственная среда (обязательный компонент);
- человек, оператор (обязательный компонент).

Последнее время стало популярно определение компьютерных систем как систем, представляющих собой мощный инструментарий для сбора, хранения и обработки информации.

АСУ предприятием.

Автоматизированная система управления или АСУ - комплекс аппаратных и программных средств, предназначенный для управления различными процессами в рамках технологического процесса, производства, предприятия. АСУ применяются в различных отраслях промышленности, энергетике, транспорте и т. п. Термин автоматизированная, в отличие от термина автоматическая подчеркивает сохранение за человеком-оператором некоторых функций, либо наиболее общего, целеполагающего характера, либо не поддающихся автоматизации.

Автоматизированная система управления предприятием (АСУП) - комплекс программных, технических, информационных, лингвистических, организационно-технологических средств и действий квалифицированного персонала, предназначенный для решения задач планирования и управления различными видами деятельности предприятия, частный случай автоматизированной системы управления (АСУ).

АСУП - автоматизированная система управления предприятием или информационная система управления предприятием, то, что сейчас называется по-английски ERP.

Внедрение автоматизированной системы управления предприятием, как и любое серьезное преобразование на предприятии, является сложным и зачастую болезненным процессом. Тем не менее, некоторые проблемы, возникающие при внедрении системы, достаточно хорошо изучены, формализованы и имеют эффективные методологии решения. Заблаговременное изучение этих проблем и подготовка к ним значительно облегчают процесс внедрения и повышают эффективность дальнейшего использования системы.

АСУ технологическим процессом.

Автоматизированные системы управления технологическими процессами (АСУ ТП) - это комплекс программных и технических средств, предназначенных для создания систем

автоматизации управления технологическим оборудованием и производственными процессами на предприятиях (автоматизация производства). АСУ ТП – комплексное решение, обеспечивающее автоматизацию основных технологических операций на производстве в целом или каком-то его участке, выпускающем относительно завершенный продукт (промышленная автоматизация). АСУ ТП может состоять из отдельных систем автоматического управления (САУ) и комплексных устройств, объединенных единым решением для автоматизации технологических процессов с целью обеспечения максимальной эффективности решения производственных задач.

Обычно структура АСУ ТП представлена единой системой операторского управления технологическим процессом, куда входят один или нескольких пультов управления; средства сбора, передачи, обработки и архивирования информации о ходе производственного процесса; типовое оборудование: датчики, контроллеры и другие средства автоматизации. Для информационной связи всех подсистем используются промышленные сети. Режим и качество технологических процессов, состояние механизмов и машин контролируется средствами автоматизации, осуществляется постоянная диагностика АСУ ТП.

Разработка и внедрение систем АСУ ТП состоят из цепи взаимосвязанных процессов, включающих в себя проектирование АСУ ТП, программное обеспечение, программирование контроллеров, диагностирование АСУ ТП, диспетчеризацию.

Системы автоматизированного проектирования.

Автоматизация проектирования занимает особое место среди информационных технологий.

Во первых, автоматизация проектирования - синтетическая дисциплина, ее составными частями являются многие другие современные информационные технологии. Так, техническое обеспечение систем автоматизированного проектирования (САПР) основано на использовании вычислительных сетей и телекоммуникационных технологий, в САПР используются персональные компьютеры и рабочие станции. Математическое обеспечение САПР отличается богатством и разнообразием используемых методов вычислительной математики, статистики, математического программирования, дискретной математики, искусственного интеллекта. Программные комплексы САПР относятся к числу наиболее сложных современных программных систем, основанных на операционных системах Unix, Windows 95/NT, языках программирования. С, С++, Java и других, современных CASE технологиях, реляционных и объектно-ориентированных системах управления базами данных (СУБД), стандартах открытых систем и обмена данными в компьютерных средах.

Во вторых, знание основ автоматизации проектирования и умение работать со средствами САПР требуется практически любому инженеру разработчику. Компьютерами насыщены проектные подразделения, конструкторские бюро и офисы. Работа конструктора за обычным кульманом, расчеты с помощью логарифмической линейки или оформление отчета на пишущей машинке стали анахронизмом. Предприятия, ведущие разработки без САПР или лишь с малой степенью их использования, оказываются неконкурентоспособными как из-за больших материальных и временных затрат на проектирование, так и из-за невысокого качества проектов. Появление первых программ для автоматизации проектирования за рубежом и в СССР относится к началу 60-х гг. Тогда были созданы программы для решения задач строительной механики, анализа электронных схем, проектирования печатных плат.

Дальнейшее развитие САПР шло по пути создания аппаратных и программных средств машинной графики, повышения вычислительной эффективности программ моделирования и анализа, расширения областей применения САПР, упрощения пользовательского интерфейса, внедрения в САПР элементов искусственного интеллекта.

К настоящему времени создано большое число программно-методических комплексов для САПР с различной степенью специализации и прикладной ориентацией. В результате автоматизация проектирования стала необходимой составной частью подготовки инженеров разных специальностей; инженер, не владеющий знаниями и не умеющий работать в САПР, не может считаться полноценным специалистом.

Подготовка инженеров разных специальностей в области САПР включает базовую и специальную компоненты. Наиболее общие положения, модели и методики автоматизированного проектирования входят в программу курса, посвященного основам САПР, более детальное изучение тех методов и программ, которые специфичны для конкретных специальностей, предусматривается в профильных дисциплинах.

Ключевые вопросы:

1. Методология проектирования АСОИУ.
2. Понятие информационной системы.
3. Основные функции системы управления.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 2. Основные понятия теории проектирования АСОИУ.

Состав и структура АСОИУ.

Состав стадии и этапов работ для АС создания всех видов регламентируется ГОСТ 24.601-86 (Единая система стандартов автоматизированных систем управления. Автоматизированные системы. Стадии создания). Общие требования и совокупность работ на всех стадиях и этапах регламентируется ГОСТ 24.602-86 (состав и содержание работ по стадии создания). Конкретное содержание работ зависит от вида объекта управления и соответственно от технологии автоматизации объекта того или иного вида.

Объект проектирования.

Объектами проектирования ИС являются отдельные элементы или их комплексы функциональных и обеспечивающих частей. Так, функциональными элементами в соответствии с традиционной декомпозицией выступают задачи, комплексы задач и функции управления. В составе обеспечивающей части ИС объектами проектирования служат элементы и их комплексы информационного, программного и технического обеспечения системы.

Субъект проектирования.

В качестве субъекта проектирования ИС выступают коллективы специалистов, которые осуществляют проектную деятельность, как правило, в составе специализированной организации, и организация-заказчик, для которой необходимо разработать ИС.

Технология проектирования.

Технология проектирования ИС – это совокупность методологии и средств проектирования ИС, а также методов и средств организации проектирования.

Технология проектирования задается регламентированной последовательностью технологических операций, выполняемых в процессе создания проекта на основе того или иного метода, в результате чего стало бы ясно, не только что должно быть сделано для создания проекта, но и как, кому и в какой последовательности это должно быть сделано.

Анализ и синтез в проектировании систем.

Для изучения систем и использования этих знаний для создания и управления системами необходимо *системное мышление*, заключающееся в сочетании аналитического и синтетического образов мышления. Суть анализа состоит в разделении целого на части, в представлении сложного в виде совокупности более простых компонент. Но чтобы познать целое, сложное, необходим и обратный процесс – синтез. Необходимость сочетания этих видов познания вытекает из свойства эмерджентности систем: целостность системы нарушается при анализе, при расчленении системы утрачиваются не только существенные свойства самой системы, но и свойства ее частей, оказавшихся отделенными от нее. Результатом анализа является лишь вскрытие состава компонент, знание о том, как система работает, но не понимание того, почему и зачем она это делает. Синтетическое мышление объясняет поведение сис-

темы, почему система работает так. При этом система должна рассматриваться как часть большего целого.

Анализ и синтез дополняют друг друга. Так, при синтезе организационной структуры необходимо сначала провести анализ деятельности создаваемой организации, выделить отдельные процессы (функции), сопоставить им организационные единицы, а затем соединить их в отдельное целое, т.е. осуществить синтез. При выборе способа функционирования организации зачастую имеет место обратное: сначала используется синтетический подход – рассматривается деятельность организации, как целого; выбирается общая цель и способ функционирования, а затем осуществляется деагрегация выбранного способа на отдельные функции. Познание систем и использование этих знаний для создания систем и управления ими осуществляется через моделирование.

Понятие декомпозиции систем.

Основной операцией анализа является разделение целого на части. Задача распадается на подзадачи, система — на подсистемы, цели — на подцели и т.д. При необходимости этот процесс повторяется, что приводит к иерархическим древовидным структурам. Обычно (если задача не носит чисто учебного характера) объект анализа сложен, слабо структурирован, плохо формализован, поэтому операцию декомпозиции выполняет эксперт. Если поручить анализ одного и того же объекта разным экспертам, то полученные древовидные списки будут различаться. Качество построенных экспертами деревьев зависит как от их компетентности, так и от применяемой методики декомпозиции.

При разделении целого на части появляются затруднения, когда требуется доказательство полноты и безизбыточности предлагаемого набора частей. Стремясь перейти от эвристического, интуитивного подхода к более осознанному, алгоритмическому выполнению декомпозиции, мы должны объяснить, почему именно так, а не иначе, именно на такое, а не на большее или меньшее количество частей мы разделяем целое. Объяснение состоит в том, что основанием всякой декомпозиции является модель рассматриваемой системы.

Понятие жизненного цикла.

Жизненный цикл (ЖЦ) - одно из базовых понятий методологии проектирования ИС. Это непрерывный процесс, который начинается с момента принятия решения о необходимости создания ИС и заканчивается в момент ее полного изъятия из эксплуатации.

Основным нормативным документом, регламентирующим ЖЦ, является международный стандарт ISO/IEC 12207 (ISO - International Organization of Standardization - Международная организация по стандартизации, IEC - International Electrotechnical Commission - Международная комиссия по электротехнике). Он определяет структуру ЖЦ, содержащую процессы, действия и задачи, которые должны быть выполнены во время создания ИС.

Структура ЖЦ по стандарту ISO/IEC 12207 базируется на трех группах процессов: основные процессы ЖЦ (приобретение, поставка, разработка, эксплуатация, сопровождение); вспомогательные процессы (документирование, управление конфигурацией, обеспечение качества, аттестация, аудит, решение проблем); организационные процессы (управление проектами, создание инфраструктуры проекта, улучшение самого ЖЦ, обучение).

Системный анализ.

Привлечение методов системного анализа для решения указанных проблем необходимо прежде всего потому, что в процессе принятия решений приходится осуществлять выбор в условиях неопределённости, которая обусловлена наличием факторов, не поддающихся строгой количественной оценке. Процедуры и методы системного анализа направлены именно на выдвижение альтернативных вариантов решения проблемы, выявление масштабов неопределённости по каждому из вариантов и сопоставление вариантов по тем или иным критериям эффективности. Специалисты по системному анализу только готовят или рекомендуют варианты решения, принятие же решения остаётся в компетенции соответствующего должностного лица (или органа).

Системный синтез.

Задачи синтеза, сводятся к выбору варианта системы, исходя из заданных ее свойств. Одной из характерных особенностей современного развития автоматизированных систем управления (АСУ) является усложнение структуры различных подсистем, что в первую очередь обусловлено ростом размеров и сложности процессов обработки и передачи информации, а также процессов управления самими подсистемами. Это выдвигает ряд проблем, связанных с научно-обоснованным построением структуры таких систем, эффективным формированием состава подсистем передачи и обработки информации. Актуальными являются вопросы постановки и формализации задач синтеза структур, разработки оптимизационных и имитационных моделей, а также построения на их основе процедур синтеза структуры систем, позволяющих учитывать динамику функционирования элементов системы. В связи со сложностью постановки и решения задач синтеза структуры сложных систем наибольший эффект от их использования может быть достигнут при создании крупномасштабных и типовых систем, в частности АСУ. Существенное влияние на структуру систем управления оказывает развитие средств вычислительной техники: появление многопроцессорных и многомашинных вычислительных комплексов и сетей ЭВМ, а также усложнение процессов обработки и обмена информацией – все это увеличивает число анализируемых вариантов построения системы, повышает требования к эффективности и качеству принимаемых проектных решений по выбору и дальнейшему развитию структуры системы.

Базируясь на оптимизационно-имитационном подходе применительно к синтезу структур сложных АСУ, можно обеспечить совместное использование в процессе синтеза оптимизационных и имитационных моделей их рациональное взаимодействие в оптимизационно-имитационных процедурах, описывающих как состав и взаимосвязи структурных элементов системы, так и динамические и стохастические аспекты их функционирования. С учетом проблем анализа и синтеза структур сложных АСУ, предлагается использовать агрегативно-декомпозиционный подход к формализации и синтезу структур систем на различных уровнях их детализации.

При исследовании структур существующих, а также проектируемых АСУ широко применяются принципы и методы имитационного моделирования и инструментальные средства имитационного моделирования.

Проблема синтеза структуры систем управления включает выбор числа уровней и подсистем управления (иерархии управления); согласование целей подсистем различных уровней; создание контуров принятия решений; оптимальное распределение выполняемых функций (задач, информационных массивов и процедур) по уровням и узлам системы; выбор структуры технических средств передачи и обработки информации.

Задачи синтеза структуры рассматриваемых систем включают: определение оптимального числа, расположения и вариантов построения элементов системы; распределение функций управления по элементам системы и выбор варианта реализации задач управления; выбор мероприятий по обеспечению требуемой живучести систем; распределение функций и задач между техническими средствами; выбор и распределение технических средств по элементам системы и т.д.

Формализация технологии проектирования.

Методика и примеры формализации систем. Формализация описания структуры методами теории графов. Топологическая декомпозиция структур АСУ. Модели описания и анализа потоков информации в АСУ.

Ключевые вопросы:

1. Состав и структура АСОИУ.
2. Анализ и синтез в проектировании систем.
3. Жизненный цикл информационной системы.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 3. Средства структурного анализа.

Диаграммы потоков данных. Диаграммы «Сущность-Связь», Диаграммы переходов состояний. Основные, вспомогательные и организационные процессы. Международный стандарт ISO/IES 12207. Характеристики основных процессов. Вспомогательные процессы для поддержки выполнения основных процессов, обеспечения качества проекта, организации верификации, проверки и тестирования ПО.

Для целей моделирования систем вообще, и структурного анализа в частности, используются три группы средств, иллюстрирующих:

- функции, которые система должна выполнять;
- отношения между данными;
- зависящее от времени поведение системы (аспекты реального времени).

Среди всего многообразия средств решения данных задач в методологиях структурного анализа наиболее часто и эффективно применяемыми являются следующие:

- DFD – диаграммы потоков данных совместно со словарями данных и спецификациями процессов или миниспецификациями;
- ERD – диаграммы «сущность-связь»;
- STD – диаграммы переходов состояний.

Ключевые вопросы:

1. Использование диаграмм при проектировании.
2. Международные стандарты.
3. Вспомогательные процессы для поддержки выполнения основных процессов.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 4. Основные, вспомогательные и организационные процессы. Международный стандарт ISO/IES 12207. Характеристики основных процессов.

Жизненный цикл ИС можно представить как ряд событий, происходящих с системой в процессе ее создания и использования. Стандарт ISO/IEC 12207:1995 – стандарт на процессы и организацию жизненного цикла. Распространяется на все виды заказного ПО.

В соответствии с базовым международным стандартом ISO/IEC 12207 все процессы ЖЦ ПО делятся на три группы:

1. Основные процессы:
 - приобретение;
 - поставка;
 - разработка;
 - эксплуатация;
 - сопровождение.
2. Вспомогательные процессы:
 - документирование;
 - управление конфигурацией;
 - обеспечение качества;
 - разрешение проблем;

- аудит;
 - аттестация;
 - совместная оценка;
 - верификация.
3. Организационные процессы:
- создание инфраструктуры;
 - управление;
 - обучение;
 - усовершенствование.

Вспомогательные процессы предназначены для поддержки выполнения основных процессов, обеспечения качества проекта, организации верификации, проверки и тестирования ПО. Организационные процессы определяют действия и задачи, выполняемые как заказчиком, так и разработчиком проекта для управления своими процессами.

Ключевые вопросы:

1. Международный стандарт ISO/IEC 12207.
2. Процессы жизненного цикла.
3. Вспомогательные процессы.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 5. Дополнительные группы процессов ЖЦ по ИС (международный стандарт ISO/IEC 15288).

Договорные процессы. Процессы предприятия. Проектные процессы. Технические процессы. Специальные процессы.

В 2002 г. был разработан и опубликован стандарт на процессы жизненного цикла систем (ISO/IEC 15288). К разработке стандарта были привлечены специалисты различных областей: системной инженерии, программирования, управления качеством, человеческими ресурсами, безопасностью и пр. Был учтен практический опыт создания систем в правительственных, коммерческих, военных и академических организациях. Стандарт применим для широкого класса систем, но его основное предназначение – поддержка создания компьютеризированных систем.

Согласно этому стандарту в структуру ЖЦ следует включать следующие группы процессов:

1. Договорные процессы – внутренние решения или решения внешнего поставщика:
 - приобретение;
 - поставка.
2. Процессы предприятия:
 - управление окружающей средой предприятия;
 - инвестиционное управление;
 - управление ЖЦ ИС;
 - управление ресурсами;
 - управление качеством.
3. Проектные процессы:
 - планирование проекта;
 - оценка проекта;
 - контроль проекта;
 - управление рисками;

- управление конфигурацией;
 - управление информационными потоками;
 - принятие решений.
4. Технические процессы:
- определение требований;
 - анализ требований;
 - разработка архитектуры;
 - внедрение;
 - интеграция;
 - верификация;
 - переход;
 - аттестация;
 - эксплуатация;
 - сопровождение;
 - утилизация.
5. Специальные процессы:
- определение и установка взаимосвязей исходя из задач и целей.

Ключевые вопросы:

1. Стандарт на процессы жизненного цикла систем (ISO/IEC 15288).
2. Группы процессов в структуре жизненного цикла.
3. Проектные процессы.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 6. Каноническое проектирование. Стадии и этапы проектирования. Модель «AS-IS». Модель «TO-BE».

Организация канонического проектирования ИС ориентирована на использование главным образом каскадной модели жизненного цикла ИС. Стадии и этапы работы описаны в стандарте ГОСТ 34.601-90. В зависимости от сложности объекта автоматизации и набора задач,

требующих решения при создании конкретной ИС, стадии и этапы работ могут иметь различную трудоемкость. Допускается объединять последовательные этапы и даже исключать некоторые из них на любой стадии проекта. Допускается также начинать выполнение работ следующей стадии до окончания предыдущей.

Модели деятельности организации создаются в двух видах:

- модель «как есть» («as-is») – отражает существующие в организации бизнеспроцессы, описывает положение дел на предприятии (оргштатная структура, взаимодействия подразделений, принятые технологии, автоматизированные и неавтоматизированные бизнес-процессы и т.д.) на момент обследования; позволяет понять, что делает и как функционирует данное предприятие с позиций системного анализа, а также на основе автоматической верификации выявить ряд ошибок и узких мест и сформулировать ряд предложений по улучшению ситуации;
- модель «как должно быть» («to-be») – интегрирует перспективные предложения руководства и сотрудников предприятия, экспертов и системных аналитиков и позволяет сформировать видение новых рациональных технологий работы предприятия.

Ключевые вопросы:

1. Формирование требований.
2. Разработка концепции ИС.

3. Техническое задание.
4. Эскизный проект.
5. Технический проект.
6. Рабочая документация.
7. Ввод в действие.
8. Сопровождение.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 7. Состав и содержание ТЗ.

Задачи решаемые при разработке ТЗ. Требования к составу и содержанию ТЗ.

Результаты обследования представляют объективную основу для формирования технического задания на информационную систему.

Техническое задание – это документ, определяющий цели, требования и основные исходные данные, необходимые для разработки автоматизированной системы управления.

При разработке технического задания необходимо решить следующие задачи:

- установить общую цель создания ИС, определить состав подсистем и функциональных задач;
- разработать и обосновать требования, предъявляемые к подсистемам;
- разработать и обосновать требования, предъявляемые к информационной базе, математическому и программному обеспечению, комплексу технических средств (включая средства связи и передачи данных);
- установить общие требования к проектируемой системе;
- определить перечень задач создания системы и исполнителей;
- определить этапы создания системы и сроки их выполнения;
- провести предварительный расчет затрат на создание системы и определить уровень экономической эффективности ее внедрения.

Ключевые вопросы:

1. Типовые требования к составу технического задания.
2. Типовые требования к содержанию технического задания.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 8. Понятие технического проекта. Состав технического проекта.

Комплекс научно-исследовательских и экспериментальных работ для выбора основных проектных решений и расчет экономической эффективности системы.

На основе технического задания и эскизного проекта разрабатывается технический проект ИС.

Технический проект системы – это техническая документация, содержащая общесистемные проектные решения, алгоритмы решения задач, а также оценку экономической эффективности АСУ и перечень мероприятий по подготовке объекта к внедрению.

На этом этапе осуществляется комплекс научно-исследовательских и экспериментальных работ для выбора основных проектных решений и расчет экономической эффективности системы.

Ключевые вопросы:

1. Технический проект.
2. Проектные решения.
3. Экономическая эффективность

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 9. Типовое проектирование ИС. Типовые проектные решения.

Типовое проектирование ИС предполагает создание системы из готовых типовых элементов. Основопологающим требованием для применения методов типового проектирования является возможность декомпозиции проектируемой ИС на множество составляющих компонентов (подсистем, комплексов задач, программных модулей и т.д.). Для реализации выделенных компонентов выбираются имеющиеся на рынке типовые проектные решения, которые настраиваются на особенности конкретного предприятия.

Типовое проектное решение (ТПР) – это тиражируемое (пригодное к многократному использованию) проектное решение.

Принятая классификация ТПР основана на уровне декомпозиции системы. Выделяются следующие классы ТПР:

- элементные ТПР – библиотеки методо-ориентированных программ – типовые решения по задаче или по отдельному виду обеспечения задачи (информационному, программному, техническому, математическому, организационному);
- подсистемные ТПР – пакеты прикладных программ – в качестве элементов типизации выступают отдельные подсистемы, разработанные с учетом функциональной полноты и минимизации внешних информационных связей;
- объектные ТПР – отраслевые проекты ИС – типовые отраслевые проекты, которые включают полный набор функциональных и обеспечивающих подсистем ИС.

Каждое типовое решение предполагает наличие, кроме собственно функциональных элементов (программных или аппаратных), документации с детальным описанием ТПР и процедур настройки в соответствии с требованиями разрабатываемой системы.

Ключевые вопросы:

1. Параметрически-ориентированное проектирование.
2. Модельно-ориентированное проектирование.
3. Репозиторий.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос. ун-та, 2008.-112 с.

Лекция 10. Классификация структурных методологий, их характеристика.

Методология структурного анализа и проектирования ПО определяет руководящие указания для оценки и выбора проекта разрабатываемого ПО, шаги работы, которые должны быть выполнены, их последовательность, правила распределения и назначения операций и методов.

В настоящее время успешно используются практически все известные методологии структурного анализа и проектирования, однако наибольшее распространение получили методологии SADT, структурного системного анализа Гейна-Сарсона, структурного анализа и

проектирования Йодана/ДеМарко, развития систем Джексона, развития структурных систем Варнье-Орра, анализа и проектирования систем реального времени Уорда-Меллора и Хатли, информационного моделирования Мартина.

Современные структурные методологии анализа и проектирования классифицируются по следующим признакам:

- по отношению к школам – Software Engineering (SE) и Information Engineering (IE);
- по порядку построения модели – процедурно-ориентированные, ориентированные на данные и информационно-ориентированные;
- по типу целевых систем – для систем реального времени и для ИС.

Ключевые вопросы:

1. Технические структурные карты Константайна и Джексона.
2. Диаграммы потоков данных в нотации Йодана/Де Марко.
3. Методологии SADT.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 11. Методология SADT.

Методология SADT разработана Дугласом Россом. На ее основе разработана методология IDEF0. Методология SADT представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной модели объекта предметной области. Функциональная модель SADT отображает функциональную структуру объекта, т.е. производимые им действия и связи между этими действиями. Основные элементы этой методологии основываются на следующих концепциях:

- графическое представление блочного моделирования. Графика блоков и дуг SADT-диаграммы отображает функцию в виде блока, а интерфейсы входа/выхода представляются дугами, соответственно входящими в блок и выходящими из него. Взаимодействие блоков друг с другом описываются посредством интерфейсных дуг, выражающих «ограничения», которые, в свою очередь, определяют, когда и каким образом функции выполняются и управляются;
- строгость и точность.

Выполнение правил SADT требует достаточной строгости и точности, не накладывая в то же время чрезмерных ограничений на действия аналитика. Правила SADT включают:

- ограничение количества блоков на каждом уровне декомпозиции;
- связность диаграмм (номера блоков);
- уникальность меток и наименований (отсутствие повторяющихся имен);
- синтаксические правила для графики (блоков и дуг);
- разделение входов и управлений (правило определения роли данных);
- отделение организации от функции, т.е. исключение влияния организационной структуры на функциональную модель.

Методология SADT может использоваться для моделирования широкого круга систем и определения требований и функций, а затем для разработки системы, которая удовлетворяет этим требованиям и реализует эти функции.

Для уже существующих систем SADT может быть использована для анализа функций, выполняемых системой, а также для указания механизмов, посредством которых они осуществляются.

Результатом применения методологии SADT является модель, которая состоит из диаграмм, фрагментов текстов и глоссария, имеющих ссылки друг на друга. Диаграммы – главные компоненты модели.

Ключевые вопросы:

1. Концепции методологии.
2. Построение SADT-модели.
3. Декомпозиция модели.
4. Типы связей между функциями.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 12. Методология структурного системного анализа Гейна-Сарсона.

Структурный анализ – это систематический пошаговый подход к анализу требований и проектированию спецификаций системы независимо от того, является ли она существующей или создается вновь.

Целью методологии Гейна-Сарсона является преобразование общих, неясных знаний о требованиях к системе в точные определения. Обе методологии фокусируют внимание на потоках данных, их главное назначение – создание базированных на графике документов по функциональным требованиям. Методологии поддерживаются традиционными нисходящими методами проектирования спецификаций и обеспечивают один из лучших способов связи между аналитиками, разработчиками и пользователями системы за счет интеграции множества следующих средств:

1. DFD – диаграммы потоков данных. Являются графическими иерархическими спецификациями, описывающими систему с позиций потоков данных. В состав DFD могут входить четыре графических символа, представляющих потоки данных, процессы преобразования входных потоков данных в выходные, внешние источники и получатели данных, а также файлы и БД, требуемые процессами для своих операций.
2. Словари данных. Являются каталогами всех элементов данных, присутствующих в DFD, включая групповые и индивидуальные потоки данных, хранилища и процессы, а также все их атрибуты.
3. Миниспецификации обработки, описывающие DFD-процессы нижнего уровня и являющиеся базой для кодогенерации. Фактически миниспецификации представляют собой алгоритмы описания задач, выполняемых процессами: множество всех миниспецификации является полной спецификацией системы. Миниспецификации содержат номер и/или имя процесса, списки входных и выходных данных и тело процесса, собственно и являющееся спецификацией алгоритма или операции, трансформирующей входные потоки данных в выходные. DFD-диаграммы являются ключевой частью документа спецификации требований. Каждый узел – процесс в DFD может развертываться в диаграмму нижнего уровня, что позволяет на любом уровне абстрагироваться от деталей.

Ключевые вопросы:

1. Графические иерархические спецификации.
2. Словари данных.
3. DFD – диаграммы потоков данных.
4. Миниспецификации обработки.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 13. Спиральная модель этапов проектирования АСОИУ.

Используется подход к организации проектирования ИС «сверху-вниз», когда сначала определяется состав функциональных подсистем, а затем постановка отдельных задач. Соответственно сначала разрабатываются такие общесистемные вопросы, как организация базы данных, технология сбора, передачи и накопления информации, а затем технология решения конкретных задач. В рамках комплексов задач программирование осуществляется по направлению от головных программных модулей к исполняющим отдельные функции модулям. При этом на первый план выходят вопросы взаимодействия интерфейсов программных модулей между собой и с базой данных, а на второй план – реализация алгоритмов.

В основе спиральной модели жизненного цикла лежит применение прототипной технологии или RAD-технологии (технологии быстрой разработки приложений).

Согласно этой технологии ИС разрабатывается путем расширения программных прототипов, повторяя путь от детализации требований к детализации программного кода. Естественно, что при прототипной технологии сокращается число итераций и меньше возникает ошибок и несоответствий, которые необходимо исправлять на последующих итерациях, а само проектирование ИС осуществляется более быстрыми темпами, упрощается создание проектной документации. Для более точного соответствия проектной документации разработанной ИС все большее значение придается ведению общесистемного репозитория и использованию CASE-технологий.

Жизненный цикл при использовании RAD-технологии предполагает активное участие на всех этапах разработки конечных пользователей будущей системы и включает четыре основные стадии информационного инжиниринга:

1. Анализ и планирование информационной стратегии. Пользователи вместе со специалистами-разработчиками участвуют в идентификации проблемной области.
2. Проектирование. Пользователи принимают участие в техническом проектировании под руководством специалистов-разработчиков.
3. Конструирование. Специалисты-разработчики проектируют рабочую версию ИС с использованием ЯВУ.
4. Внедрение. Специалисты-разработчики обучают пользователей работе в среде новой ИС.

Ключевые вопросы:

1. Взаимодействие интерфейсов программных модулей.
2. Прототипные технологии (RAD-технологии).
3. CASE-технологии.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 14. Методология функционального моделирования.

Стандарт IDEF0 представляет совокупность методов, правил и процедур, предназначенных для построения функциональной модели, являющейся иерархически связанным структурным представлением действия (или множества действий) некоторого объекта, а также вещественных и информационных объектов (данных), необходимых для функционирования или являющихся результатом этого функционирования.

Функциональная модель бизнес-процессов состоит из диаграмм, фрагментов текста и глоссария, имеющих ссылки друг на друга. Диаграммы – главные компоненты модели, которые отображают последовательности взаимосвязанных через общие объекты функций бизнес-процесса. Достоинство функциональной модели заключается в простоте графического представления, которое использует всего два конструктивных элемента:

- функциональный блок – описание функций, операций, действий, работ;

- интерфейсная дуга – линия, связывающая функциональные блоки и описывающая объекты (потоки объектов).

Методология IDEF0 основана на следующих положениях:

1. Цели моделирования. Модель разрабатывается для понимания, анализа и принятия решений о реорганизации или замене существующего либо проектировании нового БП. Модель описывает, что происходит в БП, как им управляют, какие сущности он преобразует, какие ресурсы использует и что производит, частями БП могут быть любые комбинации разнообразных сущностей, включающие людей, информацию, программное обеспечение, оборудование, изделия, сырье или энергию (энергонасосители).
2. Блочное моделирование и его графическое представление. Изучаемый БП представляется в виде набора взаимодействующих и взаимосвязанных блоков, отображающих работы, операции, действия. В IDEF0 работы, операции, действия, происходящие в БП и его элементах, принято называть функциями. Каждой функции ставится в соответствие блок. На IDEF0-диаграмме, основном документе при анализе и проектировании БП, блок представляется прямоугольником. Интерфейсы, посредством которых блок взаимодействует с другими блоками или с внешним по отношению к моделируемому БП окружению, представляются стрелками, входящими в блок или выходящими из него.
3. Лаконичность и точность. Документация, описывающая БП, должна быть точной и лаконичной. Многословные характеристики, изложенные в форме традиционных текстов, неудобны. Графический язык позволяет лаконично, однозначно и точно показать все блоки БП и все отношения и связи между ними, выявить ошибочные, лишние или дублирующие связи.
4. Передача информации. Средства IDEF0 облегчают передачу информации от одного участника разработки модели к другому. К числу таких средств относятся:
 - 4.1. диаграммы, основанные на простой графике блоков и стрелок, легко читаемые и понимаемые;
 - 4.2. метки на естественном языке для описания блоков и стрелок, а также глоссарий и сопроводительный текст для уточнения смысла элементов диаграммы;
 - 4.3. последовательная декомпозиция диаграмм, строящаяся по иерархическому принципу, при котором на верхнем уровне отображаются основные функции, а затем, на нижних уровнях, происходит их детализация и уточнение;
 - 4.4. древовидные схемы иерархии диаграмм и блоков, обеспечивающие обзорность модели в целом и всех входящих в нее деталей.
5. Строгость и формализм. Разработка моделей IDEF0 требует соблюдения ряда строгих формальных правил, обеспечивающих преимущества методологии в отношении однозначности, точности и целостности сложных многоуровневых моделей.
6. Итерационное моделирование. Разработка модели в IDEF0 представляет собой итерационную процедуру. На каждом шаге итерации разработчик предлагает вариант модели, который подвергают обсуждению, рецензированию и последующему редактированию, после чего цикл повторяется. Такая организация работы способствует оптимальному использованию знаний системного аналитика, владеющего методологией и техникой IDEF0, и знаний специалистов – экспертов в предметной области, к которой относится объект моделирования.
7. Отделение «организации» от «функции». При разработке моделей следует избегать изначальной «привязки» функций исследуемой системы к существующей организационной структуре моделируемого объекта. Это помогает избежать субъективной точки зрения, навязанной организационной структурой и ее руководством. Организационная структура должна явиться результатом использования модели. Сравнение результата с существующей структурой позволяет, во-первых, оценить адекватность модели, а во-вторых, предложить решения, направленные на совершенствование этой структуры.

Ключевые вопросы:

1. Стандарт IDEF0.
2. Методология IDEF0.
3. Компоненты синтаксиса IDEF0.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 15. Методология потоков данных DFD.

В основе данной методологии лежит построение модели анализируемой ИС – проектируемой или реально существующей. В соответствии с методологией модель системы определяется как иерархия диаграмм потоков данных (DFD), описывающих асинхронный процесс преобразования информации от ее ввода в систему до выдачи пользователю. Диаграммы верхних уровней иерархии (контекстные диаграммы) определяют основные процессы или подсистемы ИС с внешними входами и выходами. Они детализируются при помощи диаграмм нижнего уровня. Такая декомпозиция продолжается, создавая многоуровневую иерархию диаграмм, до тех пор, пока не будет достигнут такой уровень декомпозиции, на котором процесс становится элементарными и детализировать их далее невозможно.

Источники информации (внешние сущности) порождают информационные потоки (потоки данных), переносящие информацию к подсистемам или процессам. Те в свою очередь преобразуют информацию и порождают новые потоки, которые переносят информацию к другим процессам или подсистемам, накопителям данных или внешним сущностям – потребителям информации.

Таким образом, основными компонентами диаграмм потоков данных являются:

- внешние сущности;
- системы/подсистемы;
- процессы;
- накопители данных;
- потоки данных.

Ключевые вопросы:

1. Диаграммы уровней иерархии.
2. Основные компоненты диаграмм потоков данных.
3. Правила детализации.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 16. Методология IDEF3.

Для описания логики взаимодействия информационных потоков существует методология IDEF3, использующая графическое описание информационных потоков, взаимоотношений между процессами обработки информации и объектов, являющихся частью этих процессов. Диаграммы Workflow могут быть использованы в моделировании бизнес-процессов для анализа завершенности процедур обработки информации. С их помощью можно описывать сценарии действий сотрудников организации, например последовательность обработки заказа или события, которые необходимо обработать за конечное время. Каждый сценарий сопровождается описанием процесса и может быть использован для документирования каждой функции.

IDEF3 – это метод, описывающий ситуацию, когда процессы выполняются в определенной последовательности, а также описать объекты, участвующие совместно в одном процессе.

В отличие от других методик описаний процессов IDEF3 не ограничивает аналитика чрезмерно жесткими рамками синтаксиса, что может привести к созданию неполных или противоречивых моделей.

Каждая работа в IDEF3 описывает какой-либо сценарий бизнес-процесса и может являться составляющей другой работы. Поскольку сценарий описывает цель и рамки модели, важно, чтобы работы именовались отглагольным существительным, обозначающим процесс действия, или фразой, содержащей такое существительное.

Точка зрения на модель должна быть документирована. Обычно это точка зрения человека, ответственного за работу в целом. Также необходимо документировать цель модели – те вопросы, на которые призвана ответить модель.

Диаграмма является основной единицей описания в IDEF3. Важно правильно построить диаграммы, поскольку они предназначены для чтения другими людьми (а не только автором).

Единицы работы – Unit of Work (UOW) – также называемые работами (activity), являются центральными компонентами модели. В IDEF3 работы изображаются прямоугольниками с прямыми углами и имеют имя, выраженное отглагольным существительным и другое имя существительное, отображающее основной выход работы (например, «Изготовление изделия»). Также имеет идентификатор работы, который присваивается при создании и не меняется никогда. Даже если работа будет удалена, ее идентификатор не будет вновь использоваться для других работ.

Связи показывают взаимоотношения работ. Все связи в IDEF3 однонаправлены и могут быть направлены куда угодно, но обычно диаграммы IDEF3 стараются построить так, чтобы связи были направлены слева направо.

Ключевые вопросы:

1. Логика взаимодействия информационных потоков.
2. Диаграммы Workflow.
3. Единицы работы – Unit of Work (UOW).
4. Связи в IDEF3.
5. Перекрестки для слияния и разветвления стрелок.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 17. Методология разработки информационной базы IDEF1X.

Метод IDEF1, разработанный Т. Рэмей, позволяет построить модель данных, эквивалентную реляционной модели в третьей нормальной форме. На основе совершенствования методологии IDEF1 создана ее новая версия – методология IDEF1X. IDEF1X разработана с учетом таких требований, как простота изучения и возможность автоматизации. IDEF1X-диаграммы используются рядом распространенных CASE-средств (в частности, ERwin, Design/IDEF).

Сущность в методологии IDEF1X является независимой от идентификаторов или просто независимой, если каждый экземпляр сущности может быть однозначно идентифицирован без определения его отношений с другими сущностями. Сущность называется зависимой от идентификаторов или просто зависимой, если однозначная идентификация экземпляра сущности зависит от его отношения к другой сущности.

Ключевые вопросы:

1. Сущность в методологии IDEF1X.
2. Связи.
3. Атрибуты.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 18. Методология UML.

Диаграмма вариантов использования – это исходное концептуальное представление или концептуальная модель системы в процессе ее проектирования и разработки. Создание диаграммы вариантов использования имеет следующие цели:

- определить общие границы и контекст моделируемой предметной области на начальных этапах проектирования системы;
- сформулировать общие требования к функциональному поведению проектируемой системы;
- разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей;
- подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Назначение данной диаграммы состоит в следующем: проектируемая программная система представляется в форме так называемых вариантов использования, с которыми взаимодействуют внешние сущности – актеры.

Актером называется любой объект, субъект или система, взаимодействующая с моделируемой бизнес-системой извне. Вариант использования служит для описания сервисов, которые система предоставляет актеру. Другими словами каждый вариант использования определяет набор действий, совершаемый системой при диалоге с актером. При этом ничего не говорится о том, каким образом будет реализовано взаимодействие актеров с системой и собственно выполнение вариантов использования.

В самом общем случае, диаграмма вариантов использования представляет собой граф специального вида, который является графической нотацией для представления конкретных вариантов использования, актеров и отношений между этими элементами. При этом отдельные элементы диаграммы заключают в прямоугольник, который обозначает границы проектируемой системы. В то же время отношения, которые могут быть изображены на данном графе, представляют собой только фиксированные типы взаимосвязей между актерами и вариантами использования, которые в совокупности описывают сервисы или функциональные требования к моделируемой системе.

Базовыми элементами диаграммы вариантов использования являются вариант использования и актер.

Вариант использования – внешняя спецификация последовательности действий, которые система или другая сущность могут выполнять в процессе взаимодействия с актерами.

Содержание варианта использования может быть представлено в форме дополнительного пояснительного текста, который раскрывает смысл или семантику действий при выполнении данного варианта использования. Такой пояснительный текст получил название текста-сценария или просто сценария.

Диаграмма классов – диаграмма языка UML, на которой представлена совокупность декларативных или статических элементов модели, таких как классы с атрибутами и операциями, а также связывающие их отношения.

Диаграмма классов предназначена для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. При этом диаграмма классов может содержать интерфейсы, пакеты, отношения и даже отдельные экземпляры классификаторов, такие как объекты и связи.

Класс – абстрактное описание множества однородных объектов, имеющих одинаковые атрибуты, операции и отношения с объектами других классов.

Графически класс изображается в виде прямоугольника, который дополнительно может быть разделен горизонтальными линиями на разделы или секции.

Диаграмма последовательности – диаграмма, на которой показаны взаимодействия объектов, упорядоченные по времени их проявления.

На диаграмме последовательности неявно присутствует ось времени, что позволяет визуализировать временные отношения между передаваемыми сообщениями. С помощью диаграммы последовательности можно представить взаимодействие элементов модели как временной график жизни всей совокупности объектов, связанных между собой для реализации варианта использования программной системы, достижения бизнес-цели или выполнения какой-либо задачи.

На диаграмме последовательности изображаются объекты, которые непосредственно участвуют во взаимодействии, при этом никакие статические связи с другими объектами не визуализируются. Для диаграммы последовательности ключевым моментом является динамика взаимодействия объектов во времени. При этом диаграмма последовательности имеет как бы два измерения.

Одно – слева направо в виде вертикальных линий, каждая из которых изображает линию жизни отдельного объекта, участвующего во взаимодействии. Второе измерение диаграммы последовательности – вертикальная временная ось, направленная сверху вниз.

Диаграмма состояний – диаграмма, которая представляет конечный автомат.

Главное назначение диаграммы состояний – описать возможные последовательности состояний и переходов, которые в совокупности характеризуют поведение моделируемой системы в течение всего ее жизненного цикла. Диаграмма состояний представляет динамическое поведение сущностей, на основе спецификации их реакции на восприятие некоторых конкретных событий.

Системы, которые реагируют на внешние действия от других систем или от пользователей, иногда называют реактивными. Если такие действия инициируются в произвольные случайные моменты времени, то говорят об асинхронном поведении модели.

Диаграммы состояний чаще всего используются для описания поведения отдельных систем и подсистем. Они также могут быть применены для спецификации функциональности экземпляров отдельных классов, т. е. для моделирования всех возможных изменений состояний конкретных объектов.

Диаграмма состояний по существу является графом специального вида, который служит для представления конечного автомата.

Ключевые вопросы:

1. Диаграммы вариантов использования.
2. Диаграммы классов.
3. Диаграммы последовательностей.
4. Диаграммы состояний.
5. Диаграммы кооперации.
6. Диаграммы деятельности.
7. Диаграммы компонентов.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 19. Эскизный проект. Рабочий проект.

Эскизный проект предусматривает разработку предварительных проектных решений по системе и ее частям.

Выполнение стадии эскизного проектирования не является строго обязательной. Если основные проектные решения определены ранее или достаточно очевидны для конкретной ИС и объекта автоматизации, то эта стадия может быть исключена из общей последовательности работ.

Содержание эскизного проекта задается в ТЗ на систему. Как правило, на этапе эскизного проектирования определяются:

- функции ИС;
- функции подсистем, их цели и ожидаемый эффект от внедрения;
- состав комплексов задач и отдельных задач;
- концепция информационной базы и ее укрупненная структура;
- функции системы управления базой данных;
- состав вычислительной системы и других технических средств;
- функции и параметры основных программных средств.

По результатам проделанной работы оформляется, согласовывается и утверждается документация в объеме, необходимом для описания полной совокупности принятых проектных решений и достаточном для дальнейшего выполнения работ по созданию системы.

Рабочий проект – это техническая документация, утвержденная в установленном порядке, содержащая уточненные данные и детализированные общесистемные проектные решения, программы и инструкции по решению задач, а также уточненную оценку экономической эффективности автоматизированной системы управления и уточненный перечень мероприятий по подготовке объекта к внедрению.

Рабочий проект разрабатывается на основе технического проекта, утвержденного заказчиком.

На этапе рабочего проектирования заказчик должен закончить работы по подготовке объекта к внедрению системы, подготовить помещения для установки компьютеров, организовать учебу работников всех звеньев организационной структуры, разместить заказы на изготовление нестандартного оборудования.

Ключевые вопросы:

1. Содержание эскизного проекта.
2. Состав рабочей документации.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 20. Принципы построения функциональной структуры АСОИУ.

Функциональная подсистема ИС представляет собой комплекс задач с высокой степенью информационных обменов между задачами. Под задачей понимается некоторый процесс обработки информации с четко определенным множеством входной и выходной информации.

Состав функциональных подсистем во многом определяется особенностями информационной системы, ее отраслевой принадлежностью, формой собственности, размером, характером деятельности предприятия.

Функциональные подсистемы ИС могут строиться по различным принципам:

- предметному;
- функциональному;
- проблемному;
- смешанному (предметно-функциональному).

Так, с учетом предметной направленности использования ИС в хозяйственных процессах промышленного предприятия выделяют подсистемы, соответствующие управлению отдельными ресурсами:

- управление сбытом готовой продукции;
- управление производством;
- управление материально-техническим снабжением;
- управление финансами;
- управление персоналом.

При этом в подсистемах рассматривается решение задач на всех уровнях управления, обеспечивая интеграцию информационных потоков по вертикали.

Для реализации функций управления выделяют следующие подсистемы:

- планирование;
- регулирование (оперативное управление);
- учет;
- анализ.

Проблемный принцип формирования подсистем отражает необходимость гибкого и оперативного принятия управленческих решений по отдельным проблемам, например решение задач бизнес-планирования, управления проектами. Такие подсистемы могут реализовываться в виде локальных информационных систем, импортирующих данные из корпоративной информационной системы, или в виде специальных подсистем в рамках корпоративной ИС.

На практике чаще всего применяется смешанный предметно-функциональный подход, согласно которому построение функциональной структуры ИС – это разделение ее на подсистемы по характеру хозяйственной деятельности, которое должно соответствовать структуре объекта и системе управления, а также характеру выполняемых функций управления.

Ключевые вопросы:

1. Состав функциональных подсистем.
2. Принципы построения функциональных подсистем.
3. Проблемный принцип формирования подсистем.
4. Предметно-функциональный подход.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 21. Обеспечивающие подсистемы информационных систем.

Обеспечивающие подсистемы ИС являются общими для всей ИС независимо от конкретных функциональных подсистем, в которых применяются те или иные виды обеспечения. Состав обеспечивающих подсистем не зависит от выбранной предметной области. В состав обеспечивающих подсистем входят подсистемы организационного, правового, технического, математического, программного, информационного,

лингвистического и технологического обеспечения.

Подсистема «Организационное обеспечение» (ОО) является одной из важнейших подсистем ИС, от которой зависит успешная реализация целей и функций системы. В составе организационного обеспечения можно выделить четыре группы компонентов:

1. Методические материалы, регламентирующие процесс создания и функционирования системы:

- общепромышленные руководящие методические материалы по созданию ИС;
- типовые проектные решения;
- методические материалы по организации и проведению предпроектного обследования на предприятии;
- методические материалы по вопросам создания и внедрения проектной документации.

2. Совокупность средств, необходимых для эффективного проектирования и функционирования ИС (комплексы задач управления, включая типовые пакеты прикладных программ, типовые структуры управления предприятием, унифицированные системы документов, общесистемные и отраслевые классификаторы и т.п.).

3. Техническая документация, получаемая в процессе обследования, проектирования и внедрения системы: технико-экономическое обоснование, техническое задание, технический и рабочий проекты и документы, оформляющие поэтапную сдачу системы в эксплуатацию.

Организационно-штатная структура проекта, определяющая состав главных конструкторов системы и специалистов по функциональным подсистемам управления.

Подсистема «Правовое обеспечение» (ПрО) предназначена для регламентации процесса создания и эксплуатации ИС, которая включает совокупность юридических документов с констатацией регламентных отношений по формированию, хранению, обработке промежуточной и результатной информации системы.

К правовым документам, действующим на этапе создания системы, относятся:

- договор между разработчиком и заказчиком;
- документы, регламентирующие отношения между участниками процесса создания системы.

К правовым документам, создаваемым на этапе внедрения, относятся:

- характеристика статуса создаваемой системы;
- правовые полномочия подразделений ИС;
- правовые полномочия отдельных видов процессов обработки информации;
- правовые отношения пользователей в применении технических средств.

Подсистема «Техническое обеспечение» (ТО) представляет комплекс технических средств, предназначенных для обработки данных в ИС. В состав комплекса входят ЭВМ, осуществляющие обработку экономической информации, средства подготовки данных на машинных носителях, средства сбора и регистрации информации, средства передачи данных по каналам связи, средства накопления и хранения данных и выдачи результатной информации, вспомогательное оборудование и организационная техника.

Подсистема «Математическое обеспечение» (МО) – это совокупность математических моделей и алгоритмов для решения задач и обработки информации с применением вычислительной техники, а также комплекс средств и методов, позволяющих строить экономико-математические модели задач управления.

В состав МО входят:

- средства МО (средства моделирования типовых задач управления, методы многокритериальной оптимизации, математической статистики, теории массового обслуживания и др.);
- техническая документация (описание задач, алгоритмы решения задач, экономико-математические модели);
- методы выбора МО (методы определения типов задач, оценки вычислительной сложности алгоритмов, оценки достоверности результатов).

Подсистема «Программное обеспечение» (ПО) включает совокупность компьютерных программ, описаний и инструкций по их применению на ЭВМ. ПО делится на два комплекса: общее (операционные системы, операционные оболочки, компиляторы, интерпретаторы, программные среды для разработки прикладных программ, СУБД, сетевые программы и т.д.) и специальное (совокупность прикладных программ, разработанных для конкретных задач в рамках функциональных подсистем).

Подсистема «Информационное обеспечение» (ИО) – это совокупность единой системы классификации и кодирования технико-экономической информации, унифицированной системы документации и информационной базы.

В состав ИО включаются два комплекса: компоненты немашинного информационного обеспечения (классификаторы технико-экономической информации и документы) и внутримашинного информационного обеспечения (макеты/экранные формы для ввода первичных данных в ЭВМ или вывода результатной информации, структура информационной базы: входных, выходных файлов, базы данных).

Подсистема «Лингвистическое обеспечение» (ЛО) включает совокупность научно-технических терминов и других языковых средств, используемых в информационных системах, а также правил формализации естественного языка, включающих методы сжатия и раскрытия текстовой информации с целью повышения эффективности автоматизированной обработки информации и облегчающих общение человека с ИС. Языковые средства, включенные в подсистему ЛО, делятся на две группы: традиционные языки (естественные, математические, алгоритмические, языки моделирования) и языки, предназначенные для диалога с ЭВМ (информационно-поисковые языки, языки СУБД, языки операционных сред, входные языки пакетов прикладных программ).

Подсистема «Технологическое обеспечение» (ТО) ИС соответствует разделению ИС на подсистемы по технологическим этапам обработки различных видов информации:

- первичной и результатной информации (этапы технологического процесса сбора, передачи, накопления, хранения, обработки первичной информации, получения и выдачи результатной информации);
- организационно-распорядительной документации (этапы получения входящей документации, передачи на исполнение, этапы формирования и хранения дел, составления и размножения внутренних документов и отчетов);
- технологической документации и чертежей (этапы ввода в систему и актуализации шаблонов изделий, ввода исходных данных и формирования проектной документации для новых видов изделий, выдачи на плоттер чертежей, актуализации банка ГОСТов, ОСТов, технических условий, нормативных данных, подготовки и выдачи технологической документации по новым видам изделий);
- баз данных и знаний (этапы формирования баз данных и знаний, ввода и обработки запросов на поиск решения, выдачи варианта решения и объяснения к нему);
- научно-технической информации, ГОСТов и технических условий, правовых документов и дел (этапы формирования поисковых образов документов, формирования информационного фонда, ведения тезауруса справочника ключевых слов и их кодов, кодирования запроса на поиск, выполнения поиска и выдачи документа или адреса хранения документа).

Ключевые вопросы:

1. Подсистема «Организационное обеспечение».
2. Подсистема «Правовое обеспечение».
3. Подсистема «Техническое обеспечение».
4. Подсистема «Математическое обеспечение».
5. Подсистема «Программное обеспечение».
6. Подсистема «Информационное обеспечение».
7. Подсистема «Лингвистическое обеспечение».
8. Подсистема «Технологическое обеспечение».

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 22. Проектирование информационного обеспечения АСОИУ.

Основной компонентой немашинного информационного обеспечения ИС является система документации, применяемая в процессе управления экономическим объектом. Под документом понимается определенная совокупность сведений, используемая при решении технико-экономических задач, расположенная на материальном носителе в соответствии с установленной формой.

Система документации – это совокупность взаимосвязанных форм документов, регулярно используемых в процессе управления экономическим объектом. Отличительной особенностью системы экономической документации является большое разнообразие видов документов.

Существующие системы документации, характерные для неавтоматизированных ИС, отличаются большим количеством разных типов форм документов, большим объемом потоков документов и их запутанностью, дублированием информации в документах и работ по их обработке и, как следствие, низкой достоверностью получаемых результатов. Для того чтобы упростить систему документации, используют следующие два подхода:

- проведение унификации и стандартизации документов;
- введение безбумажной технологии, основанной на использовании электронных документов и новых информационных технологий их обработки.

Унификация документов выполняется путем введения единых форм документов. Т. о., вводится единообразие в наименования показателей, единиц измерения и терминов, в результате чего получается унифицированная система документации.

Унифицированная система документации (УСД) — это рационально организованный комплекс взаимосвязанных документов, который отвечает единым правилам и требованиям и содержит информацию, необходимую для управления некоторым экономическим объектом. По уровням управления, они делятся на межотраслевые системы документации, отраслевые и системы документации локального уровня, т. е. обязательные для использования в рамках предприятий или организаций.

Ключевые вопросы:

1. Принципы проектирования документооборота.
2. Система документации.
3. Требования к унифицированной системе документооборота.
4. Проектирование систем входных и выходных документов.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 23. Немашинное информационное обеспечение.

Для того чтобы обеспечить эффективный поиск, обработку на ЭВМ и передачу по каналам связи технико-экономической информации, ее необходимо представить в цифровом виде. С этой целью ее нужно сначала упорядочить (классифицировать), а затем формализовать с использованием классификатора.

Классификация – это разделение множества объектов на подмножества по их сходству или различию в соответствии с принятыми методами.

Классификация фиксирует закономерные связи между классами объектов. Под объектом понимается любой предмет, процесс, явление материального или нематериального свойства. Система классификации позволяет сгруппировать объекты и выделить определенные классы, которые будут характеризоваться рядом общих свойств. Таким образом, совокупность правил распределения объектов множества на подмножества называется системой классификации.

Свойство или характеристика объекта классификации, которое позволяет установить его сходство или различие с другими объектами классификации, называется признаком классификации. Множество или подмножество, объединяющее часть объектов классификации по одному или нескольким признакам, носит название классификационной группировки.

Классификатор – это документ, с помощью которого осуществляется формализованное описание информации в ИС, содержащей наименования объектов, наименования классификационных группировок и их кодовые обозначения.

Каждая система классификации характеризуется свойствами:

- гибкость системы – это способность допускать включение новых признаков, объектов без разрушения структуры классификатора. Необходимая гибкость определяется временем жизни системы;
- емкость системы – это наибольшее количество классификационных группировок, допускаемое в данной системе классификации;
- степень заполненности системы – определяется как частное от деления фактического количества группировок на величину емкости системы.

В настоящее время чаще всего применяются два типа систем классификации: иерархическая и многоаспектная.

При использовании иерархического метода классификации происходит последовательное разделение множества объектов на подчиненные, зависимые классификационные группировки. Получаемая на основе этого процесса классификационная схема имеет иерархическую структуру.

К положительным сторонам данной системы следует отнести логичность, простоту ее построения и удобство логической и арифметической обработки.

Серьезным недостатком иерархического метода классификации является жесткость классификационной схемы. Она обусловлена заранее установленным выбором признаков классификации и порядком их использования по ступеням классификации. Гибкость системы обеспечивается за счет избыточности в ветвях, что приводит к слабой заполненности структуры классификатора.

Многоаспектная система – это система классификации, которая использует параллельно несколько независимых признаков (аспектов) в качестве основания классификации. Аспект – точка зрения на объект классификации, который характеризуется одним или несколькими признаками.

Существуют два типа многоаспектных систем: фасетная и дескрипторная.

Фасет – это аспект классификации, который используется для образования независимых классификационных группировок.

Дескриптор – ключевое слово, определяющее некоторое понятие, которое формирует описание объекта и дает принадлежность этого объекта к классу, группе и т.д.

Ключевые вопросы:

1. Основные понятия классификации информации.
2. Система классификации и кодирования информации.
3. Понятия и основные требования к системе кодирования информации.
4. Состав и содержание операций проектирования классификаторов.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 24. Внутримашинное информационное обеспечение.

Внутримашинное информационное обеспечение включает макеты для ввода первичных данных в ЭВМ или вывода результатной информации, и структуры информационной базы: входных, выходных файлов, базы данных.

Электронная форма документа (ЭД) – это страница с пустыми полями, оставленными для заполнения пользователем. Формы могут допускать различный тип входной информации и содержать командные кнопки, переключатели, выпадающие меню или списки для выбора.

Технология обработки электронных документов требует использования специализированного программного обеспечения – программ управления документооборотом, которые зачастую встраиваются в корпоративные ИС.

Проектирование форм электронных документов обычно включает в себя выполнение следующих шагов:

- создание структуры ЭД – подготовка внешнего вида с помощью графических средств проектирования;
- определение содержания формы ЭД – выбор способов, которыми будут заполняться поля. Поля могут быть заполнены вручную или посредством выбора значений из какого-либо списка, меню, базы данных;
- определения перечня макетов экранных форм – по каждой задаче проектировщик анализирует «постановку» каждой задачи, в которой приводятся перечни используемых входных документов с оперативной и постоянной информацией и документов с результатной информацией;
- определение содержания макетов – выполняется на основе анализа состава реквизитов первичных документов с постоянной и оперативной информацией и результатных документов.

Работа заканчивается программированием разработанных макетов экранных форм и их апробацией.

Основной частью внутримашинного информационного обеспечения является информационная база.

Ключевые вопросы:

1. Проектирование форм электронных документов.
2. Информационная база.
3. Требования к организации хранения файлов в информационной базе.
4. Интегрированная информационная база.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

Лекция 25. Ввод в эксплуатацию АСОИУ.

Ввод в эксплуатацию АСУ и ее отдельных элементов представляет собой процесс постепенного перехода от существующих методов «ручного» управления к методам автоматизированного управления. Проводится он силами заказчика при участии разработчика и организаций-исполнителей.

Взаимодействие организаций заказчика, разработчика и соисполнителей осуществляется на основании договорных условий и плана-графика сдачи АСУ в промышленную эксплуатацию.

К вводу АСУ в эксплуатацию следует приступать при наличии: оформленных документов о выполнении плана мероприятий по подготовке объекта; рабочей документации на внедрение выделенной очереди или АСУ в целом; обученного персонала, обеспечивающего подготовку к вводу АСУ в эксплуатацию; принятых в эксплуатацию технических средств АСУ, обеспечивающих функционирование внедряемых комплексов задач и подсистем.

На стадии ввода АСУ в эксплуатацию осуществляются:

- опытная эксплуатация отдельных задач и их комплексов;
- приемка комплексов задач в промышленную эксплуатацию;
- проведение приемо-сдаточных испытаний;
- приемка АСУ в промышленную эксплуатацию.

Состав и очередность выполнения работ определяются согласованными планами-графиками ввода в эксплуатацию, в которых указываются состав и сроки выполнения работ: по строительству, монтажу, наладке и испытанию объектов АСУ с момента получения рабочей документации до сдачи объектов в промышленную эксплуатацию; по проведению опытной эксплуатации, приемо-сдаточных испытаний комплексов задач; по обеспечению перехода от существующих методов управления к методам, предусмотренным проектом АСУ.

Обязанности заказчика:

1. Завершить выполнение организационно-технических мероприятий по подготовке предприятия к внедрению и оформить их актами.
2. Обеспечить выполнение персоналом предприятия должностных и технологических инструкций.
3. Ввести в эксплуатацию технические средства, необходимые для внедряемого технологического процесса обработки данных.
4. Издать приказ с планом-графиком о проведении опытной эксплуатации АСУ и проанализировать совместно с разработчиком результаты опытной эксплуатации.
5. Завершить опытную эксплуатацию комплексов задач АСУ и приемку их в промышленную эксплуатацию.
6. Внести изменения в организационную структуру предприятия в соответствии с проектом АСУ.
7. Разработать проект приказа по составу приемочной комиссии; разработать и согласовать с разработчиком проект программы приемо-сдаточных испытаний; организовать работу приемочной комиссии, представить ей требуемую документацию, провести испытания АСУ.
8. Проверить эффективность реализованных решений в условиях промышленной эксплуатации, по результатам анализа функционирования системы разработать рекомендации по ее дальнейшему развитию.

Обязанности разработчика:

1. Корректировать техническую документацию по результатам опытной эксплуатации АСУ.
2. Принимать участие в разработке проекта программы приемо-сдаточных испытаний АСУ.
3. Осуществлять методическое руководство и принимать участие в сдаче задач в промышленную эксплуатацию.
4. Участвовать в работе комиссии по приемке АСУ в промышленную эксплуатацию.

Для АСУ устанавливают следующие основные виды испытаний: предварительные, опытная эксплуатация и приемочные. При необходимости допускается дополнительно проведение других видов испытаний системы и ее частей.

Предварительные испытания проводят для определения работоспособности системы и решения вопроса о возможности ее приемки в опытную эксплуатацию. Предварительные испытания следует выполнять после проведения разработчиком отладки и тестирования поставляемых программных и технических средств системы и представления им соответствующих документов об их готовности к испытаниям, а также после ознакомления персонала ИС с эксплуатационной документацией.

Опытную эксплуатацию системы проводят с целью определения фактических значений количественных и качественных характеристик системы и готовности персонала к работе в условиях ее функционирования, а также определения фактической эффективности и корректировки, при необходимости, документации.

Приемочные испытания проводят для определения соответствия системы техническому заданию, оценки качества опытной эксплуатации и решения вопроса о возможности приемки системы в постоянную эксплуатацию.

Приемочная комиссия обеспечивает: проверку документации и функционирования АСУ в сроки, определенные приказом министерства (ведомства), назначившего комиссию; организацию рабочих групп и распределение обязанностей среди членов комиссии для проверки отдельных подсистем; проверку расчета экономической эффективности АСУ и научно-технического уровня; организацию рабочих совещаний и подготовку актов приемки АСУ.

Завершающим этапом работы комиссии является составление акта, в котором указываются: состав комиссии, должности и места работы членов комиссии; срок приемки; состав исполнителей, принимавших участие в создании АСУ; основания для проведения приемки и т.д.

Дата подписания акта комиссией считается датой ввода АСУ в эксплуатацию.

На этапе сопровождения проекта выполняются два вида работ: ликвидируются последствия сбоев в работе системы и исправляются ошибки, не выявленные при внедрении проекта, а также осуществляется модернизация проекта. В процессе модернизации проект либо дорабатывается, т.е. расширяется по составу подсистем и задач, либо производится перенос системы на другую программную или техническую платформу с целью адаптации ее к изменяющимся внешним и внутренним условиям функционирования.

Ключевые вопросы:

1. Подготовка объекта к вводу.
2. Предварительные испытания.
3. Опытная эксплуатация.
4. Приемочные испытания.
5. Сопровождение АСОИУ.

Литература:

1. Гагарина Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. пособие: рек. Мин. обр. РФ/ Л. Г. Гагарина, Д. В. Киселев, Е. Л. Федотова; под ред. Л. Г. Гагариной. -М.: ФОРУМ: ИНФРА-М, 2007.-384 с.
2. Бушманов А.В. Проектирование информационных систем: курс лекций: рек. ДВ РУМЦ/ А.В. Бушманов; АмГУ, ИФФ. -Благовещенск: Изд-во Амур. гос.ун-та, 2008.-112 с.

3. Методические материалы к выполнению лабораторных работ

Лабораторная работа 1. Создание диаграммы DFD.

Диаграммы потоков данных (DFD) используются для описания документооборота и обработки информации. Подобно IDEF0, DFD представляет систему как сеть связанных между собой функций. Их можно использовать как дополнение к модели IDEF0 для более наглядного отображения текущих операций документооборота в системах обработки информации.

Источники информации - внешние сущности - порождают информационные потоки данных, переносящие информацию к процессам. Те в свою очередь преобразуют информацию и порождают новые потоки, которые переносят информацию к другим процессам, хранилищам данных или внешним сущностям - потребителям информации.

Внешняя сущность - представляет собой материальный объект или физическое лицо (заказчик, склад), представляющее собой источник или приемник информации, который находится за пределами границ анализируемой системы. Обозначается в виде прямоугольника с тенью и обычно располагаются по краям диаграммы. Одна внешняя сущность может быть использована многократно на одной диаграмме, чтобы не рисовать слишком запутанных стрелок.

Процесс - представляет собой функции системы, преобразующие входные потоки в выходные в соответствии с определенным алгоритмом. Изображаются прямоугольниками со скругленными углами. Их смысл совпадает со смыслом функций IDEF0. Они также имеют входы и выходы, но не поддерживают управления и механизмы. Физически процесс - это например, отдел, выполняющее обработку входных документов и выпуск отчетов, или программа, аппаратно реализованное логическое устройство.

Поток данных - определяет информацию, передаваемую через некоторое соединение от источника к приемнику. Реальный поток данных может быть информацией передаваемой по кабелю, пересылаемыми по почте письмами, магнитными дискетами. Поскольку в DFD каждая сторона блока не имеет четкого назначения, как в IDEF0, стрелки могут подходить и выходить из любой грани блока. В DFD также применяется двунаправленные стрелки для описания диалогов типа «команда-ответ».

Хранилище данных - позволяет описать данные, которые необходимо сохранить в памяти прежде, чем использовать в функциях. То есть в отличие от стрелок, описывающих объекты в движении, хранилища изображают объекты в покое. Хранилище физически может быть реализован в виде ящика в картотеки, таблицы в оперативной памяти, файла на магнитном диске.

Рассмотрим фрагмент проекта системы, организующей работу банкомата по обслуживанию клиента по его кредитной карте.

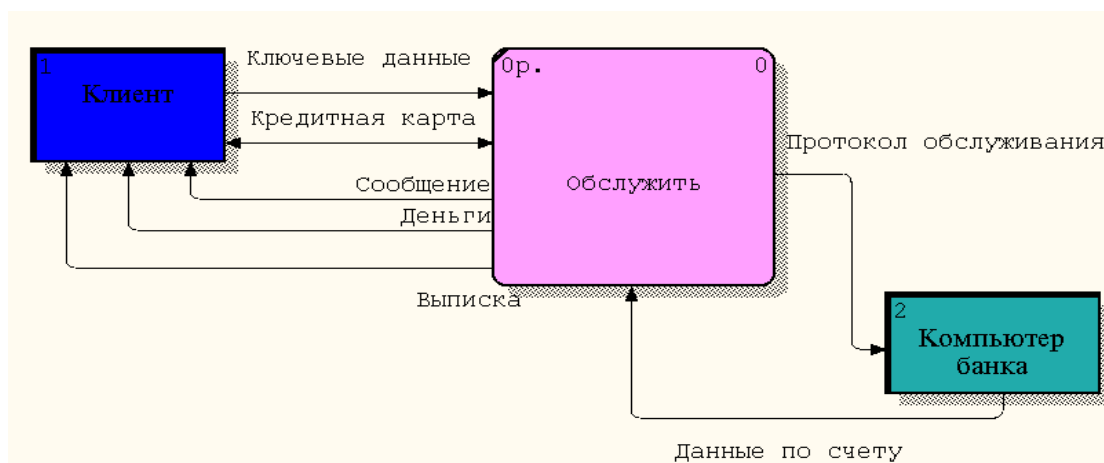
Запустите CASE средство:

Пуск | Программы | Computer Associates| AllFusion | Process Modeler | BPwin

Создайте новую модель: выберите **File | New**, в поле **Name** введите **Работа банкомата**, в области **Type** выберите **Data Flow(DFD)**, нажмите **OK**, в поле **Author** введите ваше имя, например, **Наташа**, нажмите **OK**.

Установите русский шрифт для объекта: выберите **Model | Default Fonts**, выберите **Context Activity**, выберите **Courier New** или **TimesET**, нажмите **OK**

Внимание! Установите русский шрифт для **всех** типов объектов имеющих в меню **Model | Default Fonts** - см.



Введите имя процесса и описание: выберите процесс, нажмите **МП**, выберите **Name**, введите **Обслужить**. выберите вкладку **Defenition**, в поле **Defenition** введите определение **Процесс обслуживания клиента по его кредитной карте с помощью банкомата**, выберите вкладку **Color**, выберите **розовый**, нажмите **ОК**

Добавьте внешнюю сущность: нажмите кнопку **External Reference Tool**, выберите мышью место размещения - в верхнем левом углу, введите имя сущности **Клиент**, нажмите **ОК**, нажмите кнопку **Poiter Tool**.

Введите описание: выберите внешнюю сущность **Клиент**, нажмите **МП**, выберите **Name**, выберите вкладку **Defenition**, в поле **Defenition** введите определение **Клиент банка с кредитной картой**, выберите вкладку **Color**, выберите **синий**, нажмите **ОК**.

Добавьте внешнюю сущность **Компьютер банка** в нижнем правом углу и ее описание **Хранит информацию о счетах всех клиентов**- см. 32-33.

35. Введите поток данных: нажмите кнопку **Precedence Arrow Tool**, выберите правую грань внешней сущности **Клиент**, выберите левую грань процесса **Обслужить**, нажмите кнопку **Poiter Tool**.

Введите имя и тип потока данных: выберите созданный поток данных, нажмите **МП**, выберите **Name**, в поле **Arrow Name** введите **Кредитная карта**, выберите вкладку **Style**, в области **Shape** выберите **двунаправленную** стрелку, нажмите **ОК**,

Введите описание потока данных: выберите поток данных **Кредитная карта**, нажмите **МП**, выберите вкладку **Definition**, в поле **Definition** введите **Для банковского обслуживания клиент должен предоставить кредитную карту для автоматического считывания с нее информации (пароль, лимит денег, детали клиента)**, нажмите **ОК**

Введите поток данных **Ключевые данные** и его описание **Для банковского обслуживания клиент должен ввести ключевые данные - пароль, запрос на обслуживание**

Банковское «обслуживание» должно: (1) выдать **сообщение**, приглашающее клиента ввести **ключевые данные**, (2) выдать клиенту **деньги**, (3) выдать клиенту **выписку** по проведенному обслуживанию, включающую **выписку о деньгах, выписку по балансу, выписку по операции**, проведенной банком.

Введите потоки данных **Сообщение, Деньги, Выписка** - см. **Рис.1**.

Введите описание потоков **Сообщение, Деньги, Выписка**.

Процесс «Обслужить» и внешняя сущность «Компьютер банка» обмениваются следующей информацией: (1) **данные по счету** клиента в банке, (2) **протокол обслуживания**, включающий информацию об **обработанной документации**, **изымаемой денежной сумме** и **данные по истории запроса**.

Введите потоки данных **Данные по счету, Протокол обслуживания** - см. **Рис.1**.

Введите описание потоков **Данные по счету, Протокол обслуживания**.

Создайте диаграмму декомпозиции: выберите процесс **Обслужить**, выберите на палитре инструментов кнопку декомпозиции, в окне **Activity Box Count** выберите **DFD**, в списке **Number of Activities...** выберите **4**, выберите **ОК**.

Добавьте хранилище данных: нажмите кнопку **Date store tool**, введите имя хранилища **Данные кредитной карты** - см. **Рис.2**

Добавьте еще раз хранилище данных **Данные кредитной карты** с целью избежать пересечений линий потоков данных.

Процесс **Получить пароль** осуществляет прием и проверку пароля клиента и имеет потоки: (1) внешний выходной поток **Сообщение** для информирования клиента о своей готовности принять пароль, (2) входной поток **Введенный пароль** как элемент внешнего потока **Ключевые данные**, (3) входной поток **Пароль** из хранилища **Данные кредитной карты** для проверки вводимого клиентом пароля.

Введите имя процесса **Получить пароль** - см. **Рис.2**

Введите потоки данных **Сообщение, Введенный пароль, Пароль** - см. **Рис.2**.

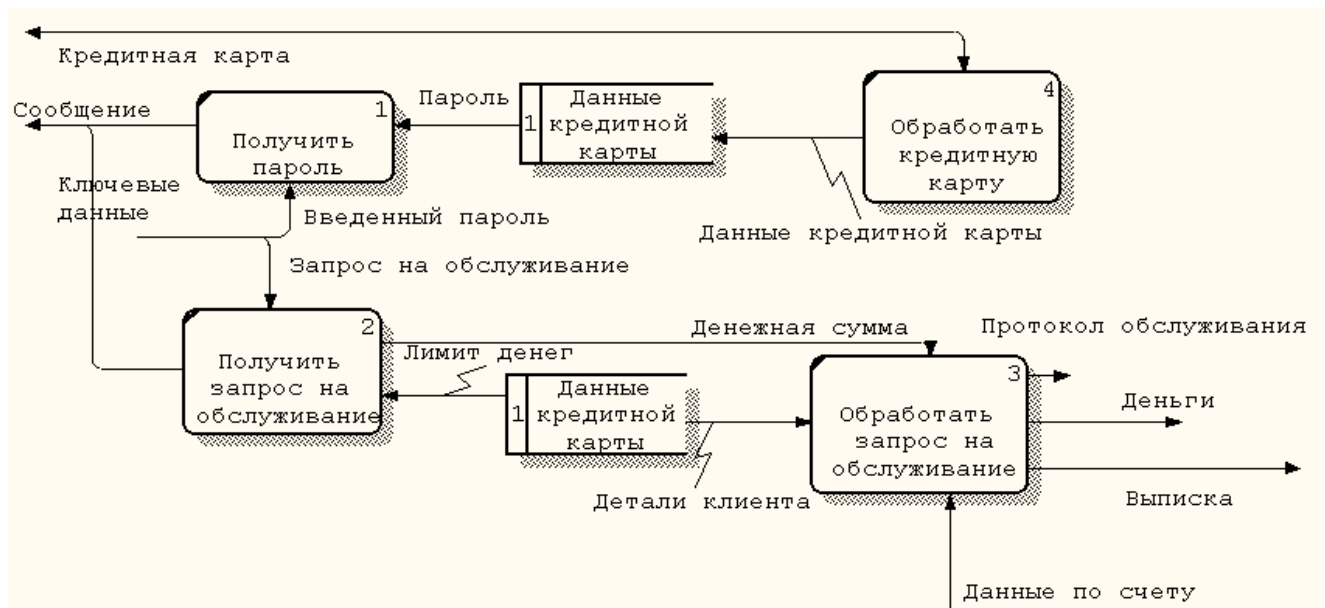
Введите описание процесса **Получить пароль** и потоков данных **Сообщение**, **Введенный пароль**, **Пароль**.

Процесс **Получить запрос на обслуживание** осуществляет прием и проверку запроса клиента на проведение необходимой ему банковской операции и имеет на входе-выходе следующие процессы: (1) внешний выходной поток **Сообщение** для информирования клиента о своей готовности принять запрос на обслуживание, (2) входной поток **Запрос на обслуживание** как элемент внешнего потока **Ключевые данные**, (3) входной поток **Лимит денег** из хранилища **Данные кредитной карты** для контроля наличия денег на счете клиента.

Введите имя процесса **Получить запрос на обслуживание** - см. **Рис. 2**

Введите потоки данных **Сообщение**, **Запрос на обслуживание**, **Лимит денег** - см. **Рис.2**

Введите описание процесса **Получить запрос на обслуживание** и потоков данных **Сообщение**, **Запрос на обслуживание**, **Лимит денег**.



Процесс **Обработать запрос на обслуживание** имеет (1) внешний входной поток **Данные по счету** (из сущности **Компьютер банка**), (2) входной поток **Детали клиента**, (3) входной поток **Денежная сумма** (из процесса **Получить запрос на обслуживание**), (4) внешние выходные потоки **Выписка**, **Деньги**, **Протокол обслуживания**.

Введите имя процесса **Обработать запрос на обслуживание** - см. **Рис.2**

Введите потоки данных **Данные по счету**, **Детали клиента**, **Выписка**, **Деньги**, **Протокол обслуживания**, **Денежная сумма** - см. **Рис. 2**.

Введите описание процесса **Обработать запрос на обслуживание** и потоков данных **Данные по счету**, **Детали клиента**, **Выписка**, **Деньги**, **Протокол обслуживания**, **Денежная сумма**.

Процесс **Обработать кредитную карту** осуществляет считывание информации с кредитной карты и имеет на входе внешний поток **Кредитная карта**, на выходе - поток **Данные кредитной карты**.

Введите имя процесса **Обработать кредитную карту** - см. **Рис.2**

Введите потоки данных **Кредитная карта**, **Данные кредитной карты** - см. **Рис.2**.

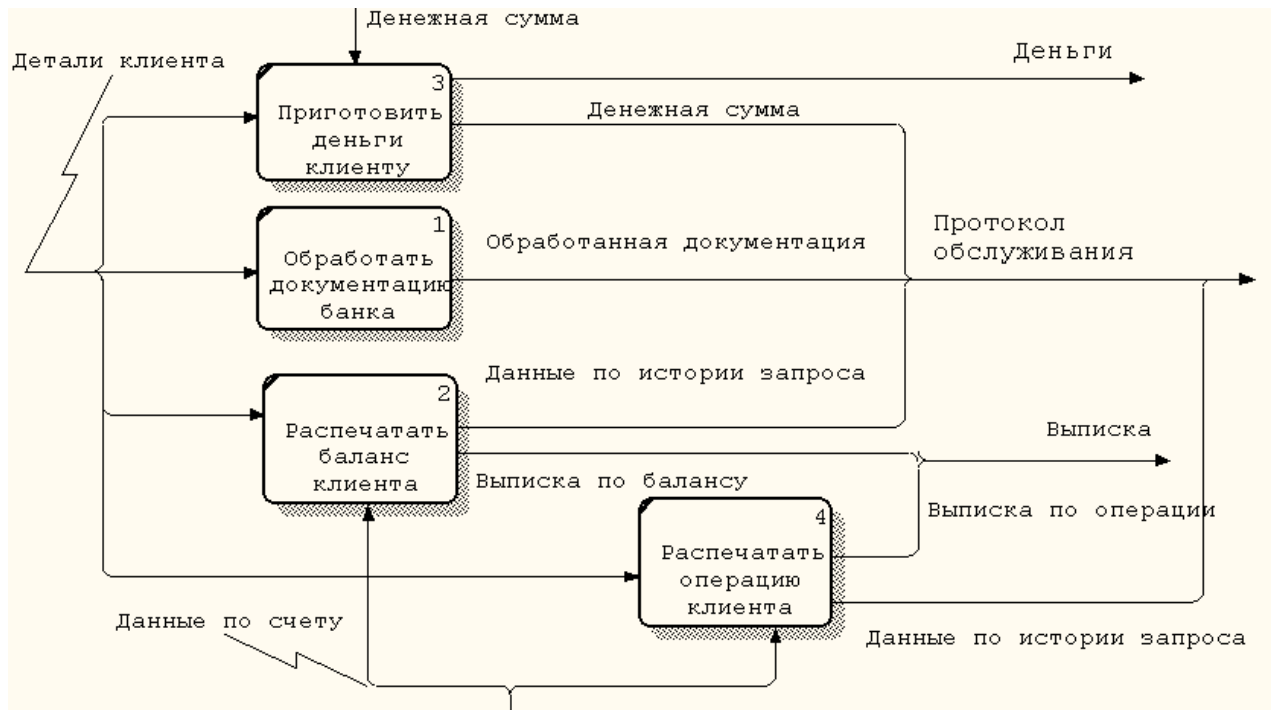
Введите описание процесса **Обработать кредитную карту** и потоков данных **Кредитная карта**, **Данные кредитной карты**.

Создайте диаграмму декомпозиции процесса **Обработать запрос на обслуживание** - см. **43**:

Процесс **Обработать документацию банка** выполняет обработку внутренней банковской документации по клиенту и имеет (1) входной поток **Детали клиента**, (2) выходной поток **Обработанная документация** (часть внешнего потока Протокол обслуживания).

Введите имя процесса **Обработать документацию банка**.

Введите потоки данных **Детали клиента, Обработанная документация** - см. **Рис.3**.



Введите описание процесса **Обработать документацию банка** и потоков данных **Детали клиента, Обработанная документация**.

Процесс **Распечатать баланс клиента** выдает справку по истории счета клиента и по балансу клиента и имеет (1) входные потоки **Детали клиента** и **Данные по балансу** (часть внешнего потока Данные по счету), (2) выходные потоки **Выписка по балансу** (часть внешнего потока Выписка) и **Данные по истории запроса** (часть внешнего потока Протокол обслуживания).

Введите имя процесса **Распечатать баланс клиента**.

Введите потоки данных **Детали клиента, Данные по балансу, Выписка по балансу, Данные по истории запроса** - см. **Рис.3**.

Введите описание процесса **Распечатать баланс клиента** и потоков данных **Детали клиента, Данные по балансу, Выписка по балансу, Данные по истории запроса**.

Процесс **Приготовить деньги клиенту** обеспечивает выдачу наличных денег и информирование компьютера банка об изъятии из банка денег. Процесс имеет (1) входные потоки **Денежная сумма** и **Детали клиента**, (2) выходные потоки **Деньги** и **Денежная сумма** (часть потока Протокол обслуживания).

Введите имя процесса **Приготовить деньги клиенту**.

Введите потоки данных **Денежная сумма, Детали клиента, Деньги, Денежная сумма** - см. **Рис.3**.

Введите описание процесса **Приготовить деньги клиенту** и потоков данных **Денежная сумма, Детали клиента, Деньги, Денежная сумма**.

Процесс **Распечатать операцию клиента** выдает справку по истории счета клиента и уведомление по проведенной операции и имеет потоки (1) входные - **Данные по счету** и **Детали клиента**, (2) выходные - **Выписка по операции** (часть потока Выписка) и **Данные по истории запроса** (часть потока Протокол обслуживания).

Введите имя процесса **Распечатать операцию клиента**.

Введите **Данные по счету, Детали клиента, Выписка по операции, Данные по истории запроса** - см. **Рис.3**.

Введите описание процесса **Распечатать операцию клиента** и потоков данных **Данные по счету, Детали клиента, Выписка по операции, Данные по истории запроса**.

Лабораторная работа 2-3. Решение поставленной задачи при помощи технологии IDEF0.

Лабораторная работа 4. Создание диаграммы IDEF3.

Лабораторная работа 5-6. Взаимосвязь моделей IDEF0 и IDEF3.

Лабораторная работа 7-8. Два подхода к построению DFD-моделей.

Лабораторная работа 9-10. Создание модели TO-BE.

Лабораторная работа 11-12. Моделирование управленческого учета на предприятии.

Лабораторная работа 13-14. Введение в CASE-пакет Rational Rose.

Лабораторная работа 15. Диаграммы вариантов использования.

Лабораторная работа 16. Диаграммы классов.

Лабораторная работа 17. Диаграммы взаимодействия.

Лабораторная работа 18. Диаграммы состояний.

Лабораторная работа 19. Диаграммы пакетов, компонентов, размещения.

Лабораторная работа 20-21. Генерация исходных текстов программ.

Лабораторная работа 22. Обратное проектирование.

4. Методические материалы к выполнению курсового проекта

Настоящие методические указания содержат основные требования и рекомендации по организации, выполнению и защите курсовых проектов (работ) по направлению 654600 - "Информатика и вычислительная техника" по специальности 230102 - "Автоматизированные системы обработки информации и управления".

В основу этих указаний положены "Инструкция по подготовке курсовых проектов (работ) в высших учебных заведениях" и квалификационная характеристика дипломированного специалиста по направлению 654600 "Информатика и вычислительная техника" по специальности 230102 - "Автоматизированные системы обработки информации и управления".

Общие положения.

Курсовой проект выполняется при обучении в девятом семестре в соответствии с установленным учебным планом.

Целью выполнения курсового проекта является формирование у студентов навыков:

- самостоятельной научно-исследовательской деятельности;
- практической деятельности;
- грамотного оформления полученных результатов в печатном виде;
- представления результатов своей работы в виде научного доклада;
- защиты полученных результатов в дискуссии.

Выбор темы работы.

Тематика курсовых проектов определяется преподавателями кафедр, осуществляющих руководство научной работой студентов. Перечень предлагаемых тем (названий) работ с указанием научного руководителя доводится до сведения студентов в течение первых трех недель текущего учебного года.

Студент самостоятельно выбирает научного руководителя и тему работы в соответствии со своими интересами, о чем лично сообщает выбранному им научному руководителю. В ходе предварительного обсуждения выбранной темы с научным руководителем и в процессе выполнения работы тема может быть изменена по согласованию между научным руководителем и студентом.

Выбор темы должен быть сделан студентом в течение четвертой недели текущего учебного года.

Проект выполняется в течение семестра и может быть продолжением ранее начатого исследования или развитием результатов, полученных студентом в течение предшествующих лет обучения.

Содержание курсового проекта.

Курсовой проект, как правило, представляет собой:

1. Исследование актуальной задачи по специальности;
2. Разработку информационной системы (ИС) или иного программного продукта.

Исследование задачи включает следующие разделы:

1. Описание (постановка) задачи.
2. Обоснование актуальности задачи.
3. Обзор информации, содержащейся в открытых источниках, посвященных данной задаче или области исследований.
4. Исследование задачи:
 - а) классификация задачи, т. е. отнесение ее к некоторому известному классу задач;
 - б) описание известных методов решения задач этого класса;
 - в) описание особенностей исследуемой задачи, ее отличительных черт, которые не позволяют применять существующие методы в стандартном виде;
 - г) предложения по модификации существующих методов для решения задачи (ближних задач) или по модификации самой задачи для применения существующих методов;
 - д) описание предлагаемых методов решения или подходов к решению с обоснованием их применимости к данной задаче;
 - е) описание возникающих в процессе решения проблем или других побочных, вспомогательных или параллельных задач.

Разработка информационной (программной) системы включает следующие разделы:

1. Постановка задачи:
 - а) описание предметной области, например, бизнес-процессов, протекающих в предметной области;
 - б) описание информационных потоков;
 - в) описание процессов обработки информации, управления и т. п., требующих автоматизации.
2. Обзор существующих программных продуктов, выполняющих аналогичные функции. Их достоинства и недостатки. Сравнение, классификация.
3. Обоснование необходимости разработки.
4. Описание математической задачи:
 - а) постановка задачи;
 - б) исследование задачи;
 - в) математическая модель;
 - г) метод решения возникающей математической задачи;

- д) алгоритм, реализующий метод решения.
5. Проект АСОИУ:
- а) архитектура, аппаратная платформа, ОС, СУБД, состав других программных продуктов;
 - б) описание структуры БД: таблицы, индексы, процедуры, триггеры, события, запросы;
 - в) описание логики программ и интерфейсов;
 - г) алгоритмы обработки данных.
6. Реализация АСОИУ:
- а) назначение и функции программы, режимы работы программы;
 - б) описание категорий пользователей программы, разграничения прав пользователей;
 - в) описание последовательности пользовательских интерфейсов, реализующих каждую функцию АИС;
 - г) описание входных данных;
 - д) описание выходных данных;
 - е) описание методов защиты данных в ИС и информационных систем в целом;
 - ж) технические характеристики ИС (тип и минимально необходимые аппаратные ресурсы вычислительной системы, требуемое программное обеспечение и т. п.).
7. Анализ:
- а) внедрение;
 - б) области применения АИС;
 - в) достоинства и недостатки по сравнению с перечисленными ранее в разделе 2 аналогами;
 - г) описание возникших в процессе разработки проблем, их причины, предложения по решению;
 - д) направления дальнейшего развития АИС.
- Реферативная работа включает следующие разделы:
1. Обоснование актуальности выбранной тематики и описание целей выполнения работы.
 2. Систематизация и анализ найденных в научной печати, в сети Интернет и других источниках материалов.
 3. Выводы.
 4. Предложения по использованию результатов работы в конкретных областях и возможные направления дальнейших исследований.

Порядок выполнения курсового проекта.

Курсовой проект выполняется в течение девятого семестра.

По итогам работы, до начала сессии, студент в установленный кафедрой срок представляет подготовленный в печатном виде материал и по решению кафедры делает сообщение по теме курсового проекта на семинаре. Руководитель на титульном листе выставляет оценку, заверяя ее своей подписью.

На титульном листе фиксируется срок представления отчета. В случае несоблюдения сроков представления или низкого качества отчета, его оценка снижается, как минимум, на один балл.

Студенты, не представившие в срок курсовые проекты, не допускаются к публичной защите. В этом случае защита переносится на начало сессии.

Оценка за курсовую работу складывается из следующих оценок:

- оценки руководителя;
- оценки публичной защиты;
- оценки оформления.

При выставлении оценки учитываются также сроки представления печатного варианта курсового проекта.

Выступление с докладом по промежуточным результатам курсового проекта на студенческой научной конференции может быть засчитано с выставлением заслуженной оценки (отлично, хорошо, удовлетворительно).

В случае отсутствия научного руководителя в период представления курсового проекта (командировка, болезнь и т. п.), курсовой проект сдается на кафедру в установленные сроки. Оценка руководителя выставляется позднее.

Структура курсового проекта.

Работа начинается с титульного листа стандартной формы, за которым следует лист задания, оглавление работы, введение, нескольких разделов, заключения, списка использованных научных источников, приложений.

Введение содержит общий обзор работы, позволяющий составить общее представление об исследуемой проблеме и полученных результатах. Во введении может быть предложена краткая аннотация отдельных разделов работы. Первый раздел должен содержать достаточно подробное описание проблемы, поставленной перед исполнителем с обоснованием ее актуальности и анализ современного состояния исследований и разработок в данной области.

В последующих разделах, число которых произвольно, описываются результаты, полученные по отдельным аспектам исследуемой проблемы. Каждый раздел может разбиваться на подразделы.

Заключение содержит перечень основных результатов, полученных в работе, и сделанных выводов. В него могут включаться рекомендации относительно перспектив продолжения данной работы.

В списке использованных источников указываются использованные автором работы научные публикации, а также другие источники, в том числе электронные, по проблемам разработки аналогичных систем, по средствам разработки, по методам решения математических задач. На все перечисленные в списке литературы источники в соответствующих местах работы должны быть сделаны ссылки (номер источника заключается в квадратные скобки).

Список использованных источников должен содержать не менее 10 печатных изданий и любого количества непечатных изданий.

Приложения могут содержать дополнительную информацию: графики, таблицы, тексты программ и т. п.

Оформление курсового проекта.

Курсовой проект представляется на кафедру в полностью готовом виде (сшитом, в переплете или в обложке).

Дополнительно к проекту прилагаются специальные (магнитные или иные) носители информации, содержащие программы (тексты и исполняемые файлы), данные или объемные приложения, включение которых в текст работы является нецелесообразным.

Текст курсовой работы оформляется в принятом для научных работ виде.

На странице располагается 30 строк., в строке 60 знаков, включая пробелы.

Следует соблюдать следующие размеры полей: левое – не менее 30мм, правое – не менее 10 мм, верхнее – не менее 15 мм, нижнее – не менее 20 мм.

Нумерация страниц выполняется арабскими цифрами, начиная с титульного листа. На титульном листе, а также на первой странице оглавления номер не ставится. На следующих страницах номер ставят в правом верхнем углу.

Работа должна быть отпечатана. Рукописные работы не принимаются.

Графический материал.

Графический материал проекта должен состоять из чертежей, выполненных в соответствии с требованиями ЕСКД, ЕСПД и других действующих ГОСТов. Чертежи должны быть

выполнены или на фолиях или в мультимедийном исполнении и снабжены штампом. Каждый чертеж должен быть снабжен основной надписью, графы которой должны быть заполнены в соответствии с образцом.

Макеты чертежей перед окончательным оформлением необходимо показать руководителю проекта.

К основным ГОСТам, которые необходимо использовать при оформлении курсового проекта, приведенные в работе относятся:

ГОСТ 34.201-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем.

ГОСТ 34.602-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы. с. 106-116.

РД 50-682-89 Методические указания. Информационная технология Комплекс стандартов и руководящих документов на автоматизированные системы. Общие положения. с. 157-161.

РД 50-680-88 Методические указания. Автоматизированные системы. Основные положения. с. 152-156.

ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания. с. 100-105.

ГОСТ 34.401-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Средства технические периферийные автоматизированных систем. Типы и технические требования.

РД 50-34.698-90 Методические указания. Информационная технология. Комплекс стандартов и руководящих документов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов. с. 127-151.

ГОСТ 34.003-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения.

Р50-34.119-90 Рекомендации. Информационная технология. Комплекс стандартов на автоматизированные системы. Архитектура локальных вычислительных сетей в системах промышленной автоматизации. Общие положения.

ГОСТ 24.301-80* Система технической документации на АСУ. Общие требования к выполнению текстовых документов.

ГОСТ 24.302-80 Система технической документации на АСУ. Общие требования к выполнению схем. с. 22-24.

ГОСТ 24.303-80 Система технической документации на АСУ. Обозначения условные графические технических средств. с.25-31.

ГОСТ 24.304-80 Система технической документации на АСУ. Требования к выполнению чертежей с.32-34.

ГОСТ 19.401-78* Единая система программной документации. Текст программы. Требования к содержанию и оформлению.

ГОСТ 19.402-78* Единая система программной документации Описание программы.

Кроме того, графический материал может быть представлен в виде фолий и мультимедийных разработок.

Защита курсового проекта.

Курсовой проект, подписанный исполнителем, представляется на проверку руководителю за 3-4 дня до назначенного срока защиты.

Защита проекта проводится индивидуально каждым студентом на заседании комиссии, состав которой утверждается кафедрой, с обязательным участием руководителя курсового проекта.

Студент, выполнивший курсовой проект, делает доклад (5-7 минут) и отвечает на вопросы комиссии.

В докладе студенту необходимо изложить важнейшие этапы, особенности и результаты работы, не углубляясь в тонкости конкретных технических решений, четко сформулировать конечные результаты.

Вопросы, задаваемые студенту, могут касаться деталей выполненного проекта, а также разделов курсов, по которым выполнялся проект.

Результаты защиты курсового проекта определяются оценками "отлично", "хорошо", "удовлетворительно", "неудовлетворительно", которые проставляются на титульном листе пояснительной записки. При оценке работы учитывается качество выполнения и оформления курсового проекта, уровень защиты проекта и ответов на вопросы, мнение руководителя.