

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

**УТВЕРЖДАЮ**

Заведующий кафедрой  
ПиИТ



проф. Махортов С.Д.  
10.03.2021

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**  
Б1.В.ДВ.04.02 Современные технологии программирования (Delphi, Fortran)

**1. Код и наименование направления подготовки/специальности:**

02.03.01 Математика и компьютерные науки

**2. Профиль подготовки/специализация:**

Квантовая теория информации

**3. Квалификация выпускника: бакалавр**

**4. Форма обучения: очная**

**5. Кафедра, отвечающая за реализацию дисциплины: программирования и информационных технологий**

**6. Составители программы: Хлебостроев Виктор Григорьевич, к.ф-м.н., доцент**

**7. Рекомендована: НМС ФКН, протокол № 5 от 10.03.2021 г.**

---

*(наименование рекомендующей структуры, дата, номер протокола,*

---

*отметки о продлении вносятся вручную)*

**8. Учебный год: 2021/2022, 2022/2023**

**Семестр(ы): 2, 3, 4**

## 9. Цели и задачи учебной дисциплины

- Изучение студентами современных технологий разработки корпоративных информационных систем

- овладение практическими навыками создания сложных программных комплексов

## 10. Место учебной дисциплины в структуре ООП: Обязательная часть

## 11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
ПК-1	Способен самостоятельно создавать прикладные программные средства на основе современных информационных технологий и сетевых ресурсов, в том числе отечественного производства	ПК-1.1	Обладает фундаментальным и знаниями в области прикладного программирования и информационных технологий	Знать: принципы определения требований Уметь: собирать требования с заказчика в рамках установленной предметной области Владеть: подходами для систематизации требований
		ПК-1.2	Умеет использовать их в профессиональной деятельности	Знать: средства для коллективной работы над требованиями Уметь: пользоваться средствами для коллективной работы над требованиями Владеть: методами распределения ресурсов для реализации поставленных задач
		ПК-1.3	Имеет практический опыт применения программных средств, используемых при построении математических моделей в естественных науках.	Знать: современные стандарты и технологии, применяемые для проектирования ИС Уметь: выполнять проектирование ИС Владеть: подходами для проектирования ИС
ПК-4	Способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования.	ПК-4.1	Владеет современными методами разработки и реализации алгоритмов математических моделей на базе языков и пакетов программ моделирования	Знать: методы прототипирования ИС Уметь: создавать прототипы ИС Владеть: инструментальными средствами для прототипирования
		ПК-4.2	Умеет разрабатывать и реализовывать алгоритмы математических	Знать: методы оценки ИС Уметь: оценивать ИС по различным критериям Владеть: подходами для выбора наилучшего варианта ИС из предложенных

			моделей на базе языков и пакетов прикладных программ моделирования	
		ПК-4.3	Имеет практический опыт разработки и реализации алгоритмов на базе языков и пакетов прикладных программ моделирования	Знать: методы оценки ИС Уметь: оценивать ИС по различным критериям Владеть: подходами для выбора наилучшего варианта ИС из предложенных
ПК-5	Способен использовать современные методы разработки и реализации конкретных математических моделей на базе языков программирования и пакетов прикладных программ моделирования.	ПК-5.1	Владеет современными методами разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	Знать: методы прототипирования ИС Уметь: создавать прототипы ИС Владеть: инструментальными средствами для прототипирования
		ПК-5.2	Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Знать: методы оценки ИС Уметь: оценивать ИС по различным критериям Владеть: подходами для выбора наилучшего варианта ИС из предложенных
		ПК-5.3	Имеет практический опыт разработки и реализации алгоритмов на базе языков и пакетов прикладных программ моделирования	Знать: методы оценки ИС Уметь: оценивать ИС по различным критериям Владеть: подходами для выбора наилучшего варианта ИС из предложенных

**12. Объем дисциплины в зачетных единицах/час — 10/360**

**Форма промежуточной аттестации зачет с оценкой, экзамен**

**13. Трудоемкость по видам учебной работы**

Вид учебной работы	Трудоемкость			
	Всего	По семестрам		
		2	3	4
Аудиторные занятия	128	64	32	32
в том числе: лекции	64	32	16	16

практические	32	16	16	
лабораторные	32	16		16
Самостоятельная работа	196	116	40	40
Форма промежуточной аттестации (зачет – 0 час. / экзамен – __ час.)	36			36
Итого:	360	180	72	108

### 13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
1	Сортировки и поиск	Сортировка простым выбором. Сортировка включениями. Сортировка простыми включениями. Сортировка бинарными включениями. Сортировка обменом. Сортировка простым обменом. Шейкер-сортировка. Сортировка Шелла. Сортировка подсчетом. Поиск в неупорядоченном массиве. Поиск с барьером. Поиск в упорядоченном массиве. Бинарный поиск. Хеширование
2	Динамические структуры данных	Указатели. Физическая структура указателя. Представление указателей в языках программирования. Операции над указателями. Действия с указателями. Операторы для указателей (pointer). Динамические переменные. Динамические структуры данных. Указатели и ссылки. Линейные списки. Основные операции. Списки, стеки, очереди. Упорядоченный список. Частотный словарь. Слияние двух упорядоченных списков. Двусвязный список. Кольцевой список.
3	Списки, стеки, очереди	Стеки. Динамическая реализация стека. Стек, реализованный с помощью массива. Очереди. Динамическая реализация очереди. Очередь, реализованная с помощью массива
4	Рекурсия	Рекурсивные определения и рекурсивные алгоритмы. Когда рекурсия необходима. Примеры рекурсивных программ. "Ханойские башни". Быстрая сортировка. Алгоритмы с возвратом. Расстановка ферзей. Задача оптимального выбора
5	Нуль-терминированные строки. Процедурные типы	Нуль-терминированные строки. Процедурные типы
6	Алгоритмы на деревьях	Понятия и определения. Основные операции с бинарными деревьями. Упорядоченные деревья. Поиск по дереву с включением. Поиск по дереву с включением. Удаление из упорядоченного дерева. Сбалансированные деревья. Включение в сбалансированное дерево. Удаление из сбалансированного дерева. Сортировки на деревьях. Турнирная сортировка. Сортировка частично упорядоченным деревом. Основы работы интерпретатора. Интерпретатор для простых операторов
7	Алгоритмы на графах	Проект для алгоритмов на графах. Структура данных. Изображение графов. Чтение и запись графов. Поиск в графах. Поиск в глубину. Поиск в ширину. Остов графа. Кратчайшие пути. Волновой алгоритм. Алгоритм Дейкстры. Алгоритм Форда-Мура-Беллмана. Алгоритм Флойда. Циклы на графах. Циклы на графах. Эйлеровы циклы. Гамильтонов цикл. Алгоритмы с возвратом. Гамильтоновы циклы и задача коммивояжера. Метод перебора. Кратчайший незамкнутый путь. Алгоритм штрафов вершин. Комбинаторные задачи на графах. Минимальная раскраска графа. Переборный алгоритм. Приближенные алгоритмы раскраски графа, основанные на понятии соцветных вершин. Алгоритмы о связности графа. Топологическая сортировка. Остовное дерево наименьшей стоимости. Алгоритмы Прима и Краскала. Выделение компонент связности

## 13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование раздела дисциплины	Виды занятий (часов)				Всего
		Лекции	Лабораторные	Практические	Самостоятельная работа	
1	Сортировки и поиск	10	4	4	28	46
2	Динамические структуры данных	10	4	4	28	46
3	Списки, стеки, очереди	10	4	4	28	46
4	Рекурсия	6	4	4	28	42
5	Нуль-терминированные строки. Процедурные типы	8	4	4	28	44
6	Алгоритмы на деревьях	10	6	6	28	50
7	Алгоритмы на графах	10	6	6	28	50
Итого:		64	32	32	196	360

## 14. Методические указания для обучающихся по освоению дисциплины

Рекомендуется работа с конспектами лекций, презентационным материалом, выполнение 3 лабораторных работ. Учебные и методические материалы по дисциплине представлены в локальной сети факультета.

## 15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины

### а) основная литература:

№ п/п	Источник
1	Тюкачев Н.А. Программирование в Delphi для начинающих: учебное пособие для студ. вузов, обуч. по специальности 351400 "Прикладная информатика (по областям)" и другим специальностям / Н.А. Тюкачев, К.С. Рыбак, Е.М. Михайлова. — СПб : БХВ-Петербург, 2007.— 651 с.
2	Тюкачев Н.А. Основы программирования в Delphi : [в 3 ч.] : [учебное пособие] / Н.А. Тюкачев, Е.Е. Михайлова. — Воронеж : Воронежский государственный университет, 2007-. Ч. 2: Алгоритмы на деревьях и графах. — 2007. — 188 с.
3	Тюкачев Н. А. Основы программирования в Delphi. Специальные вопросы : [в 3 ч.] / Н.А. Тюкачев. — Воронеж : Воронежский государственный университет, 2007-. Ч. 3. — 2007.

### б) дополнительная литература:

№ п/п	Источник
4	Андреев А. А. Программирование на языке Delphi : курс лекций / А.А. Андреев ; Тамбов. гос. ун-т им. Г.Р. Державина. — Тамбов : Изд-во ТГУ, 2009. — 254 с.
5	Архангельский А. Я. Программирование в Delphi / А. Я. Архангельский. — М. : Бином, 2005. — 1152 с.

### в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)\*:

№ п/п	Ресурс
1.	<a href="http://www.lib.vsu.ru">www.lib.vsu.ru</a> - ЗНБ ВГУ

## 16. Перечень учебно-методического обеспечения для самостоятельной работы

№ п/п	Источник
1	Тюкачев Н.А. Основы программирования в Delphi : [в 3 ч.] : [учебное пособие] / Н.А. Тюкачев, Е.Е. Михайлова. — Воронеж : Воронежский государственный университет, 2007-. Ч. 2: Алгоритмы на деревьях и графах. — 2007. — 188 с.
2	Андреев А. А. Программирование на языке Delphi : курс лекций / А.А. Андреев ; Тамбов. гос. ун-т им. Г.Р. Державина. — Тамбов : Изд-во ТГУ, 2009. — 254 с.

## 17. Информационные технологии, используемые для реализации учебной дисциплины, включая программное обеспечение и информационно-справочные системы (при необходимости)

Delphi 7 или выше

## 18. Материально-техническое обеспечение дисциплины:

Персональный компьютер с ОС Windows

---

## 19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1	Сортировки и поиск	ПК-1, ПК-4, ПК-5	ПК-1.1 ПК-1.2 ПК-1.3 ПК-4.1 ПК-4.2 ПК-4.3 ПК-5.1 ПК-5.2 ПК-5.3	Аттестация, лабораторная работа
2	Динамические структуры данных	ПК-1, ПК-4, ПК-5	ПК-1.1 ПК-1.2 ПК-1.3 ПК-4.1 ПК-4.2 ПК-4.3 ПК-5.1 ПК-5.2 ПК-5.3	Аттестация, лабораторная работа
3	Списки, стеки, очереди	ПК-1, ПК-4, ПК-5	ПК-1.1 ПК-1.2 ПК-1.3 ПК-4.1 ПК-4.2 ПК-4.3 ПК-5.1 ПК-5.2 ПК-5.3	Аттестация, лабораторная работа
4	Рекурсия	ПК-1, ПК-4, ПК-5	ПК-1.1 ПК-1.2 ПК-1.3 ПК-4.1 ПК-4.2 ПК-4.3 ПК-5.1 ПК-5.2 ПК-5.3	Аттестация, лабораторная работа
5	Нуль-терминированные строки. Процедурные типы	ПК-1, ПК-4, ПК-5	ПК-1.1 ПК-1.2 ПК-1.3 ПК-4.1 ПК-4.2 ПК-4.3 ПК-5.1 ПК-5.2 ПК-5.3	Аттестация, лабораторная работа
6	Алгоритмы на деревьях	ПК-1, ПК-4, ПК-5	ПК-1.1 ПК-1.2 ПК-1.3 ПК-4.1 ПК-4.2 ПК-4.3 ПК-5.1 ПК-5.2 ПК-5.3	Аттестация, лабораторная работа
7	Алгоритмы на графах	ПК-1, ПК-4, ПК-5	ПК-1.1 ПК-1.2 ПК-1.3 ПК-4.1 ПК-4.2 ПК-4.3 ПК-5.1 ПК-5.2 ПК-5.3	Аттестация, лабораторная работа
Промежуточная аттестация форма контроля - зачет				КИМ

## 20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

### 20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Лабораторные работы  
Беседа

Для оценивания результатов обучения на экзамене/зачете используются следующие показатели:

- 1) знание всех конструкций языка Delphi
- 2) умение декомпонировать задачу, правильно подбирать структуры для хранения данных
- 3) умение строить иерархию классов по предметной области
- 4) умение написания и отладки программ на Delphi

Для оценивания результатов обучения на зачете с оценкой используется 50-балльная шкала.

Соотношение показателей, критериев и шкалы оценивания результатов обучения.

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
Обучающийся в полной мере владеет теоретическими основами дисциплины, способен иллюстрировать ответ примерами, фактами из опыта написания программ и применять теоретические знания для решения практических задач	<i>Повышенный уровень</i>	<i>зачет</i>
Ответ на контрольно-измерительный материал не вполне соответствует двум из перечисленных показателей, но обучающийся дает правильные ответы на дополнительные вопросы. Недостаточно продемонстрировано владение системным подходом, или содержатся отдельные пробелы при описании терминологии дисциплины и ее практик.	<i>Базовый уровень</i>	<i>зачет</i>
Ответ на контрольно-измерительный материал не соответствует любым трем из перечисленных показателей, обучающийся дает неполные ответы на дополнительные вопросы. Демонстрирует лишь частичные знания, или не умеет связывать теорию с практикой, или имеет неполное представление о системном подходе, допускает существенные ошибки.	<i>Пороговый уровень</i>	<i>зачет</i>
Ответ на контрольно-измерительный материал не соответствует любым четырем из перечисленных показателей. Обучающийся демонстрирует отрывочные, фрагментарные знания, допускает грубые ошибки.	–	<i>Незачет</i>

## 20.2 Промежуточная аттестация

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в форме письменной работы и лабораторной работы. Критерии оценивания приведены выше.

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования.

Контрольно-измерительные материалы промежуточной аттестации включают в себя теоретические вопросы, позволяющие оценить уровень полученных знаний.

При оценивании используются качественные шкалы оценок.

### 20.2.1. Перечень вопросов к экзамену

1. Что такое технология программирования? Методы и средства разработки программных продуктов?
2. RUP-модель процесса разработки программного средства: принципы и структура.
3. Концептуальная модель системы: концептуальные классы, системные события и системные операции. Способ их представления в виде UML-диаграмм. Пример концептуального описания прецедента.
4. Понятие качества программных продуктов. Критерии качества.
5. Методология IDEF0, синтаксис IDEF0-моделей.
6. Диаграммы взаимодействия как элементы концептуальной модели. Синтаксис диаграмм взаимодействия. Примеры диаграмм взаимодействия.
7. Модель жизненного цикла программных средств. Фазы жизненного цикла. Этапы фазы разработки, их содержание.
8. Адаптивные процессы разработки программных средств. XP-модель процесса разработки. Принципы экстремального программирования.

9. Функциональные требования к системе. Способ их представления в виде UML-диаграммы. Пример диаграммы с использованием отношений «расширяет» и «включает».
10. Определение качества ПО в стандарте ISO 9126. Аспекты качества, их взаимное влияние.
11. Модели процесса разработки программных средств и реализуемые ими стратегии. Прогностические модели (процессы) разработки. Спиральная модель процесса разработки, ее характеристика.
12. Понятие прецедента и сценария. Пример прецедента, основного и дополнительного сценариев.
13. Модели процесса разработки программных средств и реализуемые ими стратегии. Прогностические модели (процессы) разработки. Инкрементная модель процесса разработки, ее характеристика.
14. Задачи этапа объектно-ориентированного анализа предметной области. Методика определения границ системы и ключевых абстракций. Пример проведения анализа.
15. Диаграммы потоков данных (DFD-диаграммы), их использование при моделировании предметной области.
16. Нефункциональные требования к системе, их виды. Примеры нефункциональных требований.
17. Модели процесса разработки программных средств и реализуемые ими стратегии. Прогностические модели (процессы) разработки. Каскадная модель, ее характеристика.
18. Многоуровневая модель качества ПО в стандарте ISO 9126.
19. Адаптивные процессы разработки программных средств. Scrum-модель процесса разработки.
20. Понятие модуля и модульного программирования. Преимущества модульного подхода к разработке ПО.
21. Архитектурные шаблоны (architectural patterns), их классификация.
22. Регрессионное тестирование. Критерии завершения отладочного тестирования.
23. Задачи, решаемые на стадии детального проектирования.
24. Особенности объектно-ориентированного проектирования.
25. Основные принципы тестирования.
26. Цели и задачи этапа проектирования. Понятие модели проектирования, ее отличия от концептуальной модели.
27. Обязанности программных классов, виды обязанностей. Визуализация распределения обязанностей посредством UML-диаграмм. Примеры диаграмм.
28. Восходящая и нисходящая стратегии интеграционного тестирования, механизмы их реализации.
29. Цели и задачи проектирования пользовательского интерфейса.
30. Структурные шаблоны, их назначение. Примеры структурных шаблонов с их описаниями.
31. Задачи, решаемые на стадии эскизного проектирования.
32. Структурный подход к формированию тестовых наборов. Пример реализации структурного подхода.
33. Стадии проектирования, их краткая характеристика.
34. Системное тестирование. Виды системного тестирования. Критерии полноты тестовых наборов.
35. Понятие архитектуры ПС. Проблема выбора архитектуры. Влияние архитектуры на качественные характеристики ПС.
36. Шаблоны распределения обязанностей, их назначение. Примеры применения.
37. Шаблоны анализа (analysis patterns), их классификация.
38. Шаблоны проектирования, их классификация. Правила описания шаблонов, примеры шаблонов с их описаниями.



39. Тесты и тестовые наборы. Понятие тестового покрытия. Отладочное тестирование. Соотношение структурного и функционального подходов.
40. Интеграционное тестирование. Виды интеграционного тестирования. Критерии полноты тестовых наборов.
41. Тестирование программного средства. Стадии тестирования и их характеристика.
42. Идентификация методов программных классов. Диаграммы классов, способы отображения отношений ассоциации и зависимости. Пример диаграммы классов.
43. Особенности объектно-ориентированного тестирования. Расширение области применения тестирования. Критерии тестирования моделей.
44. Цели, зависимость целей, описание целей.
45. Проектная команда, группы и роли в проектной команде.
46. Утилита модульного тестирования NUnit. Средства описания тестов.
47. Сценарий объединения правок. Конфликты и способы их разрешения.
48. Особенности методики модульного тестирования объектно-ориентированных систем. Тестирование классов.
49. Документирование программного продукта. Документация сопровождения, ее назначение и состав.
50. Понятие автоматизированного тестирования. Автотесты. Достоинства и недостатки автоматизированного тестирования.
51. Понятие сборки, манифест сборки. Сборка приложения, системы автоматизации сборки.
52. Критерии оценивания проектов, шкалы ценности проекта.
53. Директивы, категории директив.
54. Утилита NAnt, файл сборки и его структура.
55. Хранилище, его структура, правки. Команды SVN для работы с хранилищем.
56. Способы контроля хода выполнения проекта: меры и метрики. Виды метрик.
57. Базовое расписание проекта, точки контроля, распараллеливание работ.
58. Руководство проектом и особенности проектной деятельности.
59. Утверждения, параметры утверждений. Группы утверждений, классическая и закрытая модель утверждений.
60. Понятия рабочей копии и служебного каталога. Команды SVN для работы с рабочими копиями.
61. Документирование программного продукта. Пользовательская документация, ее назначение и состав.
62. Типы автоматизированного тестирования, их цели. Средства автоматизированного тестирования.
63. Система конкурирующих версий CVS, ее достоинства и недостатки.
64. Особенности методики интеграционного тестирования объектно-ориентированных систем. Тестирование кластеров и потоковое тестирование.
65. Документирование процесса разработки. Типы документов управления.
66. Генератор документации Sandcastle, его назначение и принцип работы.
67. Модели процесса разработки: каскадная, RAD, спиральная
68. Пример разработки программного средства
69. Руководство программным проектом. Размерно- и функционально-ориентированные метрики
70. Тестирование и отладка программного средства. Стадии тестирования и их характеристика. Основные принципы тестирования. Тесты и тестовые наборы. Понятие тестового покрытия.
71. Проектирование программных средств. Цели и задачи этапа проектирования. Понятие модели проектирования, ее отличия от концептуальной модели. Стадии проектирования, их краткая характеристика.
72. Понятие версии программного продукта и системы контроля версий. Модели версионирования, их сравнение.

73. Понятие модуля и модульного программирования. Преимущества модульного подхода к разработке ПО. Модули как средство физического структурирования ПО. Свойства модулей.
74. Особенности объектно-ориентированного тестирования. Расширение области применения тестирования. Критерии тестирования моделей. Тестирование классов. Тестирование кластеров и потоковое тестирование.
75. Структурный и объектно-ориентированный подходы к разработке ПО. Их сравнительный анализ. Сущность объектного подхода к разработке программных средств.
76. Анализ предметной области: цели и задачи. Модели предметной области. Формальные определения. Классификация моделей. Методология IDEF0, синтаксис IDEF0-моделей.
77. Система Subversion, ее архитектура. Хранилище, его структура, правки. Команды SVN для работы с хранилищем. Понятия рабочей копии и служебного каталога. Сценарий объединения правок. Конфликты и способы их разрешения.
78. Фаза разработки, этапы процесса разработки. Стратегии конструирования ПО: линейная, инкрементная, эволюционная
79. Объектно-ориентированный анализ предметной области. Методика определения границ системы и ключевых абстракций. Пример проведения анализа. Функциональные и нефункциональные требования к системе.
80. Стандарт ISO/IEC 12207-95: основные определения – система, модель жизненного цикла, квалификационные требования. Основные процессы, их содержание, работы и задачи процесса разработки.
81. Руководство программным проектом. Предварительные оценки проекта. Системный анализ и анализ требований. Анализ рисков. Планирование процесса разработки. Типовая структура распределения работ.
82. Критерии качества программного средства. Определение качества ПО в стандарте ISO 9126. Многоуровневая модель качества ПО. Оценочные характеристики качества программного продукта
83. Интеграционное тестирование. Виды интеграционного тестирования. Критерии полноты тестовых наборов. Регрессионное тестирование. Критерии завершения отладочного тестирования. Системное тестирование. Виды системного тестирования.
84. Понятие сборки, манифест сборки. Сборка приложения, системы автоматизации сборки. Утилита NAnt, файл сборки и его структура. Цели, зависимость целей, описание целей.
85. Понятие автоматизированного тестирования. Достоинства и недостатки автоматизированного тестирования. Средства автоматизированного тестирования. Утилита модульного тестирования NUnit. Средства описания тестов.
86. Адаптивные модели процесса разработки: экстремальное программирование, Scrum.
87. Отладочное тестирование. Соотношение структурного и функционального подходов. Примеры реализации.
88. Функциональные требования к системе. Способ их представления в виде UML-диаграммы. Пример диаграммы с использованием отношений «расширяет» и «включает». Понятие прецедента и сценария.
89. Стандарт ISO/IEC 15504 (SPICE): оценка возможностей разработчика. Связь этого стандарта с моделью зрелости предприятия SEI CMM.
90. Задачи, решаемые на стадии эскизного проектирования. Понятие архитектуры ПС. Проблема выбора архитектуры. Влияние архитектуры на качественные характеристики ПС.
91. Задачи, решаемые на стадии детального проектирования. Цели и задачи проектирования пользовательского интерфейса.