МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»
(ТУСУР)

УТВЕРЖДАЮ
Директор департамента образования
_____ П. Е. Троян
«___» _____ 20__ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Основы разработки программного обеспечения

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки / специальность: **09.03.01 Информатика и вычислительная техника**
Направленность (профиль) / специализация: **Программное обеспечение средств вычислительной техники и автоматизированных систем**
Форма обучения: **заочная (в том числе с применением дистанционных образовательных технологий)**
Факультет: **ФДО, Факультет дистанционного обучения**
Кафедра: **АСУ, Кафедра автоматизированных систем управления**
Курс: **3**
Семестр: **5**
Учебный план набора 2018 года

Распределение рабочего времени

№	Виды учебной деятельности	5 семестр	Всего	Единицы
1	Самостоятельная работа под руководством преподавателя	16	16	часов
2	Лабораторные работы	4	4	часов
3	Контроль самостоятельной работы	2	2	часов
4	Всего контактной работы	22	22	часов
5	Самостоятельная работа	149	149	часов
6	Всего (без экзамена)	171	171	часов
7	Подготовка и сдача экзамена	9	9	часов
8	Общая трудоемкость	180	180	часов
			5.0	З.Е.

Контрольные работы: 5 семестр - 1
Экзамен: 5 семестр

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Шелупанов А.А.
Должность: Ректор
Дата подписания: 28.02.2018
Уникальный программный ключ:
c53e145e-8b20-45aa-9347-a5e4dbb90e8d

Томск 2018

Согласована на портале № 45764

Рассмотрена и одобрена на заседании кафедры
протокол № 6 от «17» 5 2018 г.

ЛИСТ СОГЛАСОВАНИЯ

Рабочая программа дисциплины составлена с учетом требований федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) 09.03.01 Информатика и вычислительная техника, утвержденного 12.01.2016 года, рассмотрена и одобрена на заседании кафедры АСУ «___» _____ 20__ года, протокол №_____.

Разработчик:

доцент каф. АСУ	_____	А. К. Лукьянов
Заведующий обеспечивающей каф. АСУ	_____	А. М. Кориков

Рабочая программа дисциплины согласована с факультетом и выпускающей кафедрой:

Декан ФДО	_____	И. П. Черкашина
Заведующий выпускающей каф. АСУ	_____	А. М. Кориков

Эксперты:

Доцент кафедры технологий электронного обучения (ТЭО)	_____	Ю. В. Морозова
Доцент кафедры автоматизированных систем управления (АСУ)	_____	А. И. Исакова

Согласована на портале № 45764

1. Цели и задачи дисциплины

1.1. Цели дисциплины

Рассмотрение вопросов проектирования, разработки и испытаний больших программных систем с точки зрения требований разработчика.

Изучение современных технологий разработки программного обеспечения.

1.2. Задачи дисциплины

- Изучение методов разработки программного обеспечения.
- Изучение способов создания функциональных спецификаций.
- Изучение методов проектирования программных комплексов.
- Изучение методов создания абстрактных типов данных.
- Изучение методов доказательства правильности программ.
- Изучение методов организации тестов.
- Изучение методов сопровождения программных комплексов.

2. Место дисциплины в структуре ОПОП

Дисциплина «Основы разработки программного обеспечения» (Б1.В.ОД.8) относится к блоку 1 (вариативная часть).

Предшествующими дисциплинами, формирующими начальные знания, являются: Информатика, Программирование.

Последующими дисциплинами являются: Преддипломная практика, Проектирование систем управления.

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

- ПК-3 способностью обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности;

В результате изучения дисциплины обучающийся должен:

- **знать** принципы разработки и методы проектирования программных систем, методы управления проектированием программных систем и организации коллективов разработчиков, государственные стандарты, регламентирующие процесс разработки программных систем и их описания; теоретические основы методов разработки программного обеспечения, способы создания функциональных спецификаций, методы проектирования программных комплексов, создания абстрактных типов данных, доказательства правильности программ, организации тестов и сопровождения программных комплексов.
- **уметь** самостоятельно выполнять цикл проектирования программного обеспечения, разрабатывать спецификации и абстрактные типы данных на основе анализа требований, предъявляемых к программному обеспечению, доказывать правильность программ, проектировать и кодировать необходимые тесты, пользоваться стандартными терминами и определениями, читать научные статьи и пользоваться литературой для самостоятельного решения научно-исследовательских задач, связанных с разработкой программных систем.
- **владеть** навыками проектирования и разработки больших программных комплексов с использованием перспективных направлений и методических подходов в данной области.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 5.0 зачетных единицы и представлена в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры
		5 семестр
Контактная работа (всего)	22	22
Самостоятельная работа под руководством преподавателя (СРП)	16	16

Согласована на портале № 45764

Лабораторные работы	4	4
Контроль самостоятельной работы (КСР)	2	2
Самостоятельная работа (всего)	149	149
Подготовка к контрольным работам	32	32
Оформление отчетов по лабораторным работам	21	21
Подготовка к лабораторным работам	16	16
Самостоятельное изучение тем (вопросов) теоретической части курса	80	80
Всего (без экзамена)	171	171
Подготовка и сдача экзамена	9	9
Общая трудоемкость, ч	180	180
Зачетные Единицы	5.0	

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Разделы дисциплины и виды занятий приведены в таблице 5.1.

Таблица 5.1 – Разделы дисциплины и виды занятий

Названия разделов дисциплины	СРП, ч	Лаб. раб., ч	КСР, ч	Сам. раб., ч	Всего часов (без экзамена)	Формируемые компетенции
5 семестр						
1 Введение. Проблемы современного программирования	2	0	2	14	16	ПК-3
2 Этапы разработки программного обеспечения	2	0		14	16	ПК-3
3 Методы разработки программного обеспечения как научная дисциплина	2	0		14	16	ПК-3
4 Методы разработки программного обеспечения	2	0		14	16	ПК-3
5 Правильность программ	2	0		14	16	ПК-3
6 Тестирование	2	4		51	57	ПК-3
7 Технология разработки программ	2	0		14	16	ПК-3
8 Методы управления проектированием программных изделий	2	0		14	16	ПК-3
Итого за семестр	16	4	2	149	171	
Итого	16	4	2	149	171	

Согласована на портале № 45764

5.2. Содержание разделов дисциплины (самостоятельная работа под руководством преподавателя)

Содержание разделов дисциплин (самостоятельная работа под руководством преподавателя) приведено в таблице 5.2.

Таблица 5.2 – Содержание разделов дисциплин (самостоятельная работа под руководством преподавателя)

Названия разделов	Содержание разделов дисциплины (самостоятельная работа под руководством преподавателя)	Трудоемкость, ч	Формируемые компетенции
5 семестр			
1 Введение. Проблемы современного программирования	Краткая характеристика дисциплины, её цели и задачи, порядок изучения материала, связи с другими дисциплинами учебного плана. Перечень дисциплин, усвоение которых необходимо студентам для изучения данной дисциплины. Основы методики и форм контроля самостоятельной работы, краткая характеристика учебной литературы.	2	ПК-3
	Итого	2	
2 Этапы разработки программного обеспечения	Анализ требований, предъявляемых к системе. Определение спецификаций. Проектирование. Кодирование. Тестирование. Эксплуатация и сопровождение.	2	ПК-3
	Итого	2	
3 Методы разработки программного обеспечения как научная дисциплина	Методы управления разработкой. Методы проведения разработки программного обеспечения. Развитие методов разработки программного обеспечения.	2	ПК-3
	Итого	2	
4 Методы разработки программного обеспечения	Язык проектирования программ. Стратегия проектирования. Данные.	2	ПК-3
	Итого	2	
5 Правильность программ	Аксиомы. Правила преобразования данных. Доказательства правильности программ.	2	ПК-3
	Итого	2	
6 Тестирование	Психология и экономика тестирования программ. Экономика тестирования. Ручное тестирование. Проектирование теста.	2	ПК-3
	Итого	2	
7 Технология разработки программ	Разбиение задачи на независимые подзадачи. Разбиение задачи на одинаковые по сложности части. Рекурсия и динамическое программирование. Поиск. Сортировка. Алгоритм выбора из конечного состояния. Сопрограммы.	2	ПК-3
	Итого	2	

Согласована на портале № 45764

8 Методы управления проектированием программных изделий	Организация управления проектированием программного изделия. Организация планирования разработок программного изделия. Организация разработки программного изделия. Организация обслуживания разработки программного изделия. Организация выпуска документации. Организация испытаний программных изделий.	2	ПК-3
	Итого	2	
Итого за семестр		16	

5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами

Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами представлены в таблице 5.3.

Таблица 5.3 – Разделы дисциплины и междисциплинарные связи

Наименование дисциплин	№ разделов данной дисциплины, для которых необходимо изучение обеспечивающих и обеспечиваемых дисциплин							
	1	2	3	4	5	6	7	8
Предшествующие дисциплины								
1 Информатика	+							
2 Программирование				+	+			
Последующие дисциплины								
1 Преддипломная практика	+	+	+					
2 Проектирование систем управления				+				

5.4. Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий представлено в таблице 5.4.

Таблица 5.4 – Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Компетенции	Виды занятий				Формы контроля
	СРП	Лаб. раб.	КСР	Сам. раб.	
ПК-3	+	+	+	+	Контрольная работа, Экзамен, Проверка контрольных работ, Отчет по лабораторной работе, Тест

6. Интерактивные методы и формы организации обучения

Не предусмотрено РУП.

7. Лабораторные работы

Наименование лабораторных работ приведено в таблице 7.1.

Таблица 7.1 – Наименование лабораторных работ

Названия разделов	Наименование лабораторных работ	Трудоемкость, ч	Формируемые компетенции
5 семестр			
6 Тестирование	Написание спецификаций и проведение	4	ПК-3

Согласована на портале № 45764

	тестирования ПО. Составление руководства системного программиста.		
	Итого	4	
Итого за семестр		4	

8. Контроль самостоятельной работы

Виды контроля самостоятельной работы приведены в таблице 8.1.

Таблица 8.1 – Виды контроля самостоятельной работы

№	Вид контроля самостоятельной работы	Трудоемкость (час.)	Формируемые компетенции
5 семестр			
1	Контрольная работа с автоматизированной проверкой	2	ПК-3
Итого		2	

9. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 9.1.

Таблица 9.1 – Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
5 семестр				
1 Введение. Проблемы современного программирования	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	4		
	Итого	14		
2 Этапы разработки программного обеспечения	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	4		
	Итого	14		
3 Методы разработки программного обеспечения как научная дисциплина	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	4		
	Итого	14		
4 Методы разработки программного обеспечения	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	4		
	Итого	14		
5 Правильность	Самостоятельное изуче-	10	ПК-3	Контрольная рабо-

Согласована на портале № 45764

программ	ние тем (вопросов) теоретической части курса			та, Тест, Экзамен
	Подготовка к контрольным работам	4		
	Итого	14		
6 Тестирование	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ПК-3	Контрольная работа, Отчет по лабораторной работе, Тест, Экзамен
	Подготовка к лабораторным работам	16		
	Оформление отчетов по лабораторным работам	21		
	Подготовка к контрольным работам	4		
	Итого	51		
7 Технология разработки программ	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	4		
	Итого	14		
8 Методы управления проектированием программных изделий	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	4		
	Итого	14		
	Выполнение контрольной работы	2	ПК-3	Контрольная работа
Итого за семестр		149		
	Подготовка и сдача экзамена	9		Экзамен
Итого		158		

10. Контроль самостоятельной работы (курсовой проект / курсовая работа)

Не предусмотрено РУП.

11. Рейтинговая система для оценки успеваемости обучающихся

Рейтинговая система не используется.

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основная литература

1. Калайда, В. Т. Технология разработки программного обеспечения [Электронный ресурс]: учебное пособие / В. Т. Калайда, В. В. Романенко. – Томск: Томский межвузовский центр дистанционного образования, 2007. Доступ из личного кабинета студента. — Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 10.09.2018).

12.2. Дополнительная литература

1. Калентьев, А.А. Новые технологии в программировании [Электронный ресурс]: учеб-

Согласована на портале № 45764

ное пособие / А.А. Калентьев, Д.В. Гарайс, А. Е. Горяинов — Томск : Эль Контент, 2014. Доступ из личного кабинета студента. — Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 10.09.2018).

12.3. Учебно-методические пособия

12.3.1. Обязательные учебно-методические пособия

1. Елизаров, А. И. Технология разработки программного обеспечения [Электронный ресурс]: учебное методическое пособие / А. И. Елизаров, В. В. Романенко – Томск: Томский межвузовский центр дистанционного образования, 2007. Доступ из личного кабинета студента. — Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 10.09.2018).
2. Калайда, В. Т. Основы разработки программного обеспечения: электронный курс / В. Т. Калайда. – Томск: ТУСУР, ФДО, 2018. Доступ из личного кабинета студента.
3. Калайда, В. Т. Основы разработки программного обеспечения [Электронный ресурс]: методические указания по организации самостоятельной работы для студентов заочной формы обучения направления подготовки 09.03.01 Информатика и вычислительная техника, обучающихся с применением дистанционных образовательных технологий / В. Т. Калайда. – Томск ТУСУР, ФДО, 2018. Доступ из личного кабинета студента. — Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 10.09.2018).

12.3.2. Учебно-методические пособия для лиц с ограниченными возможностями здоровья и инвалидов

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа лиц с ограниченными возможностями здоровья и инвалидов предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

- в форме электронного документа;
- в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

- в форме электронного документа;
- в печатной форме.

12.4. Профессиональные базы данных и информационные справочные системы

1. www.compress.ru – Журнал «КомпьютерПресс»
2. www.osp.ru – Издательство «Открытые системы»
3. www.cnews.ru – Издание о высоких технологиях
4. www.it-daily.ru – Новости российского ИТ-рынка
5. www.isn.ru – Российская сеть информационного общества

13. Материально-техническое обеспечение дисциплины и требуемое программное обеспечение

13.1. Общие требования к материально-техническому и программному обеспечению дисциплины

13.1.1. Материально-техническое и программное обеспечение дисциплины

Кабинет для самостоятельной работы студентов

учебная аудитория для проведения занятий лабораторного типа, помещение для проведения групповых и индивидуальных консультаций, помещение для проведения текущего контроля и промежуточной аттестации, помещение для самостоятельной работы

634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Описание имеющегося оборудования:

- Коммутатор MicroTeak;
- Компьютер PENTIUM D 945 (3 шт.);
- Компьютер GELERON D 331 (2 шт.);

Согласована на портале № 45764

- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

– 7-zip (с возможностью удаленного доступа)
– Google Chrome
– Kaspersky Endpoint Security 10 для Windows (с возможностью удаленного доступа)
– LibreOffice (с возможностью удаленного доступа)
– MS Office версий 2010 (с возможностью удаленного доступа)
– Microsoft Windows
– OpenOffice (с возможностью удаленного доступа)

13.1.2. Материально-техническое и программное обеспечение для лабораторных работ

Кабинет для самостоятельной работы студентов

учебная аудитория для проведения занятий лабораторного типа, помещение для проведения групповых и индивидуальных консультаций, помещение для проведения текущего контроля и промежуточной аттестации, помещение для самостоятельной работы

634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Описание имеющегося оборудования:

- Коммутатор MicroTeak;
- Компьютер PENTIUM D 945 (3 шт.);
- Компьютер GELERON D 331 (2 шт.);
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

– 7-zip (с возможностью удаленного доступа)
– Google Chrome
– Kaspersky Endpoint Security 10 для Windows (с возможностью удаленного доступа)
– LibreOffice (с возможностью удаленного доступа)
– MS Office версий 2010 (с возможностью удаленного доступа)
– Microsoft Windows
– OpenOffice (с возможностью удаленного доступа)

13.1.3. Материально-техническое и программное обеспечение для самостоятельной работы

Для самостоятельной работы используются учебные аудитории (компьютерные классы), расположенные по адресам:

- 634050, Томская область, г. Томск, Ленина проспект, д. 40, 233 ауд.;
- 634045, Томская область, г. Томск, ул. Красноармейская, д. 146, 201 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 47, 126 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Состав оборудования:

- учебная мебель;
- компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 5 шт.;
- компьютеры подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду университета.

Перечень программного обеспечения:

- Microsoft Windows;
- OpenOffice;
- Kaspersky Endpoint Security 10 для Windows;
- 7-Zip;
- Google Chrome.

Согласована на портале № 45764

13.2. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья и инвалидов

Освоение дисциплины лицами с ограниченными возможностями здоровья и инвалидами осуществляется с использованием средств обучения общего и специального назначения.

При занятиях с обучающимися **с нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах, мобильной системы преподавания для обучающихся с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой занимаются обучающиеся с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При занятиях с обучающимися **с нарушениями зрениями** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для комфортного просмотра.

При занятиях с обучающимися **с нарушениями опорно-двигательного аппарата** используются альтернативные устройства ввода информации и другие технические средства приема/передачи учебной информации в доступных формах, мобильной системы обучения для людей с инвалидностью.

14. Оценочные материалы и методические рекомендации по организации изучения дисциплины

14.1. Содержание оценочных материалов и методические рекомендации

Для оценки степени сформированности и уровня освоения закрепленных за дисциплиной компетенций используются оценочные материалы в составе:

14.1.1. Тестовые задания

1. Отметьте базовые требования к программной системе, которые могут быть выделены на этапе анализа разработки ПО.

– время работы программы
– описание функций
– структура входных и выходных данных
– стоимость обработки
– вероятность ошибки
– разработка алгоритмов
– организация базы данных
– формирование структуры программной системы
– реакция на непредсказуемые действия оператора (защита от дурака)
– данные для тестирования элементов системы

2. Основные вопросы, рассматриваемые на этапе определения спецификаций, это:

– время работы программы
– описание функций
– структура входных и выходных данных
– стоимость обработки
– вероятность ошибки
– разработка алгоритмов
– организация базы данных
– формирование структуры программной системы
– реакция на непредсказуемые действия оператора (защита от дурака)
– данные для тестирования элементов системы

3. Какие действия (из перечисленных) выполняются на этапе проектирования:

– время работы программы
– описание функций
– структура входных и выходных данных
– стоимость обработки
– вероятность ошибки

Согласована на портале № 45764

– разработка алгоритмов
– организация базы данных
– формирование структуры программной системы
– реакция на непредсказуемые действия оператора (защита от дурака)
– данные для тестирования элементов системы

4. Выделите основные категории ПО:
– библиотеки
– управляющие программы
– базы данных
– прикладные программы
– файлы справки
– Internet-страницы
– системные программы

5. Методы управления разработкой…
– охватывают технические приемы работы программистов, способствующие повышению производительности их труда
– применяются для эффективной организации работы исполнителей
– должны учитывать требования различных заказчиков и включать средства индикации, тестирования и устранения ошибок, применяемые для корректировки систем

6. Зачем создается бригада главного программиста?
– для освобождения программистов для более квалифицированных работ
– для осуществления функции интерфейса между программистом и ЭВМ
– для уменьшения количества взаимосвязей между программистами

7. Отметьте верные утверждения о нисходящем проектировании (НП):
– При НП вначале проектируется управляющая программа (драйвер), затем происходит иерархическое структурирование и разбивка путем последовательного уточнения
– Язык PDL хорошо подходит для НП
– При НП вначале проектируются программы нижнего уровня
– НП также называют пошаговым совершенствованием
– Большинство систем проектируется методами НП
– Кодирование и тестирование удобнее осуществлять по принципам НП
– Большинство систем реального времени проектируется методами НП

8. Отметьте верные утверждения о восходящем проектировании (ВП):
– Основное назначение языка PDL – это реализация ВП
– При ВП вначале проектируется управляющая программа (драйвер), затем происходит иерархическое структурирование и разбивка путем последовательного уточнения
– Большинство систем реального времени проектируется методами ВП
– При ВП вначале проектируются программы нижнего уровня
– ВП также называют пошаговым совершенствованием
– Большинство систем проектируется методами ВП
– Кодирование и тестирование удобнее осуществлять по принципам ВП

9. Какие из приведенных высказываний верны?
– Программу, для каждого из множества узлов которой существует путь от точки входа через этот узел к точке выхода, можно охарактеризовать как не простую
– Простая программа является элементарной
– У простой программы существуют только одна точка входа и одна точка выхода
– В простой программе для каждого узла существует путь от точки входа через этот узел к точке выхода
– Элементарная программа не является простой
– Элементарная программа является простой

10. Отметьте истинные правила следствия:

Согласована на портале № 45764

– если {A}S{B}и B ^ C, то {B}S{C}
– если {A}S{B}и B ^ C, то {A}S{C}
– если {A}S{B}и B ^ C, то {C}S{A}
– если {A}S{B}и C ^ A, то {C}S{A}
– если {A}S{B}и C ^ A, то {C}S{B}
– если {A}S{B}и C ^ A, то {A}S{C}

11. Дана следующая программа, возводящая целое число A в целую положительную степень B (A^B = AB):

1. POWER (R,A,B);
2. declare X;
3. declare R;
4. declare A,B;
5. R=1;
6. X=0;
7. do while (X<B);
8. R=R*A;
9. X=X1;
10. end;
11. end POWER;

Входным утверждением является предикат B > 0. Каким должен быть инвариант цикла в 7-й строке?

– R = A^(B – X)
– R = A^(X 1 – B)
– R = A^B
– R = R*(X 1)
– R = A^(B – X 1)
– R = A^(X – B)
– R = A^X
– R = R*A^(X – 1)

12. Какие из приведенных определений наиболее полно характеризуют тестирование? Отметьте их.

– Тестирование представляет собой процесс, демонстрирующий отсутствие ошибок в программе.

– Цель тестирования – показать, что программа корректно исполняет предусмотренные функции.

– Тестирование – это процесс исполнения программы с целью обнаружения ошибок.

– Тестирование – это процесс, позволяющий убедиться в том, что программа выполняет свое назначение.

13. Отметьте верные утверждения о тестировании программ:

– Тестовый прогон, приведший к обнаружению ошибки, нельзя назвать удачным.
– Тестирование – процесс деструктивный.
– Удачным является тестовый прогон, приведший к обнаружению ошибки.
– Очевидно, что тестирование – процесс конструктивный (т. е. обратный деструктивному).
– Ошибки в программе могут быть и тогда, когда она делает, и что от нее не требуется.
– Набор тестов, который привел к обнаружению ошибки, считается неудачным.
– Невозможно создать тест, обнаруживающий все ошибки программы.

14. Отметьте верные утверждения об исчерпывающем входном тестировании:

– Построение любого исчерпывающего теста невозможно.
– Построение исчерпывающего теста – сложная, но реализуемая задача.
– Возможно построение только входного исчерпывающего теста.

– Критерием исчерпывающего входного тестирования является проверка каждой ветви алгоритма программы хотя бы один раз.

– Критерием исчерпывающего входного тестирования является обнаружение всех ошибок в

Согласована на портале № 45764

программе.

– Исчерпывающее входное тестирование не может обнаружить ошибок, появление которых зависит от обрабатываемых данных.

– Исчерпывающее входное тестирование является предпочтительной стратегией тестирования.

15. Отметьте верные утверждения об исчерпывающем тестировании маршрутов:

– Построение исчерпывающего теста – сложная, но реализуемая задача.

– Возможно построение только исчерпывающего теста маршрутов.

– Построение любого исчерпывающего теста невозможно.

– При исчерпывающем тестировании маршрутов подразумевается, что программа проверена полностью, если с помощью тестов удается осуществить выполнение этой программы по всем возможным ветвям алгоритма.

– Критерием исчерпывающего тестирования маршрутов является обнаружение всех ошибок в программе.

– Исчерпывающее тестирование маршрутов позволяет обнаружить ошибки, появление которых зависит от обрабатываемых данных.

– Исчерпывающее тестирование маршрутов является предпочтительной стратегией тестирования.

16. Какие технологии из перечисленных используются при разработке программ?

– Сопрограммы

– Доказательства правильности программ

– Разбиение задачи на одинаковые по сложности части

– Управления проектированием программного изделия

– Тестирование

– Разбиение задачи на независимые подзадачи

– Динамическое программирование

– Рекурсия

– Эксплуатация и сопровождение

– Алгоритм выбора из конечного состояния

17. Отметьте верные утверждения о технологиях разработки программ:

– Моделирование применяется, когда точное решение получить невозможно

– Рекурсивное решение выглядит более сложным, чем нерекурсивный вариант

– Динамическое программирование является табличным методом

– При разбиении задачи подзадачи получаются одинаковыми по сложности

– Скорость рекурсивного решения выше, чем скорость нерекурсивного

– Ввиду очевидных ограничений, моделирование практически не применимо при разработке сложных программных комплексов

– При использовании динамического программирования однажды найденный результат в будущем пересчитывается динамически

– Наиболее часто при разработке программ используется разбиение задач

– Скорость рекурсивного решения ниже, чем скорость нерекурсивного

– Моделирование применяется, когда необходимо получить точное решение

18. Какие существуют виды поиска в линейных списках?

– двоичный поиск

– поиск с возвратом

– прямой поиск

– шестнадцатеричный поиск

– выбор из конечного состояния

– линейный поиск

– десятичный поиск

– наилучшего размещения

– поиск слиянием

– хеширование

Согласована на портале № 45764

19. Какие существуют виды поиска в деревьях?

– двоичный поиск

– шестнадцатеричный поиск

– поиск в ширину

– поиск в высоту

– обход кроны дерева

– поиск с возвратом

– наилучшего размещения

– поиск в глубину

– древовидный поиск

20. В чем заключается основная цель управления проектированием программных изделий?

– Организовать планирование разработок программного изделия

– Организовать и связать взаимодействие исполнителей при создании программного продукта

– Организовать обслуживание разработки программного изделия

14.1.2. Экзаменационные тесты

1. Для чего предназначены периоды обновления ПО?

– для исключения лавинообразного нарастания версий системы

– для учета требований различных заказчиков, используя средства индикации, тестирования и устранения ошибок для корректировки системы

– для того, чтобы, несмотря на сбои, система продолжала удовлетворительно функционировать

2. Отметьте верные определения правильности и надежности систем:

– Система называется надежной, если, несмотря на сбои, она продолжает удовлетворительно функционировать

– Правильная система удовлетворяет своим спецификациям

– Надежная система не обязательно является правильной, но выдает приемлемый результат даже в том случае, когда входные данные либо условия ее использования не удовлетворяют принятым допущениям

– Система является правильной, если в ней нет ошибок, а ее внутренние данные не содержат выбросов

3. Отметьте верные определения правильности и надежности программ:

– Программа является правильной, если в ней нет ошибок, а ее внутренние данные не содержат выбросов

– Надежная программа не обязательно является правильной, но выдает приемлемый результат даже в том случае, когда входные данные либо условия ее использования не удовлетворяют принятым допущениям

– Правильная программа удовлетворяет своим спецификациям

– Программа называется надежной, если, несмотря на сбои, она продолжает удовлетворительно функционировать

4. Какие компоненты используются при строительстве иерархически структурированных программ, согласно методике Джексона?

– Структура данных

– Итерация

– Элемент

– Массив

– Выборка

– Компилятор

– Последовательность

– Естественный язык

5. Укажите шаги, выполняемые при использовании системы SREM:

– Синтез

– Трансляция

– Кодирование

Согласована на портале № 45764

– Выборка
– Декомпозиция
– Распределение
– Верификация
– Анализ
– Тестирование
– Итерация

6. Какие роли разработчиков рассматриваются при использовании системы SADT?
– Читатели
– Программист
– Главный аналитик
– Инструктор
– Главный программист (бригадир)
– Технический комитет
– Библиотекарь
– Руководитель проекта
– Отладчик
– Авторы
– Заказчик

7. Какие утверждения о защите данных от несанкционированного доступа являются верными?
– Защита данных основана на фиксированных типах данных
– Защита данных основана на размещении указателей
– Имеющиеся в языке PDL средства позволяют защитить данные лишь отчасти
– Имеющиеся в языке PDL средства позволяют полностью защитить данные

8. Укажите, элементам каких агрегативных типов данных (из перечисленных ниже) может соответствовать следующее описание на языке PDL:

```
declare 1 ELEM BASED,
2 DATA TYPE(SOME_TYPE),
2 LIST BASED,
3 DATA POINTER,
3 PTR POINTER,
2 LIST_HEAD POINTER;
```

– массиву
– структуре
– списку
– очереди
– стеку
– множеству
– графу
– дереву

9. Выделите вопросы, которые не предназначены для выявления ошибок передачи управления:
– Для всех обращений с помощью указателей или переменных-ссы¬лок память, к которой производится обращение, уже распределена?
– Возможно ли, что из-за входных условий цикл никогда не сможет выполняться?
– Если в программе содержится переключатель, то может ли значение индекса когда-ли¬бо превысить число возможных переходов?
– Верны ли предположения о порядке оценки и следования операторов для выражений, содержащих более чем один оператор?
– Будет ли программа, модуль или подпрограмма, в конечном счете, завершена?
– Будет ли каждый цикл, в конце концов, завершен?
– Если к структуре данных обращаются из нескольких процедур или подпрограмм, то опре-

Согласована на портале № 45764

делена ли эта структура одинаково в каждой процедуре?

– Существуют ли какие-ни¬будь ошибки «отклонения от нормы»?

– Существуют ли решения, подразумеваемые по умолчанию?

– Для циклов, управляемых как числом итераций, так и булевским условием, какова последовательность «погружения в тело цикла»?

– Отличаются ли типы или атрибуты переменных величин от тех, которые предполагались компилятором?

10. Выделите вопросы, которые не предназначены для выявления ошибок интерфейса:

– Равно ли число параметров, получаемых рассматриваемым модулем, числу аргументов, передаваемых каждым из вызывающих модулей? Правилен ли порядок их следования?

– Если не все атрибуты переменной явно присутствуют в описании, то понятно ли их отсутствие?

– Соответствуют ли атрибуты каждого аргумента, передаваемого другому модулю, атрибутам соответствующего параметра в рассматриваемом модуле?

– Равно ли число аргументов, передаваемых из рассматриваемого модуля другому модулю, числу параметров, ожидаемых в вызываемом модуле?

– Совпадают ли атрибуты каждого параметра с атрибутами соответствующего ему аргумента?

– Если одна и та же область памяти имеет несколько псевдонимов (имен) с различными атрибутами, то имеют ли значения данных в этой области корректные атрибуты при обращении по одному из этих псевдонимов?

– Если имеются глобальные переменные, имеют ли они одинаковые определения и атрибуты во всех модулях, которые к ним обращаются?

– Если модуль имеет несколько точек входа, передается ли параметр всегда вне зависимости от точки входа?

– Совпадают ли единицы измерения каждого параметра с единицами измерения соответствующих аргументов?

– Если вызываются встроенные функции, правильно ли заданы число, атрибуты и порядок следования аргументов?

– Если в программе содержится переключатель, то может ли значение индекса когда-ли¬бо превысить число возможных переходов?

11. Выделите вопросы, которые не предназначены для выявления ошибок ввода-вывода:

– Правильно ли для каждой переменной определены длина, тип и класс памяти?

– Правильно ли трактуются ошибочные состояния ввода-вы¬во¬да?

– Все ли файлы открыты перед их использованием?

– Верны ли предположения о порядке оценки и следования операторов для выражений, содержащих более чем один оператор?

– Являются ли правильными атрибуты файлов, описанных явно?

– Существуют ли смысловые или грамматические ошибки в тексте, выводимом программой на печать или экран дисплея?

– Равен ли размеру записи размер области памяти для ввода-вы¬во¬да?

– Согласуется ли спецификация формата с информацией в операторах ввода-вы¬во¬да?

– Правильно ли обнаруживаются и трактуются признаки конца файла?

12. На какие вопросы отвечают участники оценки программы посредством просмотра?

– Легко ли было понять программу?

– Влияет ли на результат выполнения программы способ, которым конкретный компилятор выполняет булевские выражения?

– Будет ли программа, модуль или подпрограмма, в конечном счете, завершена?

– Оказались ли результаты проектирования высокого уровня очевидными и приемлемыми?

– Существуют ли какие-ни¬будь ошибки «отклонения от нормы»?

– Оказались ли результаты проектирования низкого уровня очевидными и приемлемыми?

– Легко ли для вас модифицировать эту программу?

– Испытывали бы вы чувство удовлетворения, написав такую программу?

– Существуют ли смысловые или грамматические ошибки в тексте, выводимом программой

Согласована на портале № 45764

на печать или экран дисплея?

– Не пропущена ли в программе какая-ни¬будь функция?

13. Какие существуют виды поиска в деревьях?

– двоичный поиск
– шестнадцатеричный поиск
– поиск в ширину
– поиск в высоту
– обход кроны дерева
– поиск с возвратом
– наилучшего размещения
– поиск в глубину
– древовидный поиск

14. Какие существуют стратегии распределения памяти?

– Сопрягаемые области памяти
– Хеширование
– Наилучшее размещение
– Слияние
– Первое возможное размещение
– Выбор из конечного состояния
– Дамп
– Использование стека

15. На основе ответов на какие вопросы руководитель группы испытаний принимает решение о выпуске изделия?

– Можно ли отнести большинство неисправленных ошибок к разряду малозначительных?

– Является ли характер обнаруженных серьезных ошибок таким, что их можно устранить, выпустив программные «заплатки» в будущем?

– Не помогут ли дальнейшие испытания выявить серьезные ошибки?

– Уменьшались ли число и степень серьезности ошибок по мере перехода от одного цикла испытаний к другому?

– Имеют ли группы разработки и выпуска документации обоснованные планы устранения всех обнаруженных ошибок в фазе использования?

– Низка ли вероятность возникновения ошибки в результате действий пользователя, не предусмотренных спецификацией?

– Является ли характер обнаруженных серьезных ошибок таким, что их влияние можно устранить, сделав соответствующее пояснение в информационном листке выпуска?

– Всеми ли имеющимися тестами испытано изделие?

16. Какие процедуры обеспечивают качество программного изделия?

– простота сопровождения
– повторяемость результатов
– мозговой штурм
– использование стандартных методов проектирования и программирования
– использование творческого подхода
– анализ эксплутационных характеристик
– испытания
– восстанавливаемость после отказа

17. Выделите задачи группы испытаний:

– документирование результатов
– участие в фазовых обзорах с целью влияния на ход разработок
– руководство процессом кодирования
– проведение испытаний
– выработка оценок

18. Какие действия включает в себя производство программного изделия?

Согласована на портале № 45764

– контроль качества изделия путем испытания класса A
– упаковку и отправку потребителям
– контроль качества изделия путем испытания класса C
– тиражирование программ
– контроль качества изделия путем испытания класса B
– разработка программы
– тиражирование программ

19. Какие виды заявок поступают от пользователей программного изделия?
– заявки на возврат денег
– заявки на исправление ошибок
– заявки на расширение функций
– заявки на выпуск новых версий
– заявки на проведение проверки

20. Отметьте функции группы обслуживания:
– производственные операции
– распространение
– уборка помещений
– синхронизация работ различных групп
– системное обеспечение
– конфигурационное управление
– брошюрирование листингов

14.1.3. Темы контрольных работ

Основы разработки программного обеспечения

1. Какие вопросы возникли по мере развития средств вычислительной техники и накопления технических навыков написания программ?

– Не возникла ли необходимость в создании новых методов разработки программного обеспечения?

– Удовлетворяют ли имеющиеся языки программирования тенденциям развития отрасли?

– Не возникла ли необходимость в создании новых средств разработки и отладки программ?

– Продолжаем ли мы делать ошибки?

– Является ли процесс написания программ правильным?

2. Отметьте верное утверждение о прогнозе сроков разработки систем:

– При разработке технических систем возможен достаточно точный прогноз, тогда как при разработке программных систем он оказывается несостоятельным.

– При разработке как технических, так и программных систем достаточно точный прогноз невозможен.

– При разработке как технических, так и программных систем возможен достаточно точный прогноз.

– Ни при разработке технически систем, ни при разработке программных систем достаточно точный прогноз невозможен.

3. Какие этапы занимают 10% затрат от общего времени разработки программной системы?
– Анализ требований.
– Определение спецификаций.
– Проектирование.
– Кодирование.
– Тестирование.
– Автономное тестирование.
– Комплексное тестирование.
– Системное тестирование.
– Сопровождение.

4. Каков центральный вопрос определения спецификаций?
– Точное описание функций, реализуемых ЭВМ.
– Задание структуры входных и выходных данных.
– Определение алгоритмов обработки данных.

Согласована на портале № 45764

– Проблема организации баз данных.
– Составление подробных алгоритмов.
5. На каком этапе вносится наибольшее количество ошибок в программный код?
– Кодирование.
– Проектирование.
– Автономное тестирование.
– Комплексное тестирование.
– Тестирование в целом.
6. Какие существуют стадии тестирования?
– Системное тестирование.
– Испытательное тестирование.
– Верификационное тестирование.
– Автономное тестирование.
– Комплексное тестирование.
– Аттестационное тестирование.
7. Основные цели методов разработки программного обеспечения:
– развитие методов более точного прогнозирования затрат на создание программного обеспечения;
– анализ исходных требований программного обеспечения;
– разработка методов управления сложными системами;
– повышение надежности и правильности программного обеспечения;
– проектирование более совершенных компиляторов и других средств разработки программного обеспечения.
8. Что описывает понятие «коммуникационный взрыв»?
– Выход оборудования из строя вследствие слишком частого обращения к нему и использования его функций.
– Отказ коммуникационного оборудования вследствие слишком большого числа запросов, превышающего ширину канала связи.
– Чрезмерное усложнение взаимосвязей в коллективе разработчиков, при котором большая часть времени тратится на организацию взаимодействия.
9. Какие существуют методики оценки затрат на разработку?
– Метод экспертных оценок.
– Метод алгоритмического анализа.
– Пошаговый анализ.
– Метод Паркинсона.
– Психологический метод.
10. В чем заключается закон Паркинсона?
– Чаще всего для выполнения некоторой работы (задачи) затрачивается ровно то время, которое отведено для нее.
– Чаще всего для выполнения некоторой работы (задачи) требуется больше времени, чем для нее отведено.
– Стоимость системы, оговариваемая при заключении договора, чаще всего намеренно занижается разработчиком.
– Оценки стоимости системы различными разработчиками чаще всего довольно близки.

14.1.4. Темы лабораторных работ

Написание спецификаций и проведение тестирования ПО. Составление руководства системного программиста.

14.1.5. Методические рекомендации

Учебный материал излагается в форме, предполагающей самостоятельное мышление студентов, самообразование. При этом самостоятельная работа студентов играет решающую роль в ходе всего учебного процесса.

Начать изучение дисциплины необходимо со знакомства с рабочей программой, списком учебно-методического и программного обеспечения. Самостоятельная работа студента включает работу с учебными материалами, выполнение контрольных мероприятий, предусмотренных учеб-

Согласована на портале № 45764

ным планом.

В процессе изучения дисциплины для лучшего освоения материала необходимо регулярно обращаться к рекомендуемой литературе и источникам, указанным в учебных материалах; пользоваться через кабинет студента на сайте Университета образовательными ресурсами электронно-библиотечной системы, а также общедоступными интернет-порталами, содержащими научно-популярные и специализированные материалы, посвященные различным аспектам учебной дисциплины.

При самостоятельном изучении тем следуйте рекомендациям:

- чтение или просмотр материала необходимо осуществлять медленно, выделяя основные идеи; на основании изученного составить тезисы. Освоив материал, попытаться соотнести теорию с примерами из практики;
- если в тексте встречаются термины, следует выяснить их значение для понимания дальнейшего материала;
- необходимо осмысливать прочитанное и изученное, отвечать на предложенные вопросы.

Студенты могут получать индивидуальные консультации с использованием средств телекоммуникации.

По дисциплине могут проводиться дополнительные занятия в форме вебинаров. Расписание вебинаров публикуется в кабинете студента на сайте Университета. Запись вебинара публикуется в электронном курсе по дисциплине.

14.2. Требования к оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусмотрены дополнительные оценочные материалы, перечень которых указан в таблице 14.

Таблица 14 – Дополнительные материалы оценивания для лиц с ограниченными возможностями здоровья и инвалидов

Категории обучающихся	Виды дополнительных оценочных материалов	Формы контроля и оценки результатов обучения
С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами исходя из состояния обучающегося на момент проверки

14.3. Методические рекомендации по оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной форме;
- в печатной форме с увеличенным шрифтом;
- в форме электронного документа;
- методом чтения ассистентом задания вслух;
- предоставление задания с использованием сурдоперевода.

Лицам с ограниченными возможностями здоровья и инвалидам увеличивается время на подготовку ответов на контрольные вопросы. Для таких обучающихся предусматривается доступная форма предоставления ответов на задания, а именно:

Согласована на портале № 45764

– письменно на бумаге;
– набор ответов на компьютере;
– набор ответов с использованием услуг ассистента;
– представление ответов устно.

Процедура оценивания результатов обучения лиц с ограниченными возможностями здоровья и инвалидов по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:
– в форме электронного документа;
– в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:
– в форме электронного документа;
– в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:
– в форме электронного документа;
– в печатной форме.

При необходимости для лиц с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 45764