



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
«Дальневосточный федеральный университет»
(ДВФУ)

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

«СОГЛАСОВАНО»

«УТВЕРЖДАЮ»

Руководитель образовательной программы

Заведующий кафедрой компьютерных систем

Должиков С.В.

18 июня 2018 г.



Кулешов Е.Л.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Технологии программирования

Направление подготовки – 09.03.02 «Информационные системы и технологии»

Профиль «Информационные системы и технологии в связи»

Форма подготовки (очная)

курс 2 семестр 4
лекции 36 час.
практические занятия час.
лабораторные работы 72 час.
в том числе с использованием МАО лек. 0 / пр.0 / лаб. 36 час.
всего часов аудиторной нагрузки – 108 час.
в том числе с использованием МАО – 36 час.
самостоятельная работа 108 час.
контрольные работы (количество)
курсовая работа / курсовой проект – 4 семестр
зачет 4 семестр
экзамен 4 семестр

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования, утвержденного приказом Министерства образования и науки РФ от 12 марта 2015 г. № 219.

Рабочая программа обсуждена на заседании кафедры компьютерных систем, протокол № 14 от 18.06.2015.

Заведующий кафедрой компьютерных систем. д.т.н., профессор Кулешов Е.Л.

Составитель (ли): доцент кафедры компьютерных систем Должиков С.В. к.т.н., доцент

Оборотная сторона титульного листа РПУД

I. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « ____ » _____ 20__ г. № _____

Заведующий кафедрой _____ (подпись) _____ (И.О. Фамилия)

II. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « ____ » _____ 20__ г. № _____

Заведующий кафедрой _____ (подпись) _____ (И.О. Фамилия)

АННОТАЦИЯ

Дисциплина «Технологии программирования» предназначена для бакалавров, обучающихся по направлению подготовки 09.03.02 «Информационные системы и технологии», профиль «Информационные системы и технологии в связи», входит в блок Б1.Б. учебного плана, является базовой дисциплиной базовой части (индекс Б1.Б.18).

Общая трудоемкость освоения дисциплины составляет 6 зачетные единицы, 216 часа. Учебным планом предусмотрены лекционные занятия (36 часов), лабораторные работы (72 часа, том числе с МАО - 36 часов), самостоятельная работа студентов (108 часа, в том числе 45 часов на подготовку к экзамену). Дисциплина реализуется на 2 курсе в 4 семестре. Форма контроля по дисциплине – экзамен, зачет, курсовая работа.

Дисциплина «Технологии программирования» базируется на знаниях, полученных при изучении дисциплин программы бакалавриата по направлению подготовки «Информационные системы и технологии»: «Информатика и основы программирования», «Математика», «Инструментальные средства информационных систем», «Архитектура информационных систем».

Цель освоения дисциплины: сформировать у студентов терминологический фундамент по основам технологии программирования, модульному программированию, методам проектирования: нисходящему и восходящему, отладке, тестированию, верификации, характеристикам качества программ, CASE-технологиям.

Задачи освоения дисциплины:

- изучить принципы объектно-ориентированного проектирования и программирования;
- изучить процессы жизненного цикла программ и современных методов организации разработки программного обеспечения компьютерного;
- приобрести умения и навыки объектно-ориентированного программирования.

- изучить процесс аттестации программного средства и характеристики оценки качества программного средства, особенности этапа конструирования программного средства.

Дисциплина формирует следующие компетенции:

- способность выбирать и оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно-) для решения поставленной задачи (ОПК-6);

- способность к организации работы малых коллективов исполнителей (ПК-19);

- способность обосновывать правильность выбранной модели, сопоставляя результаты экспериментальных данных и полученных решений (ПК-24);

- способность оформлять полученные рабочие результаты в виде презентаций, научно-технических отчетов, статей и докладов на научно-технических конференциях (ПК-26);

- способность формировать новые конкурентоспособные идеи и реализовывать их в проектах (ПК-27).

Код и формулировка компетенции	Этапы формирования компетенции	
ОПК-6, способность выбирать и оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно-) для решения поставленной задачи	знает	основные архитектурные решения современных информационных систем и устройств
	умеет	выбирать и оценивать способы реализации информационных систем и устройств
	владеет	необходимым инструментарием для реализации информационных систем и устройств
ПК-19, способность к организации работы малых коллективов исполнителей	знает	методы работы в коллективе и способы организации работы малых коллективов исполнителей
	умеет	сотрудничать с коллегами по работе

	владеет	навыками организации работы малых коллективов исполнителей
ПК-24, способность обосновывать правильность выбранной модели, сопоставляя результаты экспериментальных данных и полученных решений	знает	основные научные методы анализа данных; основные методы научного познания; методологию разработки и обоснования численных методов решения корректно поставленных математических задач
	умеет	делать правильные выводы из сопоставления результатов теории и эксперимента; делать качественные выводы при переходе к предельным условиям в изучаемых проблемах
	владеет	навыками грамотной обработки результатов опыта и сопоставления их с теоретическими данными; навыками корректного формулирования результатов исследования
ПК-26, способность оформлять полученные рабочие результаты в виде презентаций, научно-технических отчетов, статей и докладов на научно-технических конференциях	знает	принципы использования программных пакетов для осуществления научных исследований
	умеет	самостоятельно разрабатывать и реализовывать алгоритмы для решения научно-исследовательских задач; использовать основные законы естественнонаучных дисциплин
	владеет	навыками выбора компьютерных методов визуализации поведения динамической системы
ПК-27, способность формировать новые конкурентоспособные идеи и реализовывать их в проектах	знает	способы формирования новых конкурентоспособных идей
	умеет	выдвигать новые идеи
	владеет	современными программными пакетами, позволяющими проводить проектирование и моделирование

		реальных ситуаций
--	--	-------------------

Для формирования вышеуказанных компетенций в рамках дисциплины «Технологии программирования» применяются следующие методы активного обучения, интерактивного обучения: лекция-беседа, лекция-конференция, круглый стол (дискуссия).

I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

Трудоемкость теоретической части курса 36 час.

1. Надежное программное средство как продукт технологии программирования. Исторический и социальный контекст программирования. – 2 час.
2. Источники ошибок в программных средствах. – 2 час.
3. Общие принципы разработки программных средств. – 2 час.
4. Внешнее описание программного средства. – 2 час.
5. Методы спецификации семантики функций. – 2 час.
6. Архитектура программного средства. – 2 час.
7. Разработка структуры программы и модульное программирование. – 2 час.
8. Разработка программного модуля. – 2 час.
9. Доказательство свойств программ. – 2 час.
10. Тестирование и отладка программного средства. – 2 час.
11. Обеспечение функциональности и надежности программного средства. – 2 час.
12. Обеспечение качества программного средства. - 2 час.
13. Документирование программных средств. – 3 час.
14. Управление разработкой и аттестации программного средства. – 3 час.
15. Объектный подход к разработке программных средств. – 3 час.
16. Компьютерная поддержка разработки и сопровождения программных средств. – 3 час.

II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА

практические занятия не предусмотрены

Лабораторные работы (72 час.)

1. Изучение возможностей и преимуществ платформы .NET и знакомство основными элементами интерфейса Visual Studio.NET. Основы программирования на языке C#. (4 час.)

2. Цикл 1 Создание консольных приложений с использованием MS Visual Studio.NET. Язык программирования C#. (17 час.)

Лабораторная работа 1. Типы данных.

Лабораторная работа 2. Операторы.

Лабораторная работа 3. Условные операторы.

Лабораторная работа 4. Циклы.

Лабораторная работа 5. Классы, объекты и методы.

Лабораторная работа 6. Конструкторы и деструкторы.

Лабораторная работа 8. Массивы

Лабораторная работа 9. Наследование.

3. Цикл 2. Создание windows приложений с использованием MS Visual Studio .NET. Язык программирования C#. (17 час.)

Лабораторная работа 1. Создание простой формы.

Лабораторная работа 2. Создание информационных окон.

Лабораторная работа 3. Создание форм с использованием обработчика событий.

Лабораторная работа 4. Создание форм с использованием панели ToolBox.

Лабораторная работа 5. Создание форм с использованием наследования.

Лабораторная работа 6. Создание калькулятора.

4. Цикл 3. Основы объектно-ориентированного программирования на языке Visual Basic .NET (17 час.)

Лабораторная работа 1. Создание простых приложений (кнопочных форм).

Лабораторная работа 2. Разработка типовых настольных приложений с использованием Visual Studio .NET.

5. Цикл 4. Разработка системы (приложения) для автоматизации выбранной предметной области. (17 час.)

1. Объектно-ориентированный анализ предметной области.
2. Построение объектной модели.
3. Дальнейшее усовершенствование объектной модели.
4. Построение динамической модели. Диаграмм состояний
5. Построение функциональной модели.
6. Построение диаграмм потоков данных. Описание операций.
7. Разработка архитектуры системы.
8. Разбиение системы на модули (подзадачи).
9. Реализация разработанной системы.

Ш. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Информационная безопасность и защита информации» представлено в Приложении 1 и включает в себя: план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию; характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению; требования к представлению и оформлению результатов самостоятельной работы; критерии оценки выполнения самостоятельной работы.

III. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства - наименование	
				текущий контроль	промежуточная аттестация
1.	Источники ошибок в программных средствах. Общие принципы разработки. Внешнее описание, постановка задачи.	ОПК6 ПК19 ПК24 ПК26 ПК27	знает	Устный опрос (собеседование)	Зачет
2.	Методы спецификации семантики функций. Архитектура ПС. Структура программ. Модульное программирование.	ОПК6 ПК19 ПК24 ПК26 ПК27	знает	Устный опрос (собеседование)	зачет
			Умеет владеет	Индивидуальный проект	
3.	Доказательство свойств программ. Тестирование и отладка программного средства.	ОПК6 ПК19 ПК24 ПК26 ПК27	знает	Устный опрос (собеседование)	зачет
			Умеет владеет	Индивидуальный проект	
4.	Обеспечение качества программного средства. Документирование программных средств. Управление разработкой и аттестации программного средства	ОПК6 ПК19 ПК24 ПК26 ПК27	Знает Умеет владеет	Устный опрос (собеседование) Индивидуальный проект	зачет

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в Приложении 2.

V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

	название	Ссылка в ЭК НБ ДВФУ	Внешняя ссылка
1	Курипта, О. В. Основы программирования и алгоритмизации [Электронный ресурс] : практикум / О. В. Курипта, О. В. Минакова, Д. К. Проскурин. — Электрон. текстовые данные. — Воронеж : Воронежский государственный архитектурно-строительный университет, ЭБС АСВ, 2015. — 133 с. — 978-5-89040-575-3..	http://lib.dvfu.ru:8080/lib/item?id=IPRbooks:IPRbooks-59123&theme=FEFU	http://www.iprbookshop.ru/59123.html
2	Основы информатики. Алгоритмизация и программирование : методические указания для иностранных граждан : учебное пособие / Т. А. Блинова, А. В. Новиков, Н. Н. Руднова ; Московский автомобильно-дорожный государственный технический университет. Москва : Русский язык. Курсы, 2015. 59 с.	http://lib.dvfu.ru:8080/lib/item?id=chamo:790829&theme=FEFU	
3	MATLAB. Программирование на C++, C#, Java и VBA. "ДМК Пресс", 2015. 498 с	http://lib.dvfu.ru:8080/lib/item?id=Lan:Lan-69956&theme=FEFU	
4	[Электронный ресурс] : лабораторный практикум / Е. И. Николаев. — Электрон. текстовые данные. — Ставрополь : Северо-Кавказский федеральный университет, 2015. — 183 с. — 2227-8397.	http://lib.dvfu.ru:8080/lib/item?id=IPRbooks:IPRbooks-62966&theme=FEFU	http://www.iprbookshop.ru/62966
5	Креативное программирование: Учебное пособие / Липовка А.Ю., Бундова Е.С., Жоров Ю.В. - Краснояр.:СФУ, 2015. - 280 с.: ISBN 978-5-7638-3356-0		http://znanium.com/catalog/product/966701

Дополнительная литература

	название	Ссылка в ЭК НБ ДВФУ	Внешняя ссылка
1	Э. Дейкстра. Заметки по структурному программированию / У. Дал, Э. Дейкстра, К. Хоор. Структурное программирование. - М.: Мир, 1975. - С. 7-97.		

2	В. Турский. Методология программирования. - М.: Мир, 1981.		
3	В.В. Липаев, Б.А. Позин, А.А. Штрик. Технология сборочного программирования. – М.: Радио и связь, 1992.		
4	Г.Майерс. Надежность программного обеспечения. - М.: Мир, 1980.		
5	Criteria for evaluation of software. - ISO TC97/SC7 #367 (Supersedes Document #327).		
6	Revised version of DP9126 - Criteria of the evaluation of software quality characteristics. ISO TC97/SC7 #610. Part 6.		
7	К. Зиглер. Методы проектирования программных систем. - М.: Мир, 1985. - С. 15-23. Criteria for Evaluation of Software. ISO TC97/SC7 #383.		

Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. <http://exercism.io/> Этот сайт предоставляет упражнения по программированию, а в частности, на: С #, С ++, Haskell, JavaScript, Lua, Objective-C, Perl 5, Python, Ruby и Scala.
2. <https://www.codewars.com/> Разнообразные задачи, достижения, тесты, просмотр наиболее лучших решений, обсуждения, ранги—все это делает сервис очень удобным и ценным для тех кто хочет порешать задачки на том или ином языке. Сейчас сервис поддерживает следующие языки: Clojure, C++, C#, Elixir, F#, Go, Haskell, Java, JavaScript, PHP, Python, Ruby, Rust, Shell, SQL, Swift, TypeScript.
3. <https://www.learncs.org/> Популярный сайт по языкам программирования.
4. <https://www.dotnetperls.com/> Учебник по C#.

Перечень информационных технологий и программного обеспечения

Место расположения компьютерной техники, на	Перечень программного обеспечения
---	-----------------------------------

<p>котором установлено программное обеспечение</p>	
<p>Владивосток, о. Русский, п. Аякс д.10, корпус L, ауд. L 450 специализированная лаборатория кафедры КС: Лаборатория администрирования информационных систем</p>	<p>Microsoft Office Professional Plus 2019, подтверждающий документ № ЭА-261-18, дата окончания лицензии 30.06.2020.</p> <p>Microsoft Windows 10 Enterprise LTSC 2019, подтверждающий документ № ЭА-261-18, дата окончания лицензии 30.06.2020.</p> <p>Microsoft SQL Server Standard Core 2017, подтверждающий документ № ЭА-261-18, дата окончания лицензии 30.06.2020.</p> <p>Microsoft Office Professional 2003, авторизационный номер лицензиата №18597359ZZE0701, бессрочная лицензия Academic, номер лицензии №18643295.</p> <p>Microsoft Windows 7 Professional, авторизационный номер лицензиата №65541663ZZE1106, бессрочная лицензия Academic, номер лицензии №46260303.</p> <p>Adobe Acrobat Pro DC, подтверждающий документ № ЭА-442-15 от 18.01.16 лот 1, дата окончания лицензии 20.01.2019.</p>

VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Дисциплина изучается в следующих организационных формах: лекционное занятие; лабораторные занятия, самостоятельное изучение теоретического материала; самостоятельное выполнение индивидуального проекта; индивидуальные и групповые консультации.

Основной формой самостоятельной работы студента является изучение конспекта лекций, их дополнение рекомендованной литературой, выполнение проекта, а также активная работа на лабораторных и практических занятиях.

К прослушиванию лекции следует готовиться, для этого необходимо знать программу курса и рекомендованную литературу. Тогда в процессе лекции легче отделить главное от второстепенного, легче сориентироваться в процессе: что записать, что самостоятельно проработать, что является трудным для понимания, а что легко усвоить.

Контроль за выполнением самостоятельной работы студента производится в виде контроля каждого этапа работы, отраженного в документации и защиты проекта.

Студент должен планировать график самостоятельной работы по дисциплине и придерживаться его.

VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для проведения исследований, связанных с выполнением задания по практике, а также для организации самостоятельной работы студентам доступно следующее лабораторное оборудование и специализированные кабинеты, соответствующие действующим санитарным и противопожарным нормам, а также требованиям техники безопасности при проведении учебных и научно-производственных работ:

Наименование оборудованных помещений и помещений для самостоятельной работы	Перечень основного оборудования
Владивосток, о. Русский, п. Аякс д.10, корпус L, ауд. L 558 учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Парты и стулья
Владивосток, о. Русский, п. Аякс д.10, корпус L, ауд. L 450 специализированная лаборатория кафедры КС: Лаборатория администрирования информационных систем	11 компьютеров (системный блок модель - 30AGCT01WW P3+монитором АОС 28" LI2868POU)
Владивосток, о. Русский, п. Аякс д.10, корпус L, ауд. L 507 специализированная лаборатория кафедры КС: Лаборатория микропроцессорной техники	Стеллажи, столы и стулья
Читальные залы Научной библиотеки ДВФУ с открытым доступом к фонду	Моноблок HP ProOne 400 All-in-One 19,5 (1600x900), Core i3-4150T, 4GB DDR3-1600 (1x4GB), 1TB HDD 7200 SATA, DVD+/-RW,GigEth,Wi-Fi,BT,usb kbd/mse,Win7Pro

(корпус А - уровень 10) Учебная аудитория для проведения самостоятельной работы и подготовки к экзамену	(64-bit)+Win8.1Pro(64-bit),1-1-1 Wty Скорость доступа в Интернет 500 Мбит/сек. Рабочие места для людей с ограниченными возможностями здоровья оснащены дисплеями и принтерами Брайля; оборудованы: портативными устройствами для чтения плоскочечатных текстов, сканирующими и читающими машинами видеувелечителем с возможностью регуляции цветовых спектров; увеличивающими электронными лупами и ультразвуковыми маркировщиками
--	--

В целях обеспечения специальных условий обучения инвалидов и лиц с ограниченными возможностями здоровья в ДВФУ все здания оборудованы пандусами, лифтами, подъемниками, специализированными местами, оснащенными туалетными комнатами, табличками информационно-навигационной поддержки.



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

**УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ
САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ**

Технологии программирования
Направление подготовки – **09.03.02 «Информационные системы и
технологии»**

Профиль «Информационные системы и технологии в связи»

Форма подготовки (очная)

**Владивосток
2015**

Самостоятельная работа студентов состоит из подготовки к практическим занятиям, работы над рекомендованной литературой, решения задач и написания компьютерных графических программ. При организации самостоятельной работы преподаватель должен учитывать уровень подготовки каждого студента. Преподаватель дает каждому студенту индивидуальные и дифференцированные задания.

п/п	Вид самостоятельной работы	Дата/сроки выполнения	Примерные нормы времени на выполнение	Форма контроля
	Источники ошибок в программных средствах. Общие принципы разработки. Внешнее описание, постановка задачи.	1-3 неделя обучения	10 часов	ПР-9
	Методы спецификации семантики функций. Архитектура ПС. Структура программ. Модульное программирование.	4-7 неделя обучения	10 часов	ПР-9
	Доказательство свойств программ. Тестирование и отладка программного средства.	8-12 неделя обучения	12 часов	ПР-9
	Обеспечение качества программного средства. Документирование программных средств. Управление разработкой и аттестации программного средства	13-18 неделя обучения	13 часов	ПР-9

Задания для самостоятельного выполнения

1. Знакомство с рекомендованной научной и научно-популярной литературой по проблемам информационной безопасности и защите информации.
2. Составление глоссария терминов.
3. Знакомство с широко применяемыми программными продуктами защиты информации.

Рекомендации по работе с литературой

Для более эффективного освоения и усвоения материала рекомендуется ознакомиться с теоретическим материалом по той или иной теме до проведения лабораторного занятия. Всю учебную литературу желательно изучать «под конспект».

Цель написания конспекта по дисциплине – сформировать навыки по поиску, отбору, анализу и формулированию учебного материала.

Работу с теоретическим материалом по теме можно проводить по следующей схеме:

- название темы;
- цели и задачи изучения темы;
- основные вопросы темы;
- характеристика основных понятий и определений, необходимых для усвоения данной темы;
- краткие выводы, ориентирующие на определенную совокупность сведений, основных идей, ключевых положений, систему доказательств, которые необходимо усвоить.

При работе над конспектом обязательно выявляются и отмечаются трудные для самостоятельного изучения вопросы, с которыми уместно обратиться к преподавателю при посещении консультаций, либо в индивидуальном порядке.

Подготовка к лабораторным занятиям

Подготовку к каждому практическому занятию каждый студент должен начать с изучения теоретического материала и ознакомления с планом, который отражает содержание предложенной темы. Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса. Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы по теме задания, правильном выполнении лабораторной работы.

В процессе занятия студент должен создать требуемый документ с помощью предлагаемого программного средства и выполнить требуемые в задании операции. Задание по лабораторной работе содержит методические указания по подготовке документа, который должен быть получен в результате выполнения работы. При подготовке к лабораторной работе следует их внимательно прочесть.

Методические указания к составлению глоссария

Глоссарий охватывает термины в рамках тематики, затрагиваемой в лекциях. Глоссарий должен содержать не менее 50 терминов, они должны быть перечислены в алфавитном порядке, соблюдена нумерация. Глоссарий должен быть оформлен по принципу реферативной работы, в обязательном порядке присутствует титульный лист и нумерация страниц. Объем работы

должен составлять 10-15 страниц. Тщательно проработанный глоссарий помогает избежать разночтений и помочь углубленному изучению материала. Глоссарии могут содержать отдельные слова, фразы, аббревиатуры, слоганы и даже целые предложения.

Критерии оценки отчетов по проектам

– 100-86 баллов выставляется, если содержание и составляющие части соответствуют выданному заданию. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 85-76 - баллов выставляется, если при выполнении задания допущено не более одной ошибки. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 75-61 балл выставляется, если при выполнении задания допущено не более двух ошибок. Продемонстрировано знание и владение навыками подготовки документа по теме. Допущено не более 2 ошибок, связанных с пониманием структуры и содержания задания.

– 60-50 баллов - если структура и содержание задания не соответствуют требуемым.



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Технологии программирования
Направление подготовки – 09.03.02 «Информационные системы и технологии»
Профиль «Информационные системы и технологии в связи»

Форма подготовки (очная)

Паспорт ФОС

Код и формулировка компетенции	Этапы формирования компетенции	
ОПК-6, способностью выбирать и оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно-) для решения поставленной задачи	знает	основные архитектурные решения современных информационных систем и устройств
	умеет	выбирать и оценивать способы реализации информационных систем и устройств
	владеет	необходимым инструментарием для реализации информационных систем и устройств
ПК-19, способностью к организации работы малых коллективов исполнителей	знает	методы работы в коллективе и способы организации работы малых коллективов исполнителей
	умеет	сотрудничать с коллегами по работе
	владеет	навыками организации работы малых коллективов исполнителей
ПК-24, способностью обосновывать правильность выбранной модели, сопоставляя результаты экспериментальных данных и полученных решений	знает	основные научные методы анализа данных; основные методы научного познания; методологию разработки и обоснования численных методов решения корректно поставленных математических задач
	умеет	делать правильные выводы из сопоставления результатов теории и эксперимента; делать качественные выводы при переходе к предельным условиям в изучаемых проблемах
	владеет	навыками грамотной обработки результатов опыта и сопоставления их с теоретическими данными; навыками корректного формулирования результатов исследования
ПК-26, способностью оформлять полученные рабочие результаты в виде презентаций, научно-технических отчетов, статей и докладов на научно-технических конференциях	знает	принципы использования программных пакетов для осуществления научных исследований
	умеет	самостоятельно разрабатывать и реализовывать алгоритмы для решения научно-исследовательских задач; использовать основные законы естественнонаучных дисциплин
	владеет	навыками выбора компьютерных методов визуализации поведения динамической системы
ПК-27, способностью формировать новые	знает	способы формирования новых конкурентоспособных идей

конкурентоспособные идеи и реализовывать их в проектах	умеет	выдвигать новые идеи
	владеет	современными программными пакетами, позволяющими проводить проектирование и моделирование реальных ситуаций

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства - наименование	
				текущий контроль	промежуточная аттестация
1	Источники ошибок в программных средствах. Общие принципы разработки. Внешнее описание, постановка задачи.	ОПК6 ПК19 ПК24 ПК26 ПК27	знает	Устный опрос (собеседование)	Зачет
2.	Методы спецификации семантики функций. Архитектура ПС. Структура программ. Модульное программирование. Доказательство свойств программ. Тестирование и отладка программного средства.	ОПК6 ПК19 ПК24 ПК26 ПК27	знает	Устный опрос (собеседование)	зачет
			Умеет владеет	Индивидуальный проект	
3.	Обеспечение качества программного средства. Документирование программных средств. Управление разработкой и аттестации программного средства Источники ошибок в программных средствах. Общие принципы разработки. Внешнее описание, постановка задачи.	ОПК6 ПК19 ПК24 ПК26 ПК27	знает	Устный опрос (собеседование)	зачет
			Умеет владеет	Индивидуальный проект	
4.	Методы спецификации семантики функций. Архитектура ПС. Структура программ. Модульное программирование.	ОПК6 ПК19 ПК24 ПК26 ПК27	Знает	Устный опрос (собеседование)	зачет
			Умеет владеет	Индивидуальный проект	

Шкала оценивания уровня сформированности компетенций

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
	знает (пороговый уровень)	основные архитектурные решения современных		
ОПК-6, способностью выбирать и	знает (пороговый уровень)	основные архитектурные решения современных	основные архитектурные решения	Способность дать ответы на вопросы

оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно-) для решения поставленной задачи		информационных систем и устройств	современных информационных систем и устройств	
	умеет (продвинутый)	выбирать и оценивать способы реализации информационных систем и устройств	выбирать и оценивать способы реализации информационных систем и устройств	Способность продемонстрировать операции и дать к ним пояснения
	владеет (высокий)	необходимым инструментарием для реализации информационных систем и устройств	необходимым инструментарием для реализации информационных систем и устройств	Способность классифицировать объекты
ПК-19, способностью к организации работы малых коллективов исполнителей	знает (пороговый уровень)	методы работы в коллективе и способы организации работы малых коллективов исполнителей	Знание методов работы	Способность дать ответы на вопросы
	умеет (продвинутый)	сотрудничать с коллегами по работе	Умение создавать программы в коллективе	Наличие созданных программ
	владеет (высокий)	навыками организации работы малых коллективов исполнителей	Владение методами определения возможностей членов коллектива	Наличие созданных программ
ПК-24, способностью обосновывать правильность выбранной модели, сопоставляя результаты экспериментальных данных и полученных решений	знает (пороговый уровень)	основные научные методы анализа данных; основные методы научного познания; методологию разработки и обоснования численных методов решения корректно поставленных математических задач	Знание методов работы	Способность дать ответы на вопросы
	умеет (продвинутый)	делать правильные выводы из сопоставления результатов теории и эксперимента; делать качественные выводы при переходе к предельным	Умение создавать программы	Наличие созданных программ

		условиям в изучаемых проблемах		
	владеет (высокий)	навыками грамотной обработки результатов опыта и сопоставления их с теоретическими данными; навыками корректного формулирования результатов исследования	Владение методами определения	Наличие созданных программ
ПК-26, способностью оформлять полученные рабочие результаты в виде презентаций, научно- технических отчетов, статей и докладов на научно- технических конференциях	знает (пороговый уровень)	принципы использования программных пакетов для осуществления научных исследований	Знание методов работы	Способность дать ответы на вопросы
	умеет (продвину тый)	самостоятельно разрабатывать и реализовывать алгоритмы для решения научно- исследовательских задач; использовать основные законы естественнонаучных дисциплин	Умение создавать программы	Наличие созданных программ
	владеет (высокий)	навыками выбора компьютерных методов визуализации поведения динамической системы	Владение методами определения	Наличие созданных программ
ПК-27, способностью формировать новые конкурентоспособн ые идеи и реализовывать их в проектах	знает (пороговый уровень)	способы формирования новых конкурентоспособных идей	Знание методов формирования	Способность дать ответы на вопросы
	умеет (продвину тый)	выдвигать новые идеи	Умение создавать программы	Наличие созданных программ
	владеет (высокий)	современными программными пакетами, позволяющими проводить проектирование и моделирование реальных ситуаций	Владение методами и современными пакетами	Наличие созданных программ

Вопросы к экзамену

1. Модель перевода и источники ошибок при разработке программных средств.
2. Специфические особенности разработки программных средств.
3. Жизненный цикл программного средства.
4. Определение требований к программному средству.
5. Спецификация качества программного средства
6. Функциональная спецификация программного средства.
7. Контроль внешнего описания программного средства.
8. Табличный подход к спецификации семантики функций. Метод таблиц решений.
9. Основные классы архитектур программных средств.
10. Понятие программного модуля и его основные характеристики.
11. Методы разработки структуры программ.
12. Метод целенаправленной конструктивной реализации.
13. Структурное программирование и пошаговая детализация. Понятие о псевдокоде.
14. Правила для установления свойств составного и условного операторов.
15. Инвариант цикла. Правило для установления свойств оператора цикла.
16. Понятие о завершаемости выполнения программы. Правило для установления завершаемости выполнения цикла.
17. Заповеди отладки программных средств.
18. Автономная отладка и тестирование программного средства.
19. Комплексная отладка и тестирование программного средства.
20. Обеспечение устойчивости программного модуля.
21. Обеспечение защиты от влияния «чужих» программ.
22. Обеспечение защиты от несанкционированного доступа к программным средствам и защиты от взлома защиты.
23. Обеспечение легкости применения программного средства.
24. Обеспечение эффективности программного средства.

Критерии выставления оценки студенту

Баллы (рейтинговая оценки)	Оценка зачета/ экзамена (стандартная)	Требования к сформированным компетенциям
86-100	«зачтено»/ «отлично»	Оценка «отлично» (зачтено) выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
76-85	«зачтено»/ «хорошо»	Оценка «хорошо» (зачтено) выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.
61-75	«зачтено»/ «удовлетворительно»	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.
0-60	«не зачтено»/ «неудовлетворительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

Текущий контроль

Текущая аттестация студентов проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной.

Текущая аттестация проводится в форме защиты проекта и осуществляется ведущим преподавателем.

Объектами оценивания выступают:

- степень усвоения теоретических знаний - оценивается в форме

собеседования;

- уровень овладения практическими умениями и навыками – оценивается в форме защиты проекта.

Критерии оценки проектов

- 100-86 баллов выставляется, если студент/группа точно определили содержание и составляющие части задания, умеют аргументированно отвечать на вопросы, связанные с заданием. Продемонстрировано знание и владение навыками самостоятельной исследовательской работы по теме. Фактических ошибок, связанных с пониманием проблемы, нет.

- 85-76 - баллов - работа студента/группы характеризуется смысловой цельностью, связностью и последовательностью изложения; допущено не более 1 ошибки при объяснении смысла или содержания проблемы. Продемонстрированы исследовательские умения и навыки. Фактических ошибок, связанных с пониманием проблемы, нет.

- 75-61 балл – проведен достаточно самостоятельный анализ основных этапов и смысловых составляющих проблемы; понимание базовых основ и теоретического обоснования выбранной темы. Привлечены основные источники по рассматриваемой теме. Допущено не более 2 ошибок в смысле или содержании проблемы

- 60-50 баллов - если работа представляет собой пересказанный или полностью переписанный исходный текст без каких бы то ни было комментариев, анализа. Не раскрыта структура и теоретическая составляющая темы. Допущено три или более трех ошибок смыслового содержания раскрываемой проблемы

Шкала оценивания

Менее 60 баллов	незачтено	неудовлетворительно
От 61 до 75 баллов	зачтено	удовлетворительно
От 76 до 85 баллов	зачтено	хорошо
От 86 до 100 баллов	зачтено	отлично

Оценочные средства для текущей аттестации

Тестовые задания для текущего контроля

Тесты

__ язык - язык программирования высокого уровня, предназначенный для решения определенного класса задач

- проблемно-ориентированный
- машино-ориентированный
- проблемно-независимый
- проблемно-зависимый

___ язык программирования - язык взаимодействия пользователя с системой программирования, реализуемый диалоговыми средствами графического интерфейса пользователя

- дисплейный
- визуальный
- диалоговый
- мониторный

___ алгоритма - графическое представление алгоритма, которое дополняется элементами словесной записи

- схема
- диаграмма
- изображение
- запись

___ дерево - это динамическая структура данных, состоящая из узлов, каждый из которых содержит, кроме данных, не более двух ссылок на различные бинарные деревья

- узловое
- бинарное
- корневое
- иерархическое

___ - единица обмена данными между программой и внешней памятью (винчестер, магнитные диски и ленты)

- элемент
- строка
- запись
- объект

___ - информация, размещаемая на внешних запоминающих устройствах, снабженная идентификатором и оформленная как единое целое средствами операционной системы или языка программирования

- файл
- структура
- блок
- запись

___ метрики прямо измеряют программный продукт и процесс его разработки

- объектно-ориентированные
- процедурно-ориентированные
- функционально-ориентированные
- размерно-ориентированные

___ модель предполагает организацию проектирования программного обеспечения сверху вниз, когда сначала определяется состав функциональных подсистем, а затем осуществляется постановка отдельных задач

- спиральная
- каскадная
- итерационная
- комбинированная

___ модель предполагает последовательный переход на следующий этап после завершения предыдущего

- итерационная
- комбинированная
- спиральная
- каскадная

___ называется алгоритм решения некоторой подзадачи из основной решаемой задачи

- вспомогательным
- дополнительным
- рекурсивным
- второстепенным

___ называется совокупность однотипных элементов, рассматриваемых как единое целое

- массивом
- записью
- строкой
- множеством

___ называются такие величины, память под которые выделяется во время компиляции и сохраняется в течение всей работы программы

- сохраняемыми
- статическими
- независимыми
- постоянными

___ оболочки - специальные программы, предназначенные для облегчения общения пользователя с командами операционной системы

- организационные
- управляющие
- функциональные
- операционные

___ осуществляют пооперационную обработку и выполнение программы

- компиляторы
- трансляторы (интерпретаторы)
- конструкторы
- генераторы

___ - поименованная часть программы (блок программы), которая может выполнять некоторые четко заданные действия над условными данными, определяемыми с помощью формальных параметров

- функция
- модуль
- подпрограмма
- процедура

___ - программное обеспечение для планирования рабочего времени, составления протоколов встреч, расписаний, ведения записей и телефонной книжки

- органайзер
- диспетчеризатор
- редактор
- интерпретатор

___ - смысловое содержание конструкций, предложений языка

- семантика
- синтаксис
- синтезатор
- конструктор

___ - совокупность правил образования цепочек символов (лексем), образующих идентификаторы (переменные и метки), операторы, операции и другие лексические компоненты языка

- лексика
- синтаксис
- анализатор
- семантика

___ - совокупность правил образования языковых конструкций, или предложений языка программирования - блоков, процедур, составных операторов, условных операторов, операторов цикла и др.

- синтаксис
- транслятор
- компилятор
- семантика

___ - совокупность связанных данных, состоящая из фиксированного числа элементов одного типа

- записи
- массивы
- строки
- множества

___ - совокупность символов, отображаемых на устройствах печати и экранах и/или вводимых с клавиатуры терминала

- алфавит
- словарь
- лексика
- глоссарий

___ - содержание и последовательность операций, точно определяющие решение задачи путем вычислительного процесса, преобразующего исходные данные в конечный результат

- диспетчер
- алгоритм
- загрузчик
- планировщик

___ - специальные программы, предназначенные для трассировки и анализа выполнения других программ

- анализаторы
- отладчики
- трассировщики
- интеграторы

___ структура алгоритма - определенный набор блоков и стандартных способов их соединения для выполнения типичных последовательных действий

- основная
- заданная

- типичная
 - базовая
- ___ - структура данных, состоящая из некоторого числа однородных элементов, каждый из которых содержит указатель на следующий элемент**

- очередь
 - список
 - запись
 - стек
- ___ технологии конструирования программного обеспечения обеспечивают автоматизированную или автоматическую поддержку методов**

- оболочки
 - утилиты
 - функции
 - процедуры
- ___ технологии конструирования программного обеспечения - последовательность шагов, использующих методы, утилиты и процедуры**

- инжиниринги
 - планировщики
 - диспетчеры
 - парадигмы
- ___ технологии программирования - это программные продукты, предназначенные для поддержки технологии программирования**

- средства
 - среда
 - оболочка
 - инструментарий
- ___ тип данных определяется только через операции, которые могут выполняться над соответствующими объектами безотносительно к способу представления этих объектов**

- объектный
 - безотносительный
 - абстрактный
 - операционный
- ___ транслирует всю программу без ее выполнения**

- коммутатор
- компилятор
- оптимизатор
- интерпретатор

___ - это динамическая структура данных, добавление элементов в которую выполняется в один конец, а выборка - из другого конца

- очередь
- стек
- запись
- список

___ - это процесс создания модели требуемого программного продукта для снятия неопределенности в требованиях заказчика

- макетирование
- разработка
- конструирование
- программирование

___ - это такой способ организации вычислительного процесса, при котором подпрограмма (процедура или функция) в ходе выполнения составляющих ее операторов обращается сама к себе

- переход
- рекурсия
- возврат
- итерация

___ - это упорядоченный набор элементов данных, в котором можно удалять и добавлять элементы, причем новый элемент всегда записывается в его конец

- запись
- список
- очередь
- стек

___ - это элементарный процесс, различаемый пользователем и перемещающий данные между внешней средой и программным приложением

- драйвер
- интерфейс
- пересылка
- транзакция

___ являются «клеем», который соединяет методы и утилиты так, что они обеспечивают непрерывную технологическую цепочку разработки

- функции
- оболочки
- интерфейсы
- процедуры

___ язык - язык программирования, который отражает архитектуру данной ЭВМ или данного класса ЭВМ

- машинно-ориентированный
- машино-классный
- машино-архитектурный
- машино-независимый

___ язык - язык программирования, структура и средства которого не связаны с конкретной ЭВМ и позволяют выполнять составленные на нем программы на любых ЭВМ, снабженных трансляторами с этого языка

- машино-конкретный
- машино-транслируемый
- машинно-независимый
- машинно-зависимый

___ метрики косвенно измеряют программный продукт и процесс его разработки

- размерно-ориентированные
- объектно-ориентированные
- функционально-ориентированные
- процедурно-ориентированные

___-технология - программный комплекс, автоматизирующий весь технологический процесс анализа, проектирования, разработки и сопровождения сложных программных систем

- NETWORK
- CASE
- INTERNET
- ETHERNET

Алгоритмы, в соответствии с которыми решение поставленных задач сводится к арифметическим действиям, называются ___ алгоритмами

- арифметическими
- математическими
- алгебраическими
- численными

В __ системе программирования осуществляется подготовка результирующего модуля, который может выполняться на ЭВМ практически независимо от программно-аппаратурной среды

- многоязыковой
- замкнутой
- интерпретирующей
- компилирующей

В __ систему программирования можно ввести новый входной язык с транслятором, не требуя изменений в системе

- открытую
- компилирующую
- одноязыковую
- интерпретирующую

В __ системе программирования осуществляется покомандная расшифровка и выполнение инструкций входного языка (в среде данной системы программирования)

- замкнутой
- компилирующей
- интерпретирующей
- открытой

Ввод-вывод данных выполняется через __ - специальную область оперативной памяти

- бампер
- буфер
- плату
- интерфейс

Величины, имеющие ссылочный тип, называются

- ссылками
- идентификаторами
- адресаторами
- указателями

Дерево является __ структурой данных, поскольку каждое поддереву также является деревом

- линейной

- итерационной
- рекурсивной
- списочной

Для многократного повторения одних и тех же действий

предусмотрен оператор

- IF
- CASE
- WITH
- FOR

Для описание функций каждого модуля и условий его сопряжения

с другими модулями применяется язык

- спецификаций
- функций
- процедур
- модулей

Для программирования ветвящихся алгоритмов используется

оператор

- FOR
- REPEAT
- WHILE
- CASE

Достоинство макетирования -

- принятие заказчиком макета за продукт
- определение полных требований к программному обеспечению
- принятие разработчиком макета за продукт
- простота реализации

Иерархическая модульная структура программного средства

обеспечивает технологию проектирования сложных программных

средств по принципу

- последовательности
- сверху - вниз
- передачи управления
- снизу-вверх

К базовым алгоритмическим структурам относятся: 1) следование;

2) переход; 3) ветвление; 4) цикл; 5) передача

- 1, 3, 4
- 2, 5
- 1, 2
- 3, 4, 5

К возможным источникам ограничений системы относятся: 1) технические; 2) системные; 3) эксплуатационные; 4) квалификационные; 5) психологические

- 2, 4
- 4, 5
- 1, 3, 5
- 1, 2, 3

К основным структурам алгоритмов относятся: 1) линейные; 2) разветвляющиеся; 3) циклические; 4) графические; 5) повторяющиеся

- 2, 5
- 1, 2, 3
- 1, 3, 4
- 4, 5

К программному обеспечению относятся: 1) системы автоматического управления; 2) операционные системы; 3) системы программирования; 4) пакеты прикладных программ; 5) системы безопасности

- 4, 5
- 1, 5
- 2, 3, 4
- 1, 2, 3

К технологии конструирования программного обеспечения относятся: 1) методы; 2) способы; 3) средства; 4) возможности; 5) процедуры

- 1, 2, 3
- 3, 4, 5
- 1, 3, 5
- 2, 4

К типам данных относятся: 1) сложные; 2) функциональные; 3) простые; 4) структурированные; 5) процедурные

- 3, 4, 5
- 1, 2, 3
- 2, 4, 5
- 1, 2

Макетирование основывается на

- многократном тестировании
- диалоге заказчика и разработчика

- диалоге разработчика и ЭВМ
 - многократном повторении итераций, в которых участвуют заказчик и разработчик
- Модель может принимать следующие формы: 1) бумажный макет; 2) отладочный макет; 3) работающий макет; 4) тестируемая программа; 5) существующая программа**

- 2, 4
- 1, 2
- 3, 4, 5
- 1, 3, 5

Операционные системы могут быть: 1) однопользовательские; 2) интерактивные; 3) многозадачные; 4) функциональные; 5) инструментальные

- 4, 5
- 2, 4
- 1, 2, 3
- 1, 3, 5

Организация цикла, когда его тело расположено перед проверкой условия, носит название цикла с

- постусловием
- предусловием
- возвратом
- параметрами

Основными функциями операционной системы являются: 1) управление данными; 2) управление доступом; 3) управление задачами (заданиями, процессами); 4) управление пропускной способностью 5) связь с человеком-оператором

- 3, 4, 5
- 1, 2, 3
- 1, 3, 5
- 2, 4

Отличительной особенностью __ системы программирования является то, что отдельные части программы могут быть подготовлены на различных языках программирования и объединены в единый модуль

- компилирующей
- открытой

- многоязыковой
- интерпретирующей

Очередь реализует принцип обслуживания

- FIFO
- FIFO
- LIFO
- LOFO

Пользовательские типы создаются на основе __ типов

- порядковых
- простых
- стандартных
- строчных

Примерно __ своего рабочего времени разработчик тратит на

кодирование

- 20%
- 40%
- 30%
- 60%

Примерно __ своего рабочего времени разработчик тратит на

тестирование и отладку

- 15%
- 20%
- 40%
- 30%

Согласно __ -технологии программное обеспечение разрабатывается путем расширения программных прототипов, повторяя путь от детализации требований к детализации программного кода

- CASE
- COM
- RAD
- UML

Стратегиями конструирования программного обеспечения являются: 1) революционная стратегия; 2) многократный проход; 3) однократный проход; 4) инкрементная стратегия; 5) эволюционная стратегия

- 2, 4, 5
- 3, 4, 5

- 1, 2
- 1, 2, 3

Существуют следующие типы систем программирования: 1) проектирующие; 2) симметричные; 3) одноязыковые; 4) замкнутые; 5) интерпретирующие

- 3, 4, 5
- 1, 2
- 1, 2, 3
- 2, 4, 5

Технология __ должна обеспечивать возможность создания модулей и групп программ с четким описанием функций и взаимодействия с другими компонентами, а также гарантии качества выполнения этих функций, стандартов и соглашений о связях

- программирования
- разработки компонент
- конструирования
- сборки программного средства из подготовленных компонент

Технология __ использует программные и информационные компоненты как законченные комплектующие изделия и должна обеспечивать их корректную сборку и сопряжение в сложных программных средствах

- сборки программного средства из подготовленных компонент
- конструирования
- разработки компонент
- программирования

Управление __ - это систематический подход к выявлению, организации и документированию требований к системе

- заданиями
- проектами
- спецификациями
- требованиями

Характеристики стратегий конструирования программного обеспечения соответствуют требованиям стандарта

- IEEE/ISO 12200.2
- IEEE/EIA 12200.2
- IEEE/EIA 12207.2
- IEEE/ISO 12207.2

Цикл с ___ - цикл, при котором сначала вычисляется некоторое логическое выражение P , в случае истинности которого выполняется тело цикла (оператор S)

- предусловием
- постусловием
- переходом
- параметрами

Языки спецификации относятся к ___информационным языкам, созданным на базе естественных языков путем наложения ограничений на их лексику и грамматику, а также путем применения специальных обозначений для элементов этих языков

- формализованным
- формальным
- фактическим
- форматным

Примеры вариантов тестовых заданий с ответами

1 вариант

№	Вопрос	Ответ
1	Какой этап жизненного цикла программы выполняется вначале? а) определение стоимости б) техническое задание в) оценка осуществимости проекта	в)
2	Процесс перевода программного продукта с одного языка программирования на другой: а) отладка б) сопровождение в) реинжиниринг	в)
3	Какие программы можно отнести к системному программному обеспечению: а) операционные системы б) прикладные программы; в) игровые программы	а)

2 вариант

№	Вопрос	Ответ
1	API-интерфейс (Application Program Interface) строится на основе: а) диаграммы классов б) диаграммы объектов в) диаграммы случаев (интерфейсов)	а)
2	В стадии разработки программы не входит: а) автоматизация программирования; б) постановка задачи; в) составление спецификаций; г) эскизный проект; д) тестирование.	а)
3	Последовательность этапов программирования: а) компилирование, компоновка, отладка; б) компоновка, отладка, компилирование; в) отладка, компилирование, компоновка; г) компилирование, отладка, компоновка.	а)