МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
Томский государственный университет систем управления и радиоэлектроники

УТВЕРЖДАЮ
Проректор по учебной работе
_____ **П.Е. Троян**
«12» 05 2016 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

Уровень основной образовательной программы: бакалавриат

Направление(я) подготовки (специальность): 09.03.03 (230700.62) – Прикладная информатика

Профиль: Прикладная информатика в экономике

Форма обучения: очная

Факультет: ФСУ, Факультет систем управления

Кафедра: АСУ, Кафедра автоматизированных систем управления

Курс 2, 3

Семестр 4, 5

Учебный план набора 2013, 2014, 2015 и последующих лет

Распределение рабочего времени:

Виды учебной работы	Семестр 4	Семестр 5	Всего	Единицы
Лекции	18	18	36	час.
Лабораторные работы	36	18	54	час.
Практические занятия	–	18	18	час.
Курсовой проект/работа (КРС) (аудиторная)	–	14	14	час.
Всего аудиторных занятий	54	68	122	час.
Из них в интерактивной форме	12	14	26	час.
Самостоятельная работа студентов (СРС)	54	76	130	час.
Всего (без экзамена)	108	144	252	час.
Самост. работа на подготовку и сдачу экзамена	–	36	36	час.
Общая трудоемкость	108	180	288	час.
(в зачетных единицах)	3	5	8	ЗЕТ

Зачет 4 семестр, экзамен 5 семестр, диф. зачет 5 семестр

Томск 2016

Согласована на портале № 7359

протокол № 2 от « 5 » 12 20 16 г.

Рабочая программа составлена с учетом требований Федерального Государственного образовательного стандарта высшего профессионального образования (ФГОС ВО) по направлению подготовки 09.03.03 Прикладная информатика (квалификация (степень) "бакалавр"), утвержденного Приказом Министерства образования и науки Российской Федерации от 12 марта 2015 г. № 207, рассмотрена и утверждена на заседании кафедры 12 февраля 2016 г., протокол № 5.

Разработчик, к.ф,-м.н., доцент каф. АСУ _____ С.Л. Миньков

Зав. обеспечивающей кафедрой АСУ
д.т.н., профессор _____ А.М. Кориков

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами специальности.

Декан, к.т.н., доцент _____ П.В. Сенченко

Заведующий профилирующей и выпускающей
кафедрой АСУ, д.т.н., профессор _____ А.М. Кориков

Эксперты:
Кафедра АСУ, доцент _____ А.И. Исакова
(место работы) (занимаемая должность)

Согласована на портале № 7359

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Дисциплина «Программная инженерия» читается в 4 и 5 семестрах и предусматривает чтение лекций, проведение лабораторных работ, выполнение курсовой работы, самостоятельных работ и проведение консультаций.

Цель дисциплины «Программная инженерия» – подготовить обучающихся к проектно-технологической деятельности в области создания компонентов программных комплексов и баз данных, автоматизации технологических процессов с использованием современных инструментальных средств и технологий.

Задачами дисциплины являются изучение основных программистских и управленческих принципов конструирования программных средств, знакомство с концепциями, методологиями, стандартами разработки программного обеспечения на всех этапах его жизненного цикла, обучение методам командной работы в проектных группах по созданию программного обеспечения.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Дисциплина «Программная инженерия» относится к числу дисциплин базовой части профессионального цикла (Б3.Б.3). Успешное овладение данной дисциплиной предполагает предварительные теоретические знания по классификации и структуре информационных систем и технологий, практические навыки по работе в инструментальных средах программирования, полученные в дисциплинах «Информационные системы и технологии» (Б3.Б.4), «Информатика и программирование» (Б2.Б.5), «Объектно-ориентированное программирование» (Б3.В.ОД.2), «Проектный менеджмент» (Б2.В.ДВ.2.1), «Информационный менеджмент» (Б2.В.ДВ.2.2).

Студенты смогут использовать полученные знания при изучении дисциплин «Проектный практикум» (Б3.Б.8), «Проектирование экономических информационных систем» (Б3.В.ДВ.1.1, Б3.В.ДВ.2.1,), подготовке выпускных квалификационных работ.

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины «Программная инженерия» направлен на формирование следующих компетенций:

профессиональные компетенции (ПК):

– способность программировать приложения и создавать программные прототипы решения прикладных задач (ПК-8);

– способность осуществлять и обосновывать выбор проектных решений по видам обеспечения информационных систем (ПК-20);

В результате освоения дисциплины бакалавр по направлению подготовки «Прикладная информатика» должен:

знать

– функциональные и технологические стандарты разработки программных комплексов;

– принципы организации проектирования и содержание этапов разработки программных комплексов;

– задачи и методы исследования и обеспечения качества и надежности программных компонентов;

уметь

– формулировать требования к создаваемым программным комплексам;

– формировать архитектуру программных комплексов для информатизации предприятий, разрабатывать программные приложения;

– использовать международные и отечественные стандарты жизненного цикла программного обеспечения.

владеть

Согласована на портале № 7359

– навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины составляет **8** зачетных единиц.

Вид учебной работы	Семестр 4	Семестр 5	Всего часов
Аудиторные занятия (всего)	**54**	**68**	**122**
В том числе:			
Лекции	18	18	36
Практические занятия (ПЗ)	–	18	18
Семинары (С)	–	–	–
Лабораторные работы (ЛР)	36	18	54
Контроль самостоятельной работы студентов (КСР)	–	14	14
Самостоятельная работа (всего)	**54**	**76**	**130**
В том числе:			
Проработка лекционного материала	16	12	28
Подготовка к лабораторным работам	26	16	42
Подготовка к практическим занятиям	–	16	16
Самостоятельное изучение тем теоретической части	12	4	16
Подготовка реферата	–	28	28
Подготовка к экзамену	**–**	**36**	**36**
Вид промежуточной аттестации (зачет, экзамен)	зачет	экзамен	
Общая трудоемкость час.	**108**	**180**	**288**
зач. ед.	**3**	**5**	**8**

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

5.1. Разделы дисциплин и виды занятий

№ п/п	Наименование раздела дисциплины	Лекц.	Практ зан.	Лаб. раб.	КРС	СРС	Всего час.	Формируемые компетенции
Семестр 4								
1	Понятие программной инженерии	2		4		6	12	ПК-20, ПК-8
2	Жизненный цикл программного обеспечения	2		4		8	14	ПК-20, ПК-8
3	Процесс разработки программного обеспечения	4		8		10	22	ПК-20, ПК-8
4	Проектирование ПО	2		4		8	14	ПК-20, ПК-8
5	Управление проектами	2		4		6	12	ПК-20, ПК-8
6	Управление требованиями	2		4		6	12	ПК-20, ПК-8
7	Конфигурационное управление	2		4		6	12	ПК-20, ПК-8
8	Верификация и аттестация ПО	2		4		4	10	ПК-20, ПК-8
Итого за 4-й семестр		**18**	**–**	**36**	**–**	**54**	**108**	
Семестр 5								
9	Управление документацией ПО	2	2	2		4	10	ПК-20, ПК-8
10	Методологии разработки ПО	4	2	2		4	12	ПК-20, ПК-8

Согласована на портале № 7359

11	Качество разработки ПО	2	2	2		4	10	ПК-20, ПК-8
12	Модернизация программного обеспечения	2	2	2		4	10	ПК-20, ПК-8
13	Оценка затрат на разработку и внедрение программных продуктов	2	4	4		14	24	ПК-20, ПК-8
14	Модели лицензирования программного обеспечения	2	2	2		6	12	ПК-20, ПК-8
15	Сертификация программного обеспечения	2	2	2		6	12	ПК-20, ПК-8
16	Управление и организация ИТ-услуг	2	2	2		6	12	ПК-20, ПК-8
17	Тематическая конференция				14	28	42	ПК-20
18	Экзамен					36	36	ПК-20
Итого за 5-ый семестр		**18**	**18**	**18**	**14**	**112**	**180**	
Итого по дисциплине		**36**	**18**	**54**	**14**	**166**	**288**	

5.2. Содержание разделов дисциплины (по лекциям)

№ п/п	Наименование разделов	Содержание разделов	Трудоемкость (час.)	Формируемые компетенции (ОК, ПК)
1	Понятие программной инженерии	Этапы развития программирования. Основные определения: программа, программное обеспечение, программный комплекс, программный продукт. Сложность как сущность крупных программных комплексов. Кризис программного обеспечения. Понятие программной инженерии. SWEBOK – свод необходимых знаний и рекомендуемых практик в области программной инженерии.	2	ПК-20, ПК-8
2	Жизненный цикл программного обеспечения	Стандартизация в области ПО. Классификация стандартов по уровням, по разработчикам. Стандарты ЕСПД, ЕСС АСУ, ИТ (СССР, РФ). Процессы жизненного цикла программного обеспечения по стандартам ISO/IEC 12207:1995 и ISO/IEC 12207:2008.	2	ПК-20, ПК-8
3	Процесс разработки программного обеспечения	Классические модели процесса: водопадная модель, V-модель, инкрементная модель, спиральная модель. Фазы процесса разработки программного обеспечения.	4	ПК-20, ПК-8
4	Проектирование ПО	Понятие архитектуры ПО. Стадии проектирования по ГОСТ 2.103-68: Техническое задание, Техническое предложение, Эскизный проект, Технический проект, Рабочий проект. Средства автоматизации проектирования.	4	ПК-20, ПК-8

Согласована на портале № 7359

5	Управление проектами	PMBOK – Свод знаний по управлению проектами. Процессы инициирования. Процессы планирования. Процессы исполнения. Процессы мониторинга и управления. Процессы завершения. Управление рисками. Методологии P2M, PRINCE2.	2	ПК-20, ПК-8
6	Управление требованиями	Виды требований: функциональные требования, нефункциональные требования. Свойства требований: ясность и недвусмысленность, полнота и непротиворечивость, необходимый уровень детализации, прослеживаемость, тестируемость и проверяемость, модифицируемость. Формализация требований. Цикл работы с требованиями: анализ, формирование, аттестация, управление.	2	ПК-20, ПК-8
7	Конфигурационное управление	Понятие конфигурационного управления. Управление версиями. Управление сборками. Средства версионного контроля. Единицы конфигурационного управления	2	ПК-20, ПК-8
8	Верификация и аттестация ПО	Инспектирование ПО. Метод «чистой комнаты». Тестирование и отладка ПО. Принципы, методы и этапы тестирования. Тестирование черного ящика. Тестирование белого ящика. Инструменты тестирования. Критерии тестирования.	2	ПК-20, ПК-8
9	Управление документацией ПО	Методы и приемы документирования. Стандарты ГОСТ Р ИСО/МЭК 9294-93 – ИТ. Руководство по управлению документированием программного обеспечения и ГОСТ Р ИСО/МЭК 15910-2002 – Процесс создания документации пользователя программного средства. Структура проектной документации. Структура технической документации. Структура пользовательской документации. Автоматизированные средства документирования.	2	ПК-20, ПК-8
10	Методологии разработки ПО	MSF: основные принципы. Модель команды. Ролевые кластеры. Масштабирование команды MSF. Модель процесса. Управление компромиссами. RUP: структура, потоки, артефакты и роли, лучшие практики. Гибкие (agile) методы разработки ПО: общее описание. Extreme Programming (XP): общее описание, основные принципы организации процесса. Scrum:	2	ПК-20, ПК-8

Согласована на портале № 7359

		общее описание, роли, практики		
11	Качество разработки ПО	Дерево характеристик качества по стандарту ISO/IEC 9126:1991. Показатели надежности программных средств. Оценка уровня дефектов программных изделий. Концепция «Шесть сигма». Стандарты серий ISO 9000 и ISO 10000. Уровни зрелости процессов по CMMI..	2	ПК-20, ПК-8
12	Эволюция программного обеспечения	Наследуемые системы. Динамика развития программ. Сопровождение ПО. Эволюция системной архитектуры	2	ПК-20, ПК-8
13	Оценка затрат на разработку и внедрение программных продуктов	Оценка экономической эффективности разработки ПП. Качественные и количественные характеристики программного обеспечения. Исследование затрат на разработку ПП. Технико-экономическое обоснование. Конструктивные модели стоимости Барри Боэма COCOMO и COCOMO II	4	ПК-20, ПК-8
14	Модели лицензирования программного обеспечения	Лицензионное соглашение (EULA). Лицензии проприетарного ПО. Критерии свободы Р.Столлмена. Лицензии открытого ПО. Характеристика отечественных и зарубежных ведущих фирм-производителей ПО	2	ПК-20, ПК-8
15	Сертификация программного обеспечения	Обоснование необходимости сертификации ПО. Этапы сертификации. Системы сертификации. TickIT: сертификация систем качества для программного обеспечения	2	ПК-20, ПК-8
16	Управление и организация ИТ-услуг	Методология управления и организации ИТ-услуг (ITIL/ITSM). Комплекс стандартов и руководств в области управления IT, аудита и IT-безопасности (COBIT). Стандарт управления и обслуживания IT-сервисов (ISO 20000)	2	ПК-20, ПК-8
		ИТОГО	36	

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

№ п/п	Наименование обеспечивающих (предыдущих) дисциплин	№№ разделов данной дисциплины, для которых необходимо изучение обеспечивающих (предыдущих) дисциплин														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1.	Информационные системы и технологии	+	+								+		+		+	+	
2.	Информатика и программирование	+		+										+			
3.	Объектно-				+						+			+			

Согласована на портале № 7359

	ориентированное программирование																
4.	Проектный менеджмент			+	+	+	+				+	+					
5.	Информационный менеджмент							+	+						+	+	+

№ п/п	Наименование последующих дисциплин	№ № разделов данной дисциплины, которые необходимы при изучении последующих дисциплин															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1.	Проектный практикум			+	+	+	+	+	+	+	+	+	+				+
2.	Проектирование экономических информационных систем		+	+	+	+	+	+	+	+	+	+	+	+			
3.	Подготовка ВКР	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

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

Перечень компетенций	Л	ПЗ	ЛР	СРС	Формы контроля *(примеры)*
ПК-8	–	+	+	+	Проверка выполнения лабораторных работ
ПК-20	+	+	+	+	Устный ответ на практических занятиях, проверка выполнения лабораторных работ, проверка рефератов

Л – лекция, ПЗ – практические занятия, ЛР – лабораторные работы; СРС – самостоятельная работа студента

6. МЕТОДЫ И ФОРМЫ ОРГАНИЗАЦИИ ОБУЧЕНИЯ

Для успешного освоения дисциплины применяются различные образовательные технологии, которые обеспечивают достижение планируемых результатов обучения согласно основной образовательной программе с учетом требований к объему занятий в интерактивной форме.

Технологии интерактивного обучения при разных формах занятий

Методы	Формы		
	Лекции (час.)	Лабораторные и практические занятия (час.)	Всего (час.)
1. Презентация с дискуссией	–	8	**8**
2. Конкурсы тематических исследований с обсуждением	–	10	**10**
Итого интерактивных занятий	**–**	**18**	**18**

Примечание.

1. Метод «Дискуссия» используется при представлении творческой работы (лабораторная работа 4).

2. Метод «Конкурсы тематических исследований» используется при обсуждении реферативных работ (п.10).

7. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

Согласована на портале № 7359

№ п/п	№ раздела дисциплины из табл. 5.1	Темы лабораторных работ	Трудо-емкость (час.)	ПК
Семестр 4			**36**	
1	3	Разработка программного приложения для моделирования развития финансовой пирамиды в среде MS Excel	12	ПК-8
2	4, 5	Разработка проекта создания БД в MS Excel с применением VBA	12	ПК-8
3	9	Применение справочной технологии QuickSheet для решения задач в среде MathCAD	4	ПК-8
4	12, 14	Инсталляция и изучение проблемно-ориентированных экономических программных продуктов	8	ПК-8
Семестр 5			**18**	
5	13	Проект автоматизированной системы технико-экономического обоснования разработки ПО	8	ПК-8
6	13	Калькулятор трудозатрат на разработку программного обеспечения на основе модели COCOMO	10	ПК-8

Методические указания по организации и выполнению лабораторных работ студентов приведены в [4,5] п. 12.3 настоящей рабочей программы.

8. ПРАКТИЧЕСКИЕ ЗАНЯТИЯ (СЕМИНАРЫ)

№ п/п	№ раздела дисциплины из табл. 5.1	Темы практических занятий	Трудо-емкость (час.)	ПК
Семестр 5			**18**	
1	5	Методологии P2M, PRINCE2 управления проектами	2	ПК-20
2	8	Практические приемы тестирования программ	2	ПК-20
3	11	Рабочая документация на программный продукт	2	ПК-20
4	10	Матрица «модель-формализм» методологий разработки ПО	2	ПК-20
5	11	Модель зрелости процессов разработки программного обеспечения SW-CMM	2	ПК-20
6	13	Технико-экономическое обоснование разработки ПО	2	ПК-20
7	14	Свободы и ограничения лицензионных соглашений на использование ПО	2	ПК-20
8	16	COBIT Cube-элементы управления ИТ	2	ПК-20
9	1	SWEBOK и профессиональные стандарты АПКИТ в области информационных технологий	2	ПК-20

Согласована на портале № 7359

Методические указания по организации и проведению практических занятий приведены в [1,3] п. 12.3 настоящей рабочей программы.

9. САМОСТОЯТЕЛЬНАЯ РАБОТА

№ п/п	№ раздела дисциплины из табл. 5.1	Тематика самостоятельной работы (детализация)	Контроль выполнения работы (Опрос, тест, дом. задание и т.д.)	Трудо-емкость (час.)	ПК
Семестр 4				**54**	
1	1-8	Проработка лекционного материала	Опрос на занятиях (устно)	16	ПК-20
2	3-5, 9	Подготовка к лабораторным работам	Отчет, защита лабор. работ	26	ПК-20, ПК-8
3	3, 4	Самостоятельное изучение тем теоретической части	Дом. задание, тест	12	ПК-20
Семестр 5				**112**	
4	9-16	Проработка лекционного материала	Опрос на занятиях (устно)	12	ПК-20
5	12, 13, 14	Подготовка к лабораторным работам	Отчет, защита лабор. работ	16	ПК-20, ПК-8
6	1, 5, 8, 10, 11, 13, 14, 16	Подготовка к практическим занятиям	Отчет, защита практ. работ	16	ПК-20, ПК-8
7	5	Самостоятельное изучение тем теоретической части	Дом. задание, тест	4	ПК-20
8	17	Подготовка рефератов	Тематическая конференция	28	ПК-20
9	18	Подготовка к экзамену	Оценка на экзамене	36	ПК-20

Примечание. Темы для самостоятельного изучения:

1. Паттерны проектирования информационных систем (8 час.).
2. Разработка критических систем (4 час.).
3. Управление персоналом в процессе разработки программного обеспечения (4 час.).

Методические указания по организации и выполнению самостоятельной работы студентов приведены в [3] п. 12.3 настоящей рабочей программы.

Контроль самостоятельной работы студентов (аудиторные занятия)

№ п/п	Темы аудиторных занятий	Трудо-емкость (час.)	ОК, ПК
Семестр 5		**14**	

Согласована на портале № 7359

1	Подготовка к тематической конференции. Цели и задачи рефератов по дисциплине «Программная инженерия». Требования к оформлению. Интернет-источники информации по тематике рефератов	4	ПК-20
2	Представление и обсуждение рефератов на тематической конференции	10	ПК-20

10. ПРИМЕРНАЯ ТЕМАТИКА РЕФЕРАТОВ

Цель выполнение реферативных работ по дисциплине «Программная инженерия» – получение обучающими представления о нормативных документах (стандартов «де-юре» и «де-факто», моделей, методологий, руководств), используемых в сфере разработки, сопровождения и эксплуатации программных средств.

Для этого необходимо провести самостоятельный поиск в сети Интернет нормативных документов, относящихся к выбранной теме, провести их обзор и сравнительный анализ, подготовить выступление на 5-7 мин. с докладом и презентацией, выполненной в MS PowerPoint.

Методические указания по проведению исследований в рамках выполнения реферативных работ студентов приведены в [2] п. 12.3 настоящей рабочей программы

Реферат оформляется в бумажном виде в соответствии с требованиями ОС ТУСУР 01-2013. Работы студенческие по направлениям подготовки и специальностям технического профиля. Общие требования и правила оформления. – Томск, ТУСУР, 2013. Стандарт находится на сайте ТУСУР по адресу:

http://tusur.ru/export/sites/ru.tusur.new/ru/education/documents/inside/tech_01-2013_new.pdf .

Примерная тематика рефератов

1. Модели лицензирования программного обеспечения.
2. Методы и стандарты документирования ПО.
3. Методы и стандарты качества разработки ПО.
4. Методы и стандарты проектирования ПО.
5. Методы и стандарты тестирования ПО.
6. Методы и стандарты сопровождения ПО.
7. Методы и стандарты управления конфигурацией ПО.
8. Методы и стандарты управления проектами по созданию ПО.
9. Методы и стандарты верификации и валидации ПО.
10. Методы и стандарты управления требованиями к ПО.
11. TickIT: сертификация систем качества для программного обеспечения.
12. Сравнительный анализ стандартов ISO/IEC 12207-95 и ISO/IEC 12207-2008.
13. Стандарт ISO/IEC 15504 (SPICE) оценки процессов разработки и поддержки ПО.
14. ISO 20000 – стандарт для управления и обслуживания IT-сервисов.
15. Сравнительный анализ моделей жизненного цикла ПО.
16. Методология RUP от IBM Rational Software.
17. Методология MSF от Microsoft.
18. Методология CDM от Oracle.
19. Agile-методология XP.
20. Agile-методология Scrum.
21. Методология RAD создания средств разработки программных продуктов.
22. Характеристика стандартов ГОСТ 19, ГОСТ 24, ГОСТ 34.
23. CASE-средства разработки информационных систем.
24. SEI CMMI: модель зрелости процесса разработки.
25. COBIT: комплекс стандартов и руководств в области управления IT, аудита и IT-безопасности.

Согласована на портале № 7359

26. ITIL/ITSM: методология управления и организации ИТ-услуг.
27. SWEBOK: свод знаний в области программной инженерии.

11. БАЛЛЬНО-РЕЙТИНГОВАЯ СИСТЕМА

11.1 Курс 2, семестр 4: контроль обучения – зачет.

Максимальный семестровый рейтинг – **100 баллов.**

По дисциплине «Программная инженерия» итоговой формой отчетности во 4 семестре является **зачет**, все 100 баллов входят в семестровую составляющую.

Для стимулирования планомерности работы студента в семестре в раскладку баллов по элементам контроля введен компонент своевременности, который применяется только для студентов, без опозданий отчитывающихся по предусмотренным элементам контроля.

На протяжении всего семестра текущая успеваемость оценивается в баллах нарастающим итогом, в том числе и результаты контрольных точек.

Текущий контроль изучения дисциплины состоит из следующих видов:

- контроль за усвоением теоретического материала – проведение двух тестов;
- контроль за своевременным выполнением четырех лабораторных работ.

В таблице содержится распределение баллов в течение 4 семестра для дисциплины «Программная инженерия», завершающейся **зачетом** и содержащей 9 лекций (18 часов), 18 лабораторных занятий (36 часов), проводимых в течение семестра, и 2 итоговых теста во время проведения двух контрольных точек.

Элементы учебной деятельности	Максимальный балл на 1-ую контрольную точку с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
Посещение занятий	4	4	4	**12**
Тестовый контроль	10	10	10	**30**
Выполнение лабораторных работ	15	20	10	**45**
Компонент своевременности	4	4	5	**13**
Итого максимум за период:	**33**	**38**	**29**	**100**
Нарастающим итогом	**33**	**71**	**100**	

Для получения зачета необходимо набрать 75 и более баллов.

11.2 Курс 3, семестр 5: контроль обучения – экзамен.

Максимальный семестровый рейтинг – **100 баллов.**

Итоговой формой отчетности в 5 семестре по дисциплине «Программная инженерия» является **экзамен**, все 100 баллов входят в семестровую составляющую.

На протяжении всего семестра текущая успеваемость оценивается в баллах нарастающим итогом, в том числе и результаты контрольных точек.

Текущий контроль изучения дисциплины состоит из следующих видов:

- контроль за усвоением теоретического материала – проведение двух тестов;
- контроль за своевременным выполнением двух лабораторных работ.
- контроль за выполнением реферата.

В таблице содержится пример распределения баллов в течение семестра для дисциплины «Программная инженерия», завершающейся экзаменом и содержащей 18 лекций (36 час.), две лабораторные работ (18 час.), девять практических занятий (18 час.), курсовая работа, экзамен

Согласована на портале № 7359

Элементы учебной деятельности	Максимальный балл на 1-ую контрольную точку с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
Посещение занятий	2	2	1	**5**
Тестовый контроль	4	4	0	**8**
Выполнение и защита результатов лабораторных работ	10	20	10	**40**
Выполнение и защита реферата			10	**10**
Компонент своевременности	4	4	4	**12**
Экзамен			**25**	**25**
Итого максимум за период	**20**	**30**	**50**	**100**
Нарастающим итогом	**18**	**50**	**100**	

По результатам текущего контроля формируется допуск студента к итоговому контролю – экзамену по дисциплине. Экзамен осуществляется в форме компьютерного тестирования по теоретической части дисциплины. Система тестирования случайным образом генерирует 21 вопрос из базы, содержащей 125 вопросов. 19 вопросов оценивается в 1 балл, два вопрос в 3 балла. Итого: 25 баллов.

В составе суммы баллов, полученной студентом по дисциплине, экзаменационная составляющая должна быть не менее 15 баллов. В противном случае экзамен считается не сданным, студент в установленном в ТУСУРе порядке обязан его пересдать.

Контрольные вопросы, которые служат для составления компьютерного тестирования, приведены в Приложении.

11.3 Методика формирования пятибалльных оценок в контрольные точки

В таблице представлен пересчет суммы баллов по первой и второй контрольной точке в традиционную оценку.

Баллы на дату контрольной точки	Оценка
Не менее 90% от максимальной суммы на дату КТ	5
От 70% до 89% от максимальной суммы на дату КТ	4
От 60% до 69% от максимальной суммы на дату КТ	3
Менее 60% от максимальной суммы на дату КТ	2

11.4 Методика формирования итоговой оценки по дисциплине

В таблице представлен пересчет итоговой суммы баллов в традиционную и международную оценку. Преобразование суммы баллов в традиционную оценку и в международную буквенную оценку происходит после успешной сдачи экзамена по дисциплине «Программная инженерия».

Согласована на портале № 7359

Оценка (ГОС)	Итоговая сумма баллов, учитывает успешно сданный экзамен	Оценка (ECTS)
5 (отлично)	**90 – 100**	A (отлично)
4 (хорошо)	**85 – 89**	B (очень хорошо)
	75 – 84	C (хорошо)
	70 – 74	D (удовлетворительно)
3 (удовлетворительно)	**65 – 69**	
	60 – 64	E (посредственно)
2 (неудовлетворительно)	**Ниже 60 баллов**	F (неудовлетворительно)

12. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

12.1 Основная литература

1. Ехлаков, Ю.П.. Введение в программную инженерию [Электронный ресурс] : учебное пособие / Ю. П. Ехлаков; Томский государственный университет систем управления и радиоэлектроники (Томск). – Электрон. текстовые дан. – Томск : [б. и.], 2011. – on-line, 148 с. – Режим доступа: http://edu.tusur.ru/training/publications/141

12.2 Дополнительная литература

1. Миньков С.Л. Разработка и применение ППП в экономике: Учебное пособие. – Томск: ТМЦДО, 2002. – 231 с. (8 экз.)
2. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. – М.: Финансы и статистика, 2002. – 349 с. (34 экз.)
3. Ван-Тассел Д. Стиль, разработка, эффективность, отладка и испытание программ. – М.: Мир, 1981. – 320 с. (3 экз.)

12.3 Перечень пособий, методических указаний и материалов, используемых в учебном процессе

1. Ехлаков, Ю.П. Введение в программную инженерию [Электронный ресурс]: методические указания по проведению практических занятий / Ю. П. Ехлаков; Томский государственный университет систем управления и радиоэлектроники (Томск). – Электрон. текстовые дан. – Томск : [б. и.], 2012. – on-line, 13 с. – Режим доступа: http://edu.tusur.ru/training/publications/667
2. Ехлаков, Ю.П. Введение в программную инженерию [Электронный ресурс]: методические указания к выполнению практических и самостоятельных работ / Ю. П. Ехлаков; Томский государственный университет систем управления и радиоэлектроники (Томск). – Электрон. текстовые дан. – Томск : [б. и.], 2012. – on-line, 14 с. – Режим доступа: http://edu.tusur.ru/training/publications/996
3. Миньков С.Л. Технико-экономическое обоснование выполнения проекта: методическое пособие / С.Л. Миньков. – Томск: ТУСУР, 2014. – 30 с. [Электронный ресурс]. – Режим доступа: http://asu.tusur.ru/learning/spec080801/d26/s080801_d26_work.pdf
4. Миньков С.Л. Программная инженерия. Лабораторный практикум. Часть 1. – Томск: ТУСУР, 2014. – 40 с. [Электронный ресурс]. – Режим доступа: http://asu.tusur.ru/learning/bak230700/d36/b230700_d36_labs1.pdf
5. Миньков С.Л. Программная инженерия. Лабораторный практикум. Часть 2. – Томск: ТУСУР, 2014. – 42 с. [Электронный ресурс]. – Режим доступа: http://asu.tusur.ru/learning/bak230700/d36/b230700_d36_labs2.pdf

12.4 Периодические издания

Согласована на портале № 7359

1. Менеджмент в России и за рубежом
2. Стандарты и качество
3. Программная инженерия
4. Управление проектами и программами

12.5 Рекомендуемые информационные сайты

1. http://www.ashmanov.com/pap – Игорь Ашманов. Управление проектами.
2. http://russian.joelonsoftware.com – Джоэл Спольски о программном обеспечении.
3. http://www.interface.ru/iservices/catalog.asp?catId=160 – Программные продукты. Статьи (INTERFACE.RU)
4. http://citforum.ru/SE/lipaev – Программная инженерия в жизненном цикле программных средств
5. http://swebok.sorlik.ru – Сергей Орлик. Основы программной инженерии (по SWEBOK)
6. http://www.gostedu.ru – Учебный портал стандартов
7. http://www.intuit.ru – Национальный открытый университет информационных технологий
8. http://window.edu.ru – Единое окно доступа к образовательным ресурсам

13. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для выполнения лабораторных работ по дисциплине используются персональные ЭВМ с процессорами Pentium 4 и выше, операционные системы MS Windows XP, Vista, 7, офисные пакеты MS Office 2003, 2007, OpenOffice.org; математический пакет MathCAD, СУБД MySQL, среда разработки Visual Studio Express. В ходе лабораторных работ осуществляется поиск информации в глобальной сети Интернет. Для текущего и итогового контроля используется компьютерное тестирование.

Лекции, практические занятия и защита курсовых работ проводятся в специализированной аудитории с компьютерным проектором для демонстрации слайд-презентаций.

Согласована на портале № 7359

ПРИЛОЖЕНИЕ

КОНТРОЛЬНЫЕ ВОПРОСЫ ПО ДИСЦИПЛИНЕ

1. Охарактеризуйте этапы развития программного обеспечения.
2. Для чего предназначено прикладное ПО?
3. Для чего предназначено системное ПО?
4. Для чего предназначены инструментальные CASE-средства?
5. Дайте определение программного продукта (ПП).
6. Дайте определение программного комплекса (ППП).
7. Для чего предназначены обрабатывающие модули ППП?
8. Для чего предназначены управляющие модули ППП?
9. Для чего предназначены обслуживающие модули ППП?
10. Что такое информационная база ППП?
11. Что такое оболочка ППП?
12. Опишите суть пакетного режима работы ППП.
13. Опишите суть диалогового режима работы ППП.
14. Что такое модель предметной области ППП?
15. На основе какого входного языка осуществляется управление современными ППП?
16. Каковы характеристики программного модуля как отдельной функциональной единицы ППП
17. Для чего предназначен справочный интерфейс ППП?
18. Для чего предназначен интерфейс управления ППП?
19. Для чего предназначен интерфейс ввода-вывода информации в ППП?
20. Для чего предназначен информационный интерфейс ППП?
21. Чем WIMP-интерфейс отличается от SILK-интерфейса.
22. Что такое программная инженерия (software engineering)?
23. За какими проектами по разработке ПО закрепилось название «смертельный марш» («death march» – определение Э. Йордона)?
24. Что представляет собой методология разработки ПО? Назовите известные методологии разработки ПО и их отличительные признаки.
25. Что такое жизненный цикл программного продукта?
26. Что изменилось в российской стандартизации в связи с принятием Федерального закона «О техническом регулировании»?
27. Назовите стандарты СССР в области разработки ПО.
28. Назовите стандарты РФ в области разработки ПО.
29. Определите стадии разработки программного изделия в соответствии с ГОСТ 19.102-77.
30. Что такое профиль стандартов?
31. Назовите международные организации по стандартизации, российские стандартизирующие организации, стандартизирующие организации США.
32. Что определяет стандарт ГОСТ Р ИСО/МЭК 12207:1999?
33. Что определяет стандарт ГОСТ Р ИСО/МЭК 12207:2010?
34. Какова структура стандарта ГОСТ Р ИСО/МЭК 12207:1999?
35. Опишите процессы жизненного цикла ПО в соответствии со стандартом ГОСТ Р ИСО/МЭК 12207:2010.
36. В чем отличие каскадной модели жизненного цикла ПО от спиральной модели?
37. Какими средствами обеспечивается концептуальная целостность программного изделия?
38. Что включает в себя структурный подход к проектированию ПО?
39. Что включает в себя объектно-ориентированный подход к проектированию ПО?
40. Охарактеризуйте участников проекта по созданию ПП, их приоритеты и цели (на примере методологии Microsoft Solutions Framework).

Согласована на портале № 7359

41. В чем заключается этап проектировании ПО? Какая документация создается в процессе выполнения этого этапа?
42. Что входит в спецификацию модуля ПП?
43. На какой модели жизненного цикла ПО основано экстремальное программирование?
44. В чем заключается процесс тестирования программ?
45. В чем заключается процесс отладки программы?
46. Что такое детерминированное тестирование?
47. Что такое статическое тестирование?
48. Что такое стохастическое тестирование?
49. Что такое комплексное тестирование ПП?
50. Что такое системное тестирование ПП ?
51. Что такое бета-тестирование ПП?
52. Что такое альфа-тестирование ПП?
53. Для кого предназначена технологическая документация на ПП? Ее структура?
54. Для кого предназначена эксплуатационная документация на ПП? Ее структура?
55. В чем заключается сопровождение ПП?
56. Как реализуется монолитный метод сборки программных модулей при тестировании ПП?
57. Как реализуется пошаговый метод сборки программных модулей при тестировании ПП?
58. Что такое открытые системы? Назовите свойства открытых систем.
59. Что определяют стандарты POSIX?
60. Опишите лицензии: а) OEM; б) trialware; в) demoware; г) shareware; д) adware; е) freeware; ж) FPP; и) GPL.
61. Что определяет лицензионное соглашение на приобретаемое ПО? Какие вы знаете виды лицензионных соглашений?
62. Назовите характеристики качества ПП в соответствии со стандартом ISO 9126:1991? Опишите «дерево качества» ПО.
63. Как формируется комплексный показатель качества ПП?
64. Что такое надежность ПП?
65. Что составляет функциональную надежность программных систем?
66. Что считается дефектом программного изделия?
67. Когда дефекты и аномалии при функционировании ПП называют сбоем?
68. Когда дефекты и аномалии при функционировании ПП называют отказом?
69. Сколько дефектов может содержаться в программном изделии объемом в тысячу KSLOC при уровне качества «6 сигма»?
70. Опишите шаги реализации концепции качества «6 сигма» (Six Sigma).
71. Охарактеризуйте уровни модели CMM. Чем CMMI отличается от CMM?
72. Укажите отличия методологий разработки ПО: «Fix&Code», «Agile», RUP, MSF.
73. Чему посвящена серия стандартов ISO 9000, разработанная Международной организацией по стандартизации? Чем отличаются стандарты ISO 9000, ISO 9001, ISO 9004?
74. Чему посвящена серия стандартов ISO 10000, разработанная Международной организацией по стандартизации?
75. Для чего используется единица измерения KAELOC? SLOC?
76. Для чего используется единица измерения FPs?
77. Какие объекты программного изделия считаются функциональными точками?
78. На чем основана конструктивная модель стоимости Б.Боэма, предназначенная для оценки стоимости проекта создания ПП?
79. Чем отличаются базовая, промежуточная и детальная модели КОМОСТ?
80. Сравните модель COCOMO и COCOMO II.
81. Что такое открытые системы? Назовите свойства открытых систем.

Согласована на портале № 7359

82. Что определяет лицензионное соглашение на приобретаемое ПО? Какие вы знаете виды лицензионных соглашений?
83. Какое ПО называют проприетарным?
84. Чем проприетарное ПО отличается от коммерческого ПО?
85. Чем открытое ПО отличается от свободного ПО?
86. Какое ПО называют «ПО общественного домена» (public domain)?
87. Что означает термин «копилефт»?
88. Какие права дает пользователю ПО лицензии GNU GPL?
89. Назовите «четыре критерия свободы ПО по Столлману».
90. Чем программный прототип отличается от проектного паттерна?
91. Что такое спецификация ПО?
92. Какие модели, как правило, образуют архитектуру программной системы?
93. Чем процедура верификации отличается от процедуры аттестации?
94. Классификация рисков при разработке ПО.
95. Что такое SWEBOK?
96. Что такое PMBOK?
97. На чем основывается оценка значимости проекта разработки ПО?
98. Какие программные системы называют критическими?
99. Объясните термин «реинжиниринг программных систем».
100. Что такое сертификация как процесс?
101. Для чего нужна сертификация программного обеспечения?
102. Как организована система сертификации в РФ? Какие формы сертификации существуют по отношению к ПО?
103. Может ли быть сертифицировано зарубежное программное средство?
104. Что такое TickIT?

Согласована на портале № 7359

Приложение к рабочей программе

Федеральное государственное бюджетное образовательное учреждение высшего образования

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

УТВЕРЖДАЮ

Проректор по учебной работе

_____ **П. Е. Троян**

«4» 07 2016 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ «ПРОГРАММНАЯ ИНЖЕНЕРИЯ»

Уровень основной образовательной программы бакалавриат

Направление подготовки 09.03.03 – Прикладная информатика

Профиль(и) Прикладная информатика в экономике

Форма обучения очная

Факультет систем управления

Кафедра автоматизированных систем управления

Курс 2,3

Семестр 4,5

Учебный план набора 2013, 2014, 2015 гг.

Зачет 4 семестр, экзамен 5 семестр

Томск 2016

Согласована на портале № 7359

1. ВВЕДЕНИЕ

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины «Программная инженерия» и представляет собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.) и методов их использования, предназначенных для измерения уровня достижения студентом установленных результатов обучения.

ФОС по дисциплине используется при проведении текущего контроля успеваемости и промежуточной аттестации студентов.

Перечень закрепленных за дисциплиной «Программная инженерия» компетенций приведен в таблице 1.

Таблица 1 – Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенции
ПК-8	─ способность программировать приложения и создавать программные прототипы решения прикладных задач	**Знать:** – принципы организации проектирования и содержание этапов разработки программных комплексов. **Уметь:** – формировать архитектуру программных комплексов для информатизации предприятий, разрабатывать программные приложения. **Владеть:** – навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.
ПК-20	─ способность осуществлять и обосновывать выбор проектных решений по видам обеспечения информационных систем	**Знать:** – задачи и методы исследования и обеспечения качества и надежности программных компонентов **Уметь:** – формулировать требования к создаваемым программным комплексам; – использовать международные и отечественные стандарты жизненного цикла программного обеспечения

2. РЕАЛИЗАЦИЯ КОМПЕТЕНЦИЙ

2.1 Компетенция ПК-8

ПК-8 - способность программировать приложения и создавать программные прототипы решения прикладных задач конструирования и отладке программных средств

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания, представлены в таблице 2.

Таблица 2 – Этапы формирования компетенции и используемые средства оценивания

Согласована на портале № 7359

Состав	Знать	Уметь	Владеть
Содержание этапов	– принципы организации проектирования и содержание этапов разработки программных комплексов.	– формировать архитектуру программных комплексов для информатизации предприятий, разрабатывать программные приложения.	– навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов
Виды занятий	– Лекции; – Лабораторные работы; – Самостоятельная работа студентов – Тематическая конференция	– Лабораторные работы; – Самостоятельная работа студентов – Зачет	– Лабораторные работы; – Самостоятельная работа студентов
Используемые средства оценивания	– Тестирование; – Выступление с докладом на тематической конференции; – Экзамен	– Защита лабораторных работ	– Защита лабораторных работ

Общие характеристики показателей и критериев оценивания компетенции на всех этапах приведены в таблице 3.

Таблица 3 – Общие характеристики показателей и критериев оценивания компетенции по этапам

Показатели и критерии	Знать	Уметь	Владеть
ОТЛИЧНО (высокий уровень)	Обладает фактическими и теоретическими знаниями в пределах изучаемой области с пониманием границ применимости	Обладает диапазоном практических умений, требуемых для развития творческих решений, абстрагирования проблем	Самостоятельно планирует и выполняет работу, используя современный информационный инструментарий
ХОРОШО (базовый уровень)	Знает факты, принципы, процессы, общие понятия в пределах изучаемой области	Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования	Берет ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем
УДОВЛЕТВОРИТЕЛЬНО (низкий уровень)	Обладает низким уровнем общих	Обладает умениями на низком уровне,	Работает только при прямом наблюде-

Согласована на портале № 7359

	знаний	которые не достаточны для выполнения даже простых задач	нии

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 4.

Таблица 4 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	Знать	Уметь	Владеть
ОТЛИЧНО (высокий уровень)	– Обладает отличными знаниями принципов организации проектирования и содержание этапов разработки программных комплексов. – Знает методику технико-экономического обоснования разработки программного продукта	– Умеет формировать архитектуру программных комплексов для информатизации предприятий; – Умеет разрабатывать программные приложения.	– Владеет навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ; – Владеет навыками использования современных технологий программирования, тестирования и документирования программных комплексов.
ХОРОШО (базовый уровень)	– Знает принципы организации проектирования и содержание этапов разработки программных комплексов; – Знает показатели надежности программных средств.	– Умеет разрабатывать программные приложения.	– Владеет навыками использования современных технологий программирования, тестирования и документирования программных комплексов
УДОВЛЕТВОРИТЕЛЬНО (низкий уровень)	– Имеет общее представление о понятиях архитектуры ПО и стадиях проектирования по ГОСТу.	– Обладает низким уровнем знаний о формировании архитектуры программных комплексов для информатизации предприятий	– Слабо ориентируется во владении методами и приемами документирования и тестирования ПО..

2.2 Компетенция ПК-20

Согласована на портале № 7359

ПК-20 - способность осуществлять и обосновывать выбор проектных решений по видам обеспечения информационных систем.

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания, представлены в таблице 5.

Таблица 5 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь
Содержание этапов	– задачи и методы исследования и обеспечения качества и надежности программных компонентов	– формулировать требования к создаваемым программным комплексам; – использовать международные и отечественные стандарты жизненного цикла программного обеспечения
Виды занятий	– Лекции; – Лабораторные работы – Самостоятельная работа студентов	– Лекции; – Лабораторные работы; – Подготовка реферата; – Самостоятельная работа студентов
Используемые средства оценивания	– Тестирование; – Зачет; – Экзамен	– Тестирование; – Выступление с докладом на тематической конференции; – Экзамен

Общие характеристики показателей и критериев оценивания компетенции на всех этапах приведены в таблице 6.

Таблица 6 – Общие характеристики показателей и критериев оценивания компетенции по этапам

Показатели и критерии	Знать	Уметь
ОТЛИЧНО (высокий уровень)	Обладает фактическими и теоретическими знаниями в пределах изучаемой области с пониманием границ применимости	Обладает диапазоном практических умений, требуемых для развития творческих решений, абстрагирования проблем
ХОРОШО (базовый уровень)	Знает факты, принципы, процессы, общие понятия в пределах изучаемой области	Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования
УДОВЛЕТВОРИТЕЛЬНО (низкий уровень)	Обладает низким уровнем общих знаний	Обладает умениями на низком уровне, которые не достаточны для выполнения даже простых задач

Согласована на портале № 7359

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 7.

Таблица 7 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	Знать	Уметь
ОТЛИЧНО (высокий уровень)	– Отлично знает задачи и методы исследования и обеспечения качества и надежности программных компонентов; – Обладает знаниями по оценке уровня дефектов программных изделий.	– Умеет формулировать требования к создаваемым программным комплексам; – Умеет использовать международные и отечественные стандарты жизненного цикла программного обеспечения
ХОРОШО (базовый уровень)	– Знает показатели надежности программных средств, а также задачи и методы исследования и обеспечения их качества; .	– Умеет использовать международные и отечественные стандарты жизненного цикла программного обеспечения
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	– Обладает низким уровнем знаний о задачах и методах исследования и обеспечения качества и надежности программных компонентов.	– Обладает низким уровнем знаний в области формулирования требований к создаваемым программным комплексам.

3. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ

Для реализации вышеперечисленных задач обучения используются следующие материалы: типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы, в составе, приведенном ниже.

3.1 Темы лабораторных занятий

1. Разработка программного приложения для моделирования развития финансовой пирамиды в среде MS Excel
2. Разработка проекта создания БД в MS Excel с применением VBA
3. Применение справочной технологии QuickSheet для решения задач в среде MathCAD
4. Инсталляция и изучение проблемно-ориентированных экономических программных продуктов
5. Проект автоматизированной системы технико-экономического обоснования разработки программного продукта
6. Разработка калькулятора трудозатрат на разработку программного обеспечения на основе модели COCOMO

3.2 Примерные темы рефератов

Согласована на портале № 7359

1. Модели лицензирования программного обеспечения.
2. Методы и стандарты документирования ПО.
3. Методы и стандарты качества разработки ПО.
4. Методы и стандарты проектирования ПО.
5. Методы и стандарты тестирования ПО.
6. Методы и стандарты сопровождения ПО.
7. Методы и стандарты управления конфигурацией ПО.
8. Методы и стандарты управления проектами по созданию ПО.
9. Методы и стандарты верификации и валидации ПО.
10. Методы и стандарты управления требованиями к ПО.
11. TickIT: сертификация систем качества для программного обеспечения.
12. Сравнительный анализ стандартов ISO/IEC 12207-95 и ISO/IEC 12207-2008.
13. Стандарт ISO/IEC 15504 (SPICE) оценки процессов разработки и поддержки ПО.
14. ISO 20000 – стандарт для управления и обслуживания IT-сервисов.
15. Сравнительный анализ моделей жизненного цикла ПО.
16. Методология RUP от IBM Rational Software.
17. Методология MSF от Microsoft.
18. Методология CDM от Oracle.
19. Agile-методология XP.
20. Agile-методология Scrum.
21. Методология RAD создания средств разработки программных продуктов.
22. Характеристика стандартов ГОСТ 19, ГОСТ 24, ГОСТ 34.
23. CASE-средства разработки информационных систем.
24. SEI CMMI: модель зрелости процесса разработки.
25. COBIT: комплекс стандартов и руководств в области управления IT, аудита и IT-безопасности.
26. ITIL/ITSM: методология управления и организации ИТ-услуг.
27. SWEBOK: свод знаний в области программной инженерии.

3.3 Контрольные вопросы для подготовки к экзамену

1. Охарактеризуйте этапы развития программного обеспечения.
2. Для чего предназначено прикладное ПО?
3. Для чего предназначено системное ПО?
4. Для чего предназначены инструментальные CASE-средства?
5. Дайте определение программного продукта (ПП).
6. Дайте определение программного комплекса (ППП).
7. Для чего предназначены обрабатывающие модули ППП?
8. Для чего предназначены управляющие модули ППП?
9. Для чего предназначены обслуживающие модули ППП?
10. Что такое информационная база ППП?
11. Что такое оболочка ППП?
12. Опишите суть пакетного режима работы ППП.
13. Опишите суть диалогового режима работы ППП.
14. Что такое модель предметной области ППП?
15. На основе какого входного языка осуществляется управление современными ППП?
16. Каковы характеристики программного модуля как отдельной функциональной единицы ППП
17. Для чего предназначен справочный интерфейс ППП?
18. Для чего предназначен интерфейс управления ППП?
19. Для чего предназначен интерфейс ввода-вывода информации в ППП?

Согласована на портале № 7359

20. Для чего предназначен информационный интерфейс ППП?
21. Чем WIMP-интерфейс отличается от SILK-интерфейса.
22. Что такое программная инженерия (software engineering)?
23. За какими проектами по разработке ПО закрепилось название «смертельный марш» («death march» – определение Э. Йордона)?
24. Что представляет собой методология разработки ПО? Назовите известные методологии разработки ПО и их отличительные признаки.
25. Что такое жизненный цикл программного продукта?
26. Что изменилось в российской стандартизации в связи с принятием Федерального закона «О техническом регулировании»?
27. Назовите стандарты СССР в области разработки ПО.
28. Назовите стандарты РФ в области разработки ПО.
29. Определите стадии разработки программного изделия в соответствии с ГОСТ 19.102-77.
30. Что такое профиль стандартов?
31. Назовите международные организации по стандартизации, российские стандартизирующие организации, стандартизирующие организации США.
32. Что определяет стандарт ГОСТ Р ИСО/МЭК 12207:1999?
33. Что определяет стандарт ГОСТ Р ИСО/МЭК 12207:2010?
34. Какова структура стандарта ГОСТ Р ИСО/МЭК 12207:1999?
35. Опишите процессы жизненного цикла ПО в соответствии со стандартом ГОСТ Р ИСО/МЭК 12207:2010.
36. В чем отличие каскадной модели жизненного цикла ПО от спиральной модели?
37. Какими средствами обеспечивается концептуальная целостность программного изделия?
38. Что включает в себя структурный подход к проектированию ПО?
39. Что включает в себя объектно-ориентированный подход к проектированию ПО?
40. Охарактеризуйте участников проекта по созданию ПП, их приоритеты и цели (на примере методологии Microsoft Solutions Framework).
41. В чем заключается этап проектировании ПО? Какая документация создается в процессе выполнения этого этапа?
42. Что входит в спецификацию модуля ПП?
43. На какой модели жизненного цикла ПО основано экстремальное программирование?
44. В чем заключается процесс тестирования программ?
45. В чем заключается процесс отладки программы?
46. Что такое детерминированное тестирование?
47. Что такое статическое тестирование?
48. Что такое стохастическое тестирование?
49. Что такое комплексное тестирование ПП?
50. Что такое системное тестирование ПП ?
51. Что такое бета-тестирование ПП?
52. Что такое альфа-тестирование ПП?
53. Для кого предназначена технологическая документация на ПП? Ее структура?
54. Для кого предназначена эксплуатационная документация на ПП? Ее структура?
55. В чем заключается сопровождение ПП?
56. Как реализуется монолитный метод сборки программных модулей при тестировании ПП?
57. Как реализуется пошаговый метод сборки программных модулей при тестировании ПП?
58. Что такое открытые системы? Назовите свойства открытых систем.
59. Что определяют стандарты POSIX?

Согласована на портале № 7359

60. Опишите лицензии: а) OEM; б) trialware; в) demoware; г) shareware; д) adware; е) freeware; ж) FPP; и) GPL.

61. Что определяет лицензионное соглашение на приобретаемое ПО? Какие вы знаете виды лицензионных соглашений?

62. Назовите характеристики качества ПП в соответствии со стандартом ISO 9126:1991? Опишите «дерево качества» ПО.

63. Как формируется комплексный показатель качества ПП?

64. Что такое надежность ПП?

65. Что составляет функциональную надежность программных систем?

66. Что считается дефектом программного изделия?

67. Когда дефекты и аномалии при функционировании ПП называют сбоем?

68. Когда дефекты и аномалии при функционировании ПП называют отказом?

69. Сколько дефектов может содержаться в программном изделии объемом в тысячу KSLOC при уровне качества «6 сигма»?

70. Опишите шаги реализации концепции качества «6 сигма» (Six Sigma).

71. Охарактеризуйте уровни модели CMM. Чем CMMI отличается от CMM?

72. Укажите отличия методологий разработки ПО: «Fix&Code», «Agile», RUP, MSF.

73. Чему посвящена серия стандартов ISO 9000, разработанная Международной организацией по стандартизации? Чем отличаются стандарты ISO 9000, ISO 9001, ISO 9004?

74. Чему посвящена серия стандартов ISO 10000, разработанная Международной организацией по стандартизации?

75. Для чего используется единица измерения KAELOC? SLOC?

76. Для чего используется единица измерения FPs?

77. Какие объекты программного изделия считаются функциональными точками?

78. На чем основана конструктивная модель стоимости Б.Боэма, предназначенная для оценки стоимости проекта создания ПП?

79. Чем отличаются базовая, промежуточная и детальная модели КОМОСТ?

80. Сравните модель COCOMO и COCOMO II.

81. Что такое открытые системы? Назовите свойства открытых систем.

82. Что определяет лицензионное соглашение на приобретаемое ПО? Какие вы знаете виды лицензионных соглашений?

83. Какое ПО называют проприетарным?

84. Чем проприетарное ПО отличается от коммерческого ПО?

85. Чем открытое ПО отличается от свободного ПО?

86. Какое ПО называют «ПО общественного домена» (public domain)?

87. Что означает термин «копилефт»?

88. Какие права дает пользователю ПО лицензии GNU GPL?

89. Назовите «четыре критерия свободы ПО по Столлману».

90. Чем программный прототип отличается от проектного паттерна?

91. Что такое спецификация ПО?

92. Какие модели, как правило, образуют архитектуру программной системы?

93. Чем процедура верификации отличается от процедуры аттестации?

94. Классификация рисков при разработке ПО.

95. Что такое SWEBOK?

96. Что такое PMBOK?

97. На чем основывается оценка значимости проекта разработки ПО?

98. Какие программные системы называют критическими?

99. Объясните термин «реинжиниринг программных систем».

100. Что такое сертификация как процесс?

101. Для чего нужна сертификация программного обеспечения?

102. Как организована система сертификации в РФ? Какие формы сертификации существуют по отношению к ПО?

Согласована на портале № 7359

103. Может ли быть сертифицировано зарубежное программное средство?
104. Что такое TickIT?

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Для обеспечения процесса обучения и решения задач обучения используются следующие материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций (приведены в разделах 12.1 – 12.3 рабочей программы):

1. Ехлаков, Ю.П.. Введение в программную инженерию [Электронный ресурс] : учебное пособие / Ю. П. Ехлаков; Томский государственный университет систем управления и радиоэлектроники (Томск). – Электрон. текстовые дан. – Томск : [б. и.], 2011. – on-line, 148 с. – Режим доступа: http://edu.tusur.ru/training/publications/141
2. Миньков С.Л. Разработка и применение ППП в экономике: Учебное пособие. – Томск: ТМЦДО, 2002. – 231 с. (8 экз.)
3. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. – М.: Финансы и статистика, 2002. – 349 с. (34 экз.)
4. Ван-Тассел Д. Стиль, разработка, эффективность, отладка и испытание программ. – М.: Мир, 1981. – 320 с. (3 экз.)

5. Ехлаков, Ю.П. Введение в программную инженерию [Электронный ресурс]: методические указания по проведению практических занятий / Ю. П. Ехлаков; Томский государственный университет систем управления и радиоэлектроники (Томск). – Электрон. текстовые дан. – Томск : [б. и.], 2012. – on-line, 13 с. – Режим доступа: http://edu.tusur.ru/training/publications/667
6. Ехлаков, Ю.П. Введение в программную инженерию [Электронный ресурс]: методические указания к выполнению практических и самостоятельных работ / Ю. П. Ехлаков; Томский государственный университет систем управления и радиоэлектроники (Томск). – Электрон. текстовые дан. – Томск : [б. и.], 2012. – on-line, 14 с. – Режим доступа: http://edu.tusur.ru/training/publications/996
7. Миньков С.Л. Технико-экономическое обоснование выполнения проекта: методическое пособие / С.Л. Миньков. – Томск: ТУСУР, 2014. – 30 с. [Электронный ресурс]. – Режим доступа: http://asu.tusur.ru/learning/spec080801/d26/s080801_d26_work.pdf
8. Миньков С.Л. Программная инженерия. Лабораторный практикум. Часть 1. – Томск: ТУСУР, 2014. – 40 с. [Электронный ресурс]. – Режим доступа: http://asu.tusur.ru/learning/bak230700/d36/b230700_d36_labs1.pdf
9. Миньков С.Л. Программная инженерия. Лабораторный практикум. Часть 2. – Томск: ТУСУР, 2014. – 42 с. [Электронный ресурс]. – Режим доступа: http://asu.tusur.ru/learning/bak230700/d36/b230700_d36_labs2.pdf

Согласована на портале № 7359