МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

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

УТВЕРЖДАЮ
Проректор по учебной работе
_____ П. Е. Троян
«___» _____ 20__ г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Теория и технология программирования

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки (специальность): **27.03.03 Системный анализ и управление**
Направленность (профиль): **Системный анализ и управление в информационных технологиях**
Форма обучения: **очная**
Факультет: **ФВС, Факультет вычислительных систем**
Кафедра: **МиСА, Кафедра моделирования и системного анализа**
Курс: **3**
Семестр: **6**
Учебный план набора 2015 года

Распределение рабочего времени

№	Виды учебной деятельности	6 семестр	Всего	Единицы
1	Лекции	34	34	часов
2	Лабораторные занятия	34	34	часов
3	Всего аудиторных занятий	68	68	часов
4	Самостоятельная работа	76	76	часов
5	Всего (без экзамена)	144	144	часов
6	Подготовка и сдача экзамена	36	36	часов
7	Общая трудоемкость	180	180	часов
		5.0	5.0	З.Е

Экзамен: 6 семестр

Томск 2017

Согласована на портале № 9637

Рассмотрена и одобрена на заседании кафедры
протокол № 32 от «27» 12 2016 г.

ЛИСТ СОГЛАСОВАНИЙ

Рабочая программа составлена с учетом требований Федерального Государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) 27.03.03 Системный анализ и управление, утвержденного 2015-03-11 года, рассмотрена и утверждена на заседании кафедры «___» _____ 20__ года, протокол №_____.

Разработчики:

старший преподаватель каф. МиСА _____ Панов С. А.

Заведующий обеспечивающей каф. МиСА _____ Дмитриев В. М.

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами направления подготовки (специальности).

Декан ФВС _____ Козлова Л. А.

Заведующий выпускающей каф. МиСА _____ Дмитриев В. М.

Эксперты:

доцент каф. МиСА _____ Ганджа Т. В.

Согласована на портале № 9637

1. Цели и задачи дисциплины

1.1. Цели дисциплины

Целью данного курса является изучение основных принципов программирования, разработки программного обеспечения (ПО), введения в теорию жизненного цикла программного обеспечения, изучения теории автоматов и программирование на языках высокого и сверхвысокого уровня.

1.2. Задачи дисциплины

- построение этапов разработки ПО;
- методы конструирования ПО;
- методология разработки ПО;
- изучение программных сред для разработки;
- тестирование и отладка программного кода.

2. Место дисциплины в структуре ОПОП

Дисциплина «Теория и технология программирования» (Б1.Б.18) относится к блоку 1 (базовая часть).

Предшествующими дисциплинами, формирующими начальные знания, являются следующие дисциплины: Вычислительная математика, Вычислительные машины, системы и сети, Информатика, Объектно-ориентированное программирование, Пакеты прикладных программ MathCad, Пакеты прикладных программ MathLab, Программирование и основы алгоритмизации.

Последующими дисциплинами являются: Надежность информационных систем.

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

- ОПК-1 готовностью применять методы математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также методов гуманитарных, экономических и социальных наук;
- ОПК-7 способностью к освоению новой техники, новых методов и новых технологий;
- ПК-6 способностью создавать программные комплексы для системного анализа и синтеза сложных систем;

В результате изучения дисциплины студент должен:

- **знать** основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; основные принципы отладки и тестирования программных продуктов; методы и средства разработки технической документации.
- **уметь** осуществлять разработку кода программного модуля на современных языках программирования; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; оформлять документацию на программные средства; использовать инструментальные средства для автоматизации оформления документации.
- **владеть** навыками разработки как сложных компьютерных программ, так и отдельных их модулей.

4. Объем дисциплины и виды учебной работы

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

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры
		6 семестр
Аудиторные занятия (всего)	68	68
Лекции	34	34
Лабораторные занятия	34	34

Согласована на портале № 9637

Самостоятельная работа (всего)	76	76
Оформление отчетов по лабораторным работам	36	36
Проработка лекционного материала	20	20
Самостоятельное изучение тем (вопросов) теоретической части курса	20	20
Всего (без экзамена)	144	144
Подготовка и сдача экзамена	36	36
Общая трудоемкость час	180	180
Зачетные Единицы Трудоемкости	5.0	5.0

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Разделы дисциплины и виды занятий приведены в таблице 5.1.

Таблица 5.1 – Разделы дисциплины и виды занятий

№	Названия разделов дисциплины	Лекции	Лабораторные работы	Самостоятельная работа	Всего часов (без экзамена)	Формируемые компетенции
1	Основные этапы решения задач на ЭВМ	4	4	14	22	ОПК-1, ОПК-7, ПК-6
2	Методология проектирования ПО	8	8	16	32	ОПК-1, ОПК-7, ПК-6
3	Испытание, сопровождение, документирование ПО	6	6	14	26	ОПК-1, ОПК-7, ПК-6
4	Разработка ПО на языках сверхвысокого уровня	10	10	18	38	ОПК-1, ОПК-7, ПК-6
5	Теория вычислительных процессов	6	6	14	26	ОПК-1, ОПК-7, ПК-6
	Итого	34	34	76	144	

5.2. Содержание разделов дисциплины (по лекциям)

Содержание разделов дисциплин (по лекциям) приведено в таблице 5.2.

Таблица 5.2 - Содержание разделов дисциплин (по лекциям)

Названия разделов	Содержание разделов дисциплины по лекциям	Трудоемкость, ч	Формируемые компетенции
6 семестр			
1 Основные этапы решения задач на ЭВМ	Принципы разработки ТЗ. Выбор программных средств для разработки ПО. Оценка качества программного	4	ОПК-1, ОПК-7, ПК-6

Согласована на портале № 9637

	продукта.		
	Итого	4	
2 Методология проектирования ПО	Проектирование "сверху вниз". Проектирование структуры данных. Формы представления проекта. Структурное и модульное программирование. Стиль программирования. Объектно-ориентированное проектирование (ООП). CASE-технология: индустриальная разработка систем обработки информации. Разработка интерфейса пользователя. Принципы проектирования пользовательского интерфейса. Программирование оконной системы с использованием объектных библиотек.	8	ОПК-1, ОПК-7, ПК-6
	Итого	8	
3 Испытание, сопровождение, документирование ПО	Источники ошибок в ПО. Отладка, тестирование программ. Сопровождение ПО. Документирование программного продукта.	6	ОПК-1, ОПК-7, ПК-6
	Итого	6	
4 Разработка ПО на языках сверхвысокого уровня	Введение в языки сверхвысокого уровня. Технология разработки на языках LISP и PROLOG. Основной алфавит функциональных языков. Рекурсивные и лямбда исчисления. Задачи искусственного интеллекта. Построение экспертных систем. Объектно-ориентированный подход в языке LISP.	10	ОПК-1, ОПК-7, ПК-6
	Итого	10	
5 Теория вычислительных процессов	Задачи трансляторов. Компиляторы, трансляторы, препроцессоры. Теория Хомского. Языки и грамматики. Распознающие автоматы. Теория контекстно-свободных языков.	6	ОПК-1, ОПК-7, ПК-6
	Итого	6	
Итого за семестр		34	

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

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

Таблица 5.3 - Разделы дисциплины и междисциплинарные связи

№	Наименование дисциплин	№ разделов данной дисциплины, для которых необходимо изучение обеспечивающих и обеспечиваемых дисциплин

Согласована на портале № 9637

		1	2	3	4	5
Предшествующие дисциплины						
1	Вычислительная математика	+	+	+	+	+
2	Вычислительные машины, системы и сети	+				
3	Информатика	+	+	+	+	+
4	Объектно-ориентированное программирование		+	+	+	+
5	Пакеты прикладных программ MathCad					+
6	Пакеты прикладных программ MathLab					+
7	Программирование и основы алгоритмизации	+	+	+	+	+
Последующие дисциплины						
1	Надежность информационных систем	+		+		

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

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

Таблица 5. 4 – Соответствие компетенций и видов занятий, формируемых при изучении дисциплины

Компетенции	Виды занятий			Формы контроля
	Лекции	Лабораторные занятия	Самостоятельная работа	
ОПК-1	+	+	+	Экзамен, Конспект самоподготовки, Защита отчета, Отчет по лабораторной работе, Опрос на занятиях
ОПК-7	+	+	+	Экзамен, Конспект самоподготовки, Защита отчета, Отчет по лабораторной работе, Опрос на занятиях
ПК-6	+	+	+	Экзамен, Конспект самоподготовки, Защита отчета, Отчет по лабораторной работе, Опрос на занятиях

6. Интерактивные методы и формы организации обучения

Не предусмотрено РУП

7. Лабораторный практикум

Содержание лабораторных работ приведено в таблице 7.1.

Согласована на портале № 9637

Таблица 7. 1 – Содержание лабораторных работ

Названия разделов	Содержание лабораторных работ	Трудоемкость, ч	Формируемые компетенции
6 семестр			
1 Основные этапы решения задач на ЭВМ	Методика составления ТЗ	4	ОПК-1, ОПК-7, ПК-6
	Итого	4	
2 Методология проектирования ПО	Изучение методик проектирования	8	ОПК-1, ОПК-7, ПК-6
	Итого	8	
3 Испытание, сопровождение, документирование ПО	Инструменты и методы тестирования ПО	6	ОПК-1, ОПК-7, ПК-6
	Итого	6	
4 Разработка ПО на языках сверхвысокого уровня	Разработка на функциональных языках программирования.	10	ОПК-1, ОПК-7, ПК-6
	Итого	10	
5 Теория вычислительных процессов	Изучение работы компиляторов, трансляторов	6	ОПК-1, ОПК-7, ПК-6
	Итого	6	
Итого за семестр		34	

8. Практические занятия

Не предусмотрено РУП

9. Самостоятельная работа

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

Таблица 9.1 - Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов	Виды самостоятельной работы	Трудоемкость ч	Формируемые компетенции	Формы контроля
6 семестр				
1 Основные этапы решения задач на ЭВМ	Самостоятельное изучение тем (вопросов) теоретической части курса	4	ОПК-1, ОПК-7, ПК-6	Защита отчета, Конспект самоподготовки, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Проработка лекционного материала	4		
	Оформление отчетов по лабораторным работам	6		
	Итого	14		
2 Методология	Самостоятельное	4	ОПК-1,	Защита отчета, Конспект

Согласована на портале № 9637

проектирования ПО	изучение тем (вопросов) теоретической части курса		ОПК-7, ПК-6	самоподготовки, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Проработка лекционного материала	4		
	Оформление отчетов по лабораторным работам	8		
	Итого	16		
3 Испытание, сопровождение, документирование ПО	Самостоятельное изучение тем (вопросов) теоретической части курса	4	ОПК-1, ОПК-7, ПК-6	Защита отчета, Конспект самоподготовки, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Проработка лекционного материала	4		
	Оформление отчетов по лабораторным работам	6		
	Итого	14		
4 Разработка ПО на языках сверхвысокого уровня	Самостоятельное изучение тем (вопросов) теоретической части курса	4	ОПК-1, ОПК-7, ПК-6	Защита отчета, Конспект самоподготовки, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Проработка лекционного материала	4		
	Оформление отчетов по лабораторным работам	10		
	Итого	18		
5 Теория вычислительных процессов	Самостоятельное изучение тем (вопросов) теоретической части курса	4	ОПК-1, ОПК-7, ПК-6	Защита отчета, Конспект самоподготовки, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Проработка лекционного материала	4		
	Оформление отчетов по лабораторным работам	6		
	Итого	14		
Итого за семестр		76		
	Подготовка к экзамену	36		Экзамен
Итого		112		

9.1. Темы для самостоятельного изучения теоретической части курса

1. Основные этапы решения задач на ЭВМ.
2. Методология проектирования ПО.
3. Испытание, сопровождение, документирование ПО.
4. Разработка ПО на языках сверхвысокого уровня.
5. Теория вычислительных процессов.
6. Компьютерная графика.
7. Межпроцессное взаимодействие в ОС Windows.

Согласована на портале № 9637

8. Метрология и качество программного обеспечения.
9. Объектно-ориентированное программирование.
10. Программирование (процедурное).
11. Разработка приложений в .Net Framework (C#).
12. Разработка распределенных приложений (Java).
13. Технология разработки программного обеспечения.
14. Управление данными с помощью ADO.Net, ASP.Net.

10. Курсовая работа

Не предусмотрено РУП

11. Рейтинговая система для оценки успеваемости студентов

11.1. Балльные оценки для элементов контроля

Таблица 11.1 – Бальные оценки для элементов контроля

Элементы учебной деятельности	Максимальный балл на 1-ую КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
6 семестр				
Защита отчета	4	8	8	20
Конспект самоподготовки	4	5	6	15
Опрос на занятиях	4	8	8	20
Отчет по лабораторной работе	4	5	6	15
Итого максимум за период	16	26	28	70
Экзамен				30
Нарастающим итогом	16	42	70	100

11.2. Пересчет баллов в оценки за контрольные точки

Пересчет баллов в оценки за контрольные точки представлен в таблице 11.2.

Таблица 11. 2 – Пересчет баллов в оценки за контрольные точки

Баллы на дату контрольной точки	Оценка
≥ 90% от максимальной суммы баллов на дату КТ	5
От 70% до 89% от максимальной суммы баллов на дату КТ	4
От 60% до 69% от максимальной суммы баллов на дату КТ	3
< 60% от максимальной суммы баллов на дату КТ	2

11.3. Пересчет суммы баллов в традиционную и международную оценку

Пересчет суммы баллов в традиционную и международную оценку представлен в таблице 11.3.

Таблица 11. 3 – Пересчет суммы баллов в традиционную и международную оценку

Оценка (ГОС)	Итоговая сумма баллов, учитывает успешно сданный экзамен	Оценка (ECTS)
5 (отлично) (зачтено)	90 - 100	A (отлично)
4 (хорошо) (зачтено)	85 - 89	B (очень хорошо)
	75 - 84	C (хорошо)

Согласована на портале № 9637

	70 - 74	D (удовлетворительно)
3 (удовлетворительно) (зачтено)	65 - 69	
	60 - 64	E (посредственно)
2 (неудовлетворительно) (не зачтено)	Ниже 60 баллов	F (неудовлетворительно)

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основная литература

1. Технология разработки программного обеспечения: Учебное пособие / Калайда В. Т., Романенко В. В. - 2012. 220 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/2076, свободный.
2. Новые технологии в программировании: Учебное пособие / Гарайс Д. В., Горяинов А. Е., Калентьев А. А. - 2014. 176 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/5796, свободный.
3. Теория и технологии программирования: Курс лекций / Панов С. А. - 2015. 116 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/5013, свободный.

12.2. Дополнительная литература

1. Мирютов А. А. Проектирование программных систем: учебное пособие / А. А. Мирютов. – Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники, Кафедра электронных систем. - Томск: ТУСУР, 2008. - 233 с. (наличие в библиотеке ТУСУР - 100 экз.)
2. Буч, Гради. Объектно-ориентированный анализ и проектирование с примерами приложений на C++: Пер. с англ. / Гради Буч; Ред. пер. И. Романовский, Ред. пер. Ф. Андреев. - 2-е изд. - М.: БИНОМ, 2000; СПб. : Невский Диалект, 2000. - 360 с.: ил. - Библиогр. в конце глав. - Библиогр.: с. 479-548. -Предм. указ.: с. 549-558. - ISBN 5-7989-0067-3 (в пер.). - ISBN 5-7940-0017-1. (наличие в библиотеке ТУСУР - 1 экз.)

12.3. Учебно-методическое пособие и программное обеспечение

1. Теория и технологии программирования: Методические указания к лабораторным работам / Панов С. А. - 2015. 28 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/5014, свободный.
2. Теория и технология программирования: Методические указания по самостоятельной работе / Панов С. А. - 2015. 11 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/5015, свободный.

12.4. Базы данных, информационно справочные и поисковые системы

1. База данных для хранения и редактирования методических материалов, задач и параметров.

13. Материально-техническое обеспечение дисциплины

8 ПК, сборники с описаниями лабораторных работ.

14. Фонд оценочных средств

Фонд оценочных средств приведен в приложении 1.

15. Методические рекомендации по организации изучения дисциплины

Без рекомендаций.

Согласована на портале № 9637

Приложение 1

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

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

УТВЕРЖДАЮ
Проректор по учебной работе
_____ П. Е. Троян
«___» _____ 20__ г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Теория и технология программирования

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки (специальность): **27.03.03 Системный анализ и управление**
Направленность (профиль): **Системный анализ и управление в информационных технологиях**
Форма обучения: **очная**
Факультет: **ФВС, Факультет вычислительных систем**
Кафедра: **МиСА, Кафедра моделирования и системного анализа**
Курс: **3**
Семестр: **6**

Учебный план набора 2015 года

Разработчики:

- старший преподаватель каф. МиСА Панов С. А.

Экзамен: 6 семестр

Томск 2017

Согласована на портале № 9637

1. Введение

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины (практики) 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 (практике) используется при проведении текущего контроля успеваемости и промежуточной аттестации студентов.

Перечень закрепленных за дисциплиной (практикой) компетенций приведен в таблице 1.

Таблица 1 – Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенций
ПК-6	способностью создавать программные комплексы для системного анализа и синтеза сложных систем	Должен знать основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; основные принципы отладки и тестирования программных продуктов; методы и средства разработки технической документации.; Должен уметь осуществлять разработку кода программного модуля на современных языках программирования; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; оформлять документацию на программные средства; использовать инструментальные средства для автоматизации оформления документации.; Должен владеть навыками разработки как сложных компьютерных программ, так и отдельных их модулей.;
ОПК-7	способностью к освоению новой техники, новых методов и новых технологий	
ОПК-1	готовностью применять методы математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также методов гуманитарных, экономических и социальных наук	

Общие характеристики показателей и критериев оценивания компетенций на всех этапах приведены в таблице 2.

Таблица 2 – Общие характеристики показателей и критериев оценивания компетенций 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	Контролирует работу, проводит оценку, совершенствует действия работы
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

Согласована на портале № 9637

			решении проблем
Удовлетворительно (пороговый уровень)	Обладает базовыми общими знаниями	Обладает основными умениями, требуемыми для выполнения простых задач	Работает при прямом наблюдении

2 Реализация компетенций

2.1 Компетенция ПК-6

ПК-6: способностью создавать программные комплексы для системного анализа и синтеза сложных систем.

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания представлены в таблице 3.

Таблица 3 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	основы объектно-ориентированного программирования в Ruby, особенности создания классов и их использование в разрабатываемых приложениях, основы проектирования объектно-ориентированного программного обеспечения с использованием языка Ruby.	разрабатывать программное обеспечение средствами объектно-ориентированного языка Ruby и использовать его на практике.	навыками программирования на языке Ruby.
Виды занятий	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену;	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену;	• Лабораторные занятия; • Самостоятельная работа;
Используемые средства оценивания	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки; • Экзамен;	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки; • Экзамен;	• Отчет по лабораторной работе; • Экзамен; • Экзамен;

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

Таблица 4 – Показатели и критерии оценивания компетенции на этапах

Состав	Знать	Уметь	Владеть
Отлично (высокий уровень)	• определяет набор свойств и методов	• строит диаграмму классов приложения;	• свободно модифицирует

Согласована на портале № 9637

	объекта для разработки соответствующего класса с помощью объектно-ориентированного программирования; • описывает структуру будущего приложения;	• выбирает инструментальные средства разработки приложений;	разработанные приложения; • свободно применяет инструментальные средства разработки приложений;
Хорошо (базовый уровень)	• излагает основные принципы объектно-ориентированного программирования;	• умеет корректно выражать и аргументированно обосновывать положения предметной области знания;	• разрабатывает программное обеспечение средствами объектно-ориентированного языка;
Удовлетворительно (пороговый уровень)	• дает определения основных терминов и понятий;	• умеет работать со справочной литературой; • умеет представлять результаты своей работы;	• владеет терминологией предметной области знания;

2.2 Компетенция ОПК-7

ОПК-7: способностью к освоению новой техники, новых методов и новых технологий.

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	Владеть
Содержание этапов	основы математики, физики и информатики.	работать в среде современных операционных систем; ставить и решать прикладные задачи с использованием современных информационно-коммуникационных технологий.	навыками решения задач в области математики, физики и информатики.
Виды занятий	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену;	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену;	• Лабораторные занятия; • Самостоятельная работа;
Используемые средства оценивания	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки;	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки;	• Отчет по лабораторной работе; • Экзамен; • Экзамен;

Согласована на портале № 9637

	• Экзамен;	• Экзамен;	

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 6.

Таблица 6 – Показатели и критерии оценивания компетенции на этапах

Состав	Знать	Уметь	Владеть
Отлично (высокий уровень)	• анализирует связи между различными понятиями;	• свободно применяет методы решения задач в незнакомых ситуациях; • умеет математически выражать и аргументированно доказывать положения предметной области знания;	• способен руководить междисциплинарной командой;
Хорошо (базовый уровень)	• понимает связи между различными понятиями; • аргументирует выбор метода решения задачи; составляет план решения задачи; • графически иллюстрирует задачу;	• самостоятельно подбирает и готовит для эксперимента необходимое оборудование; • применяет методы решения задач в незнакомых ситуациях; • умеет корректно выражать и аргументированно обосновывать положения предметной области знания;	• критически осмысливает полученные знания; • компетентен в различных ситуациях (работа в междисциплинарной команде);
Удовлетворительно (пороговый уровень)	• дает определения основных понятий; • воспроизводит основные факты, идеи; • распознает объекты; • знает основные методы решения типовых задач и умеет их применять на практике;	• умеет работать со справочной литературой; • использует приборы, указанные в описании лабораторной работы; • умеет представлять результаты своей работы;	• владеет терминологией предметной области знания;

2.3 Компетенция ОПК-1

ОПК-1: готовностью применять методы математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также методов гуманитарных, экономических и социальных наук.

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания представлены в таблице 7.

Таблица 7 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	основы математики, физики, химии, системного анализа,	использовать теоретические знания при объяснении	навыками исследований в области математики, физики, химии,

Согласована на портале № 9637

	теории управления, теории знаний, теории и технологии программирования, а также гуманитарных, экономических и социальных наук.	результатов экспериментов, применять знания в области математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также гуманитарных, экономических и социальных наук для освоения общепрофессиональных дисциплин и решения профессиональных задач.	системного анализа, теории управления, теории знаний, теории и технологии программирования, а также гуманитарных, экономических и социальных наук.
Виды занятий	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену;	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену;	• Лабораторные занятия; • Самостоятельная работа;
Используемые средства оценивания	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки; • Экзамен;	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки; • Экзамен;	• Отчет по лабораторной работе; • Экзамен; • Экзамен;

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 8.

Таблица 8 – Показатели и критерии оценивания компетенции на этапах

Состав	Знать	Уметь	Владеть
Отлично (высокий уровень)	• анализирует связи между различными понятиями; • представляет способы и результаты использования различных моделей; • математически обосновывает выбор метода и план решения задачи;	• свободно применяет методы решения задач в незнакомых ситуациях; • умеет математически выражать и аргументированно доказывать положения предметной области знания;	• способен руководить междисциплинарной командой; • свободно владеет разными способами представления физической информации в графической и математической форме;
Хорошо (базовый уровень)	• понимает связи между различными понятиями; • имеет представление	• самостоятельно подбирает и готовит для эксперимента необходимое	• компетентен в различных ситуациях (работа в междисциплинарной

Согласована на портале № 9637

	о моделях; • аргументирует выбор метода решения задачи; составляет план решения задачи; • графически иллюстрирует задачу;	оборудование; • применяет методы решения задач в незнакомых ситуациях; • умеет корректно выражать и аргументированно обосновывать положения предметной области знания;	команде); • владеет разными способами представления информации; • критически осмысливает полученные знания;
Удовлетворительно (пороговый уровень)	• дает определения основных понятий; • воспроизводит основные факты, идеи; • распознает объекты; • знает основные методы решения типовых задач и умеет их применять на практике;	• умеет работать со справочной литературой; • использует приборы, указанные в описании лабораторной работы; • умеет представлять результаты своей работы;	• владеет терминологией предметной области знания; • способен корректно представить знания в математической форме;

3 Типовые контрольные задания

Для реализации вышеперечисленных задач обучения используются 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 следующем составе.

3.1 Вопросы на самоподготовку

- Основные этапы решения задач на ЭВМ.
- Методология проектирования ПО.
- Испытание, сопровождение, документирование ПО.
- Разработка ПО на языках сверхвысокого уровня.
- Теория вычислительных процессов.
- Компьютерная графика.
- Межпроцессное взаимодействие в ОС Windows.
- Метрология и качество программного обеспечения.
- Объектно-ориентированное программирование.
- Программирование (процедурное).
- Разработка приложений в .Net Framework (C#).
- Разработка распределенных приложений (Java).
- Технология разработки программного обеспечения.
- Управление данными с помощью ADO.Net, ASP.Net.

3.2 Темы опросов на занятиях

- Основные этапы решения задач на ЭВМ.
- Методология проектирования ПО.
- Испытание, сопровождение, документирование ПО.
- Разработка ПО на языках сверхвысокого уровня.
- Теория вычислительных процессов.
- Компьютерная графика.
- Межпроцессное взаимодействие в ОС Windows.
- Метрология и качество программного обеспечения.
- Объектно-ориентированное программирование.

Согласована на портале № 9637

– Программирование (процедурное).
– Разработка приложений в .Net Framework (C#).
– Разработка распределенных приложений (Java).
– Технология разработки программного обеспечения.
– Управление данными с помощью ADO.Net, ASP.Net.

3.3 Экзаменационные вопросы

– 1. Для чего используется класс Time? 2. Что будет выведено на экран (в консоль), если выполнить программу со следующим содержимым: puts ‘12’ + 12? 3. За что отвечает библиотека ActiveRecord? 4. Для чего используется класс Hash? 5. Что будет выведено на экран (в консоль), если выполнить программу со следующим содержимым: puts 1 > 2? 6. Что такое gem и каково его назначение? 7. Для чего используется класс String? 8. Назовите команду для генерации случайного числа. 9. Что такое GitHub? 10. Для чего используется класс Array? 11. Для чего используется метод to_s, и к каким типам данных он применим? 12. Для чего в шаблоне (макете) веб-приложения Ruby on Rails используется ключевое слово yield? 13. Для чего используется библиотека Math? 14. Перечислите основные типы данных (классы) в языке программирования Ruby. 15. Что такое HAML и ERB, и для чего они используются? 16. Для чего используется метод rand? 17. Какое расширение должно быть у файлов, содержащих исходный код на языке программирования Ruby? 18. Какие репозитарии (хранилища исходного кода) приложений Ruby on Rails Вы знаете? 19. Для чего используется метод reverse? 20. Напишите формат однострочной проверки. 21. Как расшифровывается аббревиатура MVC и что она означает? 22. Для чего используется метод srand? 23. Какие операторы ветвления Вы знаете? 24. Приведите пример маршрута. 25. Для чего используется метод length? 26. Для чего используются команда new в языке программирования Ruby? 27. Что такое модель, контроллер и представление? 28. Для чего используется метод upcase? 29. Для чего используются классы Fixnum, Bignum и Float? 30. В какой папке хранятся контроллеры в веб-приложении Ruby on Rails? 31. Для чего используется метод downcase? 32. Как в языке Ruby обозначается операция «не равно»? 33. В какой папке хранятся модели в веб-приложении Ruby on Rails? 34. Для чего используется метод swapcase? 35. Как запрещено именовать переменные в языке программирования Ruby? 36. Для чего используются миграции (migration)? 37. Для чего используется метод capitalize? 38. Для чего используются операции if и else? 39. В какой папке хранятся представления и шаблоны (макеты) в веб-приложении Ruby on Rails? 40. Для чего используется метод puts? 41. В каких случаях на экран (в консоль) выводится значение nil? 42. Как обозначается действие (action) в контроллере Ruby on Rails? 43. Для чего используется метод gets? 44. Для чего используется метод each? 45. Для чего используются маршруты в приложениях Ruby on Rails?

3.4 Темы лабораторных работ

– Методика составления ТЗ
– Изучение методик проектирования
– Инструменты и методы тестирования ПО
– Разработка на функциональных языках программирования.
– Изучение работы компиляторов, трансляторов

4 Методические материалы

Для обеспечения процесса обучения и решения задач обучения используются следующие материалы:

– методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы фор-мирования компетенций, согласно п. 12 рабочей программы.

4.1. Основная литература

1. Технология разработки программного обеспечения: Учебное пособие / Калайда В. Т., Романенко В. В. - 2012. 220 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/2076, свободный.

2. Новые технологии в программировании: Учебное пособие / Гарайс Д. В., Горяинов А. Е., Калентьев А. А. - 2014. 176 с. [Электронный ресурс] - Режим доступа:

Согласована на портале № 9637

https://edu.tusur.ru/publications/5796, свободный.

3. Теория и технологии программирования: Курс лекций / Панов С. А. - 2015. 116 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/5013, свободный.

4.2. Дополнительная литература

1. Мирютов А. А. Проектирование программных систем: учебное пособие / А. А. Мирютов. – Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники, Кафедра электронных систем. - Томск: ТУСУР, 2008. - 233 с. (наличие в библиотеке ТУСУР - 100 экз.)

2. Буч, Гради. Объектно-ориентированный анализ и проектирование с примерами приложений на C++: Пер. с англ. / Гради Буч; Ред. пер. И. Романовский, Ред. пер. Ф. Андреев. - 2-е изд. - М.: БИНОМ, 2000; СПб. : Невский Диалект, 2000. - 360 с.: ил. - Библиогр. в конце глав. - Библиогр.: с. 479-548. -Предм. указ.: с. 549-558. - ISBN 5-7989-0067-3 (в пер.). - ISBN 5-7940-0017-1. (наличие в библиотеке ТУСУР - 1 экз.)

4.3. Учебно-методическое пособие и программное обеспечение

1. Теория и технологии программирования: Методические указания к лабораторным работам / Панов С. А. - 2015. 28 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/5014, свободный.

2. Теория и технология программирования: Методические указания по самостоятельной работе / Панов С. А. - 2015. 11 с. [Электронный ресурс] - Режим доступа: https://edu.tusur.ru/publications/5015, свободный.

4.4. Базы данных, информационно справочные и поисковые системы

1. База данных для хранения и редактирования методических материалов, задач и параметров.

Согласована на портале № 9637