МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

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

УТВЕРЖДАЮ
Проректор по учебной работе

_____ **П. Е. Троян**
«6» _____ 2016 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Теория и технология программирования

Направление(я) подготовки 27.03.03 «Системный анализ и управление»

Направленность (профиль) Системный анализ и управление в информационных технологиях

Квалификация (степень) бакалавр

Форма обучения очная

Факультет ВС, вычислительных систем

Кафедра МиСА, моделирования и системного анализа

Курс 3 **Семестр** 5, 6

Учебный план набора 2014 года

Распределение рабочего времени:

№	Виды учебной работы	Семестр 1	Семестр 2	Семестр 3	Семестр 4	Семестр 5	Семестр 6	Семестр 7	Семестр 8	Всего	Единицы
1.	**Лекции**					18	17			35	**часов**
2.	**Лабораторные работы**					18	17			35	**часов**
3.	**Практические занятия**					-	-			-	**часов**
4.	**Курсовой проект/работа (КРС) (аудиторная)**					-	-			-	**часов**
5.	**Всего аудиторных занятий (Сумма 1-4)**					36	34			70	**часов**
6.	**Самостоятельная работа студентов (СРС)**					36	38			74	**часа**
7.	**Всего (без экзамена) (Сумма 5,7)**					72	72			144	**часа**
8.	**Самост. работа на подготовку, сдачу экзамена**					-	36			36	**часов**
9.	**Общая трудоемкость (Сумма 8,9)**					72	108			180	**часов**
	(в зачетных единицах)					2	3			5	**ЗЕТ**

Зачет 5 **семестр** **Дифф. зачет** нет **семестр**

Экзамен 6 **семестр**

Томск 2016

Согласована на портале № 7657

Рассмотрена и одобрена на заседании кафедры
протокол № 26 от « 7 » 6 2016 г.

Лист согласований

Рабочая программа составлена с учетом требований Федерального Государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки 27.03.03 Системный анализ и управление (уровень бакалавриата), утвержденного приказом Минобрнауки России 11.03.2015 г. № 195, рассмотрена и утверждена на заседании кафедры « 7 » 06 2016 г., протокол № 26 .

Разработчик ст. преподаватель каф. МиСА _____ С.А. Панов

Зав. кафедрой МиСА, д.т.н. _____ В.М. Дмитриев

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

Декан ФВС _____ Л.А. Козлова

Зав. профилирующей и
Выпускающей кафедрой МиСА _____ В.М. Дмитриев

Эксперты:

доцент каф. МиСА	_____	Ганджа Т.В.
(место работы)	(занимаемая должность)	(инициалы, фамилия)
_____	_____	_____
(место работы)	(занимаемая должность)	(инициалы, фамилия)

Согласована на портале № 7657

1. Цели и задачи дисциплины.

Целью данного курса является изучение основных принципов программирования, разработки программного обеспечения (ПО), введения в теорию жизненного цикла ПО, изучения теории автоматов и программирование на языках высокого и сверхвысокого уровня. Курс дает основные знания по следующим вопросам: построение этапов разработки ПО, методы конструирования ПО, методология разработки ПО, изучение программных сред для разработки ПО, тестирование и отладка программного кода.

2. Место дисциплины в структуре ОПОП:

Дисциплина «Теория и технология программирования» относится к базовой части Блока 1 «Дисциплины (модули)» (Б1.Б.19).

Изучение данной дисциплины базируется на следующих дисциплинах: «Информатика», «Программирование и основы алгоритмизации», «Вычислительные машины, системы и сети», «Объектно-ориентированное программирование», «Вычислительная математика», «Пакеты прикладных программ».

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин: «Надежность информационных систем».

3. Требования к результатам освоения дисциплины:

Процесс изучения дисциплины направлен на формирование следующих компетенций:

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

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 зачетных единиц.

Вид учебной работы	Всего часов	Семестр	
		V	VI
Аудиторные занятия (всего)	36	34	70
В том числе:	-	-	-
Лекции	35	18	17
Лабораторные работы (ЛР)	35	18	17
Практические занятия (ПЗ)	-	-	-
Семинары (С)	-	-	-
Коллоквиумы (К)	-	-	-
Курсовой проект/(работа) (аудиторная нагрузка)	-	-	-
Самостоятельная работа (всего)	74	36	38
В том числе:	-	-	-
Подготовка отчетов по лабораторным работам			
Проработка лекционного материала			
Изучение тем (вопросов) теоретической части курса,			

Согласована на портале № 7657

отводимых на самостоятельную проработку			
Подготовка к экзамену	36	-	36
Вид промежуточной аттестации	-	Зачёт	Экзамен
Общая трудоемкость час	180	72	108
Зачетные Единицы Трудоемкости	5	2	3

5. Содержание дисциплины

5.1. Разделы дисциплин и виды занятий

№ п/п	Наименование раздела дисциплины	Лекции	Лаборат. занятия	Практич. занятия.	Курсовой П/Р (КРС)	Самост. работа студента	Всего час. (без экзам)	Формируемые компетенции (ОК, ПК, ПСК)
Семестр 5								
1.	Основные этапы решения задач на ЭВМ	6	6	-	-	12	24	ОПК-1, ОПК-7, ПК-6
2.	Методология проектирования ПО	6	6	-	-	12	24	ОПК-1, ОПК-7, ПК-6
3.	Испытание, сопровождение, документирование ПО	6	6	-	-	12	24	ОПК-1, ОПК-7, ПК-6
Семестр 6								
4.	Разработка ПО на языках сверхвысокого уровня	8	8	-	-	19	35	ОПК-1, ОПК-7, ПК-6
5.	Теория вычислительных процессов	9	9	-	-	19	37	ОПК-1, ОПК-7, ПК-6
	Итого	35	35	-	-	74	144	

5.2. Содержание разделов дисциплины (по лекциям)

№ п/п	Наименование разделов	Содержание разделов	Трудоемкость (час.)	Формируемые компетенции (ОК, ПК, ПСК)
Семестр 5				
1.	Основные этапы решения задач на ЭВМ	Принципы разработки ТЗ. Выбор программных средств для разработки ПО. Оценка качества программного продукта.	6	ОПК-1, ОПК-7, ПК-6
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.	6	ОПК-1, ОПК-7, ПК-6
3.	Испытание, сопровождение, документирование ПО	Источники ошибок в ПО. Отладка, тестирование программ. Сопровождение ПО. Документирование программного продукта.	6	ОПК-1, ОПК-7, ПК-6
Семестр 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.	8	ОПК-1, ОПК-7, ПК-6

Согласована на портале № 7657

5.	Теория вычислительных процессов	Задачи трансляторов. Компиляторы, трансляторы, препроцессоры. Теория Хомского. Языки и грамматики. Распознающие автоматы. Теория контекстно-свободных языков.	9	ОПК-1, ОПК-7, ПК-6

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

№ п/п	Наименование обеспечивающих (предыдущих) и обеспечиваемых (последующих) дисциплин	№ № разделов данной дисциплины из табл.5.1, для которых необходимо изучение обеспечивающих (предыдущих) и обеспечиваемых (последующих) дисциплин				
		1	2	3	4	5
Предыдущие дисциплины						
1.	Информатика	+	+	+	+	+
2.	Программирование и основы алгоритмизации	+	+	+	+	+
3.	Вычислительные машины, системы и сети	+				
4.	Объектно-ориентированное программирование		+	+	+	+
5.	Вычислительная математика	+	+	+	+	+
6.	Пакеты прикладных программ					+
Последующие дисциплины						
1.	Надежность информационных систем	+		+		

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

Перечень компетенций	Виды занятий			Формы контроля по всем видам занятий
	Л	Лаб	СРС	
ОПК-1	+	+	+	Опрос, конспект, отчет по лабораторной работе, зачёт, экзамен
ОПК-7	+	+	+	Опрос, конспект, отчет по лабораторной работе, зачёт, экзамен
ПК-6	+	+	+	Опрос, конспект, отчет по лабораторной работе, зачёт, экзамен

Л – лекция, Пр – практические и семинарские занятия, Лаб – лабораторные работы, КР/КП – курсовая работа/проект, СРС – самостоятельная работа студента

6. Лабораторный практикум

№ п/п	№ раздела дисциплины из табл. 5.1	Наименование лабораторных работ	Трудо-емкость (час.)	Компетенции ОК, ПК, ПСК
Семестр 5				
1.	1	Методика составления ТЗ	6	ОПК-1, ОПК-7, ПК-6
2.	2	Изучение методик проектирования	6	ОПК-1, ОПК-7, ПК-6
3.	3	Инструменты и методы тестирования ПО	6	ОПК-1, ОПК-7, ПК-6
Семестр 6				
4.	4	Разработка на функциональных языках программирования.	8	ОПК-1, ОПК-7, ПК-6
5.	5	Изучение работы компиляторов,	9	ОПК-1, ОПК-7,

Согласована на портале № 7657

		трансляторов		ПК-6

7. Практические занятия (семинары)

Практические занятия не предусмотрены учебным планом.

8. Самостоятельная работа

№ п/п	№ раздела дисциплины из табл. 5.1	Виды самостоятельной работы	Трудо-емкость (час.)	Компе-тенции ОК, ПК, ПСК	Контроль выполнения работы (Опрос, тест, дом.задание, и т.д)
Семестр 5, 6					
1.	1-5	Подготовка отчетов по лабораторным работам	30	ОПК-1, ОПК-7, ПК-6	Отчет по лабораторной работе
2.	1-5	Проработка лекционного материала	30	ОПК-1, ОПК-7, ПК-6	Опрос, конспект
3.	1-5	Изучение тем (вопросов) теоретической части курса, отводимых на самостоятельную проработку	14	ОПК-1, ОПК-7, ПК-6	Опрос, конспект, зачёт, экзамен
4.	1-5	Подготовка к экзамену	36	ОПК-1, ОПК-7, ПК-6	Оценка на экзамене

Темы для самостоятельного изучения:

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.
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.

9. Примерная тематика курсовых проектов (работ)

Курсовая работа не предусмотрена учебным планом.

10. Рейтинговая система для оценки успеваемости студентов

Таблица 10.1 - Дисциплина «Теория и технология программирования» 5 семестр (зачёт, лекции, лабораторные работы)

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
Посещение занятий	3	3	3	**9**
Тестовый контроль	4	4	4	**12**
Лабораторные работы	20	25	22	**67**
Компонент своевременности	4	4	4	**12**
Итого максимум за период:	**31**	**36**	**33**	**100**
Нарастающим итогом	**31**	**67**	**100**	**100**

Оценка «зачтено» проставляется студентам, набравшим 60 баллов и более в зачетную неделю, при условии сдачи всех лабораторных работ.

Согласована на портале № 7657

Таблица 10.2 – Дисциплина «Теория и технология программирования» 6 семестр (экзамен, лекции, лабораторные работы)

Элементы учебной деятельности	Максимальный балл на 1-ую КТ с начала семестра	Максимальный балл за период между 1 КТ и 2 КТ	Максимальный балл за период между 2 КТ и на конец семестра	Всего за семестр
Посещение занятий	4	8	8	**20**
Выполнение и защита результатов лабораторных работ	8	10	12	**30**
Компонент своевременности	4	8	8	**20**
Итого максимум за период:	**16**	**26**	**28**	**70**
Сдача экзамена				**30**
Нарастающим итогом	**16**	**42**	**70**	**100**

Экзаменационный балл (30 баллов) формируется с учетом письменного ответа на три (3) вопроса. Каждый вопрос имеет вес 10 баллов.

Пример: Билет № 1

1) Для чего используется класс *Time*? (10 баллов)

2) Что будет выведено на экран (в консоль), если выполнить программу со следующим содержимым: *puts '12' + 12*? (10 баллов)

3) За что отвечает библиотека *ActiveRecord*? (10 баллов)

Примечания:

Сдача экзамена является обязательной.

Неудовлетворительной сдачей экзамена считается экзаменационная составляющая менее 10 баллов. При неудовлетворительной сдаче экзамена (< 10 баллов) или неявке по неуважительной причине на экзамен экзаменационная составляющая приравнивается к нулю (0). В этом случае студент в установленном в университете порядке обязан пересдать экзамен.

Таблица 10.3 – 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

Таблица 10.4 – 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)
	70 - 74	D (удовлетворительно)
3 (удовлетворительно) (зачтено)	**65 – 69**	
	60 - 64	E (посредственно)
2 (неудовлетворительно), (не зачтено)	**Ниже 60 баллов**	F (неудовлетворительно)

11. Учебно-методическое и информационное обеспечение дисциплины:

11.1 Основная литература:

1. Калентьев А. А. Новые технологии в программировании: учебное пособие / А. А. Калентьев. — Томск: Эль Контент, 2014. — 176 с. (1 экз. в библ.)
2. Технология разработки программного обеспечения: учебное пособие / В. Т. Калайда, В. В. Романенко. - 2012. 220 с. [Электронный ресурс: http://edu.tusur.ru/training/publications/2076]

Согласована на портале № 7657

11.2 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.: ил. – Библиогр.: с. 233. (100 экз. в библ.)
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 (1 экз. в библ.)

11.3 Учебно-методические пособия и требуемое программное обеспечение:

Для лабораторных работ

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

Для самостоятельной работы

1. Панов С. А. Теория и технология программирования: Методические указания по самостоятельной работе / С. А. Панов. – 2015. 11 с. [Электронный ресурс: https://edu.tusur.ru/training/publications/5015]
2. Панов С. А. Теория и технологии программирования: Курс лекций / С. А. Панов. – 2015. 116 с. [Электронный ресурс: https://edu.tusur.ru/training/publications/5013]

Программное обеспечение

1. Среда разработки приложений «NetBeans IDE».
2. Язык программирования сверхвысокого уровня «Ruby» с веб-фреймворком «Rails».
3. СУБД MySQL.

11.4 Базы данных, информационно-справочные и поисковые системы:

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

12. Материально-техническое обеспечение дисциплины:

- 8 ПК, сборники с описаниями лабораторных работ.

13. Методические рекомендации по организации изучения дисциплины: не предусмотрены.

БИБЛИОТЕКА
ТУСУР

Согласована на портале № 7657

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**
«6» 09 2016 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Теория и технология программирования

Направление(я) подготовки 27.03.03 «Системный анализ и управление»

Направленность (профиль) Системный анализ и управление в информационных технологиях

Квалификация (степень) бакалавр

Форма обучения очная

Факультет ВС, вычислительных систем

Кафедра МиСА, моделирования и системного анализа

Курс 3 **Семестр** 5, 6

Учебный план набора 2014 **года**

Зачет 5 **семестр** **Диф. зачет** нет **семестр**

Экзамен 6 **семестр**

Томск 2016

Согласована на портале № 7657

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 компетенции
ОПК-1	готовность применять методы математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также методы гуманитарных, экономических и социальных наук	Должен знать основы математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также гуманитарных, экономических и социальных наук. Должен уметь использовать теоретические знания при объяснении 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. Должен владеть навыками исследований в области математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также гуманитарных, экономических и социальных наук.
ОПК-7	способность к освоению новой техники, новых методов и новых технологий	Должен знать основы математики, физики и информатики. Должен уметь работать в среде современных операционных систем; ставить и решать прикладные задачи с использованием современных информационно-коммуникационных технологий. Должен владеть навыками решения задач в области математики, физики и

Согласована на портале № 7657

		информатики.
ПК-6	способность создавать программные комплексы для системного анализа и синтеза сложных систем	Должен знать 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. Должен уметь разрабатывать программное обеспечение средствами объектно-ориентированного языка Ruby и использовать его на практике. Должен владеть навыками программирования на языке Ruby.

Реализация компетенций

1 Компетенция ОПК-1

ОПК-1: готовность применять методы математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также методы 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 2.

Таблица 2– Этапы формирования компетенции и используемые средства оценивания

1. Состав	**Знать**	**Уметь**	**Владеть**
Содержание этапов	Знает основы математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также гуманитарных, экономических и социальных наук.	Умеет использовать теоретические знания при объяснении 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 общепрофессиональны	Владеет навыками исследований в области математики, физики, химии, системного анализа, теории управления, теории знаний, теории и технологии программирования, а также гуманитарных, экономических и социальных наук.

Согласована на портале № 7657

		х дисциплин и решения профессиональных задач.	
Виды занятий	• Лекции	• Лабораторные работы; • Самостоятельная работа студентов	• Лабораторные работы
Используемые средства оценивания	• Зачёт; • Экзамен	• Отчёт и защита лабораторных работ; • Конспект самостоятельной работы	• Защита лабораторных работ; • Зачет; • Экзамен

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

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

Согласована на портале № 7657

Таблица 4 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	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 о моделях; • 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); • владеет разными способами представления информации
Удовлетворительно (пороговый уровень)	• дает определения основных понятий; • воспроизводит основные факты,	• умеет работать со справочной литературой; • использует приборы,	• владеет терминологией предметной области знания; • способен корректно

Согласована на портале № 7657

	идеи; • распознает объекты; • знает основные методы решения типовых задач и умеет их применять на практике	указанные в описании лабораторной работы; • умеет представлять результаты своей работы	представить знания в математической форме

2 Компетенция ОПК-7

ОПК-7: способность 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

2. Состав	Знать	Уметь	Владеть
Содержание этапов	Знает основы математики, физики и информатики.	Умеет работать в среде современных операционных систем; ставить и решать прикладные задачи с использованием современных информационно-коммуникационных технологий.	Владеет навыками решения задач в области математики, физики и информатики.
Виды занятий	• Лекции	• Лабораторные работы; • Самостоятельная работа студентов	• Лабораторные работы
Используемые средства оценивания	• Зачёт; • Экзамен	• Отчёт и защита лабораторных работ; • Конспект самостоятельной работы	• Защита лабораторных работ; • Зачёт; • Экзамен

Согласована на портале № 7657

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

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	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	• способен руководить междисциплинарной командой

Согласована на портале № 7657

		области знания	
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

3 Компетенция ПК-6

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

Согласована на портале № 7657

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания представлены в таблице 8.

Таблица 8 – Этапы формирования компетенции и используемые средства оценивания

3. Состав	Знать	Уметь	Владеть
Содержание этапов	Знает 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.	Умеет разрабатывать программное обеспечение средствами объектно-ориентированного языка Ruby и использовать его на практике.	Владеет навыками программирования на языке Ruby.
Виды занятий	• Лекции	• Лабораторные работы; • Самостоятельная работа студентов	• Лабораторные работы
Используемые средства оценивания	• Зачёт; • Экзамен	• Отчёт и защита лабораторных работ; • Конспект самостоятельной работы	• Защита лабораторных работ; • Зачёт; • Экзамен

Общие характеристики показателей и критериев оценивания компетенции на всех этапах приведены в таблице 9.

Таблица 9 – Общие характеристики показателей и критериев оценивания компетенции по этапам

Показатели и критерии	Знать	Уметь	Владеть
Отлично (высокий уровень)	Обладает фактическими и теоретическими	Обладает диапазоном практических умений, требуемых для	Контролирует работу, проводит оценку, совершенствует

Согласована на портале № 7657

	знаниями в пределах изучаемой области с пониманием границ применимости	развития творческих решений, абстрагирования проблем	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 решении проблем
Удовлетворительно (пороговый уровень)	Обладает базовыми общими знаниями	Обладает основными умениями, требуемыми для выполнения простых задач	Работает при прямом наблюдении

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 10.

Таблица 10 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	**Знать**	**Уметь**	**Владеть**
Отлично (высокий уровень)	• определяет набор свойств и методов объекта для разработки соответствующего класса с помощью объектно-ориентированного программирования; • описывает структуру будущего приложения	• строит диаграмму классов приложения; • выбирает инструментальные средства разработки приложений	• свободно модифицирует 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

Согласована на портале № 7657

Удовлетворительно (пороговый уровень)	• дает определения основных терминов и понятий	• умеет работать со справочной литературой; • умеет представлять результаты своей работы	• владеет терминологией предметной области знания

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:

Темы лабораторных работ:

1. Методика составления технического задания.
2. Изучение методик проектирования.
3. Инструменты и методы тестирования программного обеспечения.
4. Разработка на функциональных языках программирования.
5. Изучение работы компиляторов, трансляторов.

Темы для самостоятельной работы:

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*.
8. Метрология и качество программного обеспечения.
9. Объектно-ориентированное программирование.

Согласована на портале № 7657

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*.

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*?

Согласована на портале № 7657

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*?

Согласована на портале № 7657

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*?

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 формирования компетенций, в составе:

Методические материалы:

Этап «Должен знать»

1. Калентьев А. А. Новые технологии в программировании: учебное пособие / А. А. Калентьев. — Томск: Эль Контент, 2014. — 176 с. (1 экз. в библ.)

2. Технология разработки программного обеспечения: учебное пособие / В. Т. Калайда, В. В. Романенко. - 2012. 220 с. [Электронный ресурс: http://edu.tusur.ru/training/publications/2076]

3. 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.: ил. – Библиогр.: с. 233. (100 экз. в библ.)

4. 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 (1 экз. в библ.)

5. Панов С. А. Теория и технологии программирования: Курс лекций / С. А. Панов. – 2015. 116 с. [Электронный ресурс: https://edu.tusur.ru/training/publications/5013]

Этап «Должен уметь»

Согласована на портале № 7657

1. Панов С. А. Теория и технология программирования: Методические указания по самостоятельной работе / С. А. Панов. – 2015. 11 с. [Электронный ресурс: https://edu.tusur.ru/training/publications/5015]

2. Панов С. А. Теория и технологии программирования: Методические указания к лабораторным работам / С. А. Панов. – 2015. 28 с. [Электронный ресурс: https://edu.tusur.ru/training/publications/5014]

Этап «Должен владеть»

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

Согласована на портале № 7657