МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

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

УТВЕРЖДАЮ

Проректор по учебной работе

_____ П.Е. Троян

«___» _____ 2016 г.

РАБОЧАЯ ПРОГРАММА

учебной дисциплины

«Объектно-ориентированные языки и системы программирования»

Уровень основной образовательной программы магистратура

Направление(я) подготовки (специальность): Прикладная математика и информатика 01.04.02

Магистерская программа Математическое и программное обеспечение вычислительных комплексов и компьютерных сетей

Форма обучения очная

Факультет факультет систем управления

Кафедра автоматизированных систем управления

Курс 1

Семестр 1-2

Учебный план набора 2015 года и последующих лет.

Распределение рабочего времени:

Виды учебной работы	Семестр 1	Семестр 2	Единицы
Лекции	12	6	часов
Лабораторные работы	18	36	часов
Практические занятия	не предусмотрено	18	часов
Курсовой проект/работа (КРС) (аудиторная)	не предусмотрено	8	часов
Всего аудиторных занятий	30	68	часов
Из них в интерактивной форме	8	8	часов
Самостоятельная работа студентов (СРС)	42	40	часов
Всего (без экзамена)	72	108	часов
Самост. работа на подготовку, сдачу экзамена	36	не предусмотрено	часов
Общая трудоемкость	108	108	часов
(в зачетных единицах)	3	3	ЗЕТ

Экзамен 1 семестр **Диф. зачет 2 семестр**

Зачет не предусмотрено

Томск 2016

Согласована на портале № 6836

Рассмотрена и одобрена на заседании кафедры
протокол № 2 от « 1 » 12 2016 г.

Рабочая программа составлена с учетом требований Федерального Государственного образовательного стандарта высшего профессионального образования (ФГОС ВПО) по направлению подготовки 01.04.02 Прикладная математика и информатика
(квалификация (степень) "магистр"), утвержденного Приказом Министерства образования и науки Российской Федерации от 28 августа 2015 г. N 911,

рассмотрена и утверждена на заседании кафедры АСУ

протокол № 5 от “ 12 ” февраля 2016 г.

Разработчик

Доцент каф. АСУ, к.т.н. _____Е.Б. Грибанова

Зав. обеспечивающей кафедрой АСУ

д.т.н., профессор _____ А.М. Кориков

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами

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

Заведующий профилирующей и выпускающей

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

Эксперты:

Доцент кафедры АСУ ТУСУР _____ А.И. Исакова

Согласована на портале № 6836

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Дисциплина «Объектно-ориентированные языки и системы программирования» читается в 1-2 семестрах и предусматривает чтение лекций, проведение лабораторных занятий, выполнение курсового проекта и получение различного рода консультаций.

Целью дисциплины является формирование у студентов систематизированных знаний и практических навыков в области объектно-ориентированного программирования.

Основной **задачей** изучения дисциплины является формирование у студентов теоретических знаний и практических навыков объектно-ориентированного программирования с использованием языка высокого уровня Java.

В результате изучения курса студенты должны усвоить следующие понятия и определения: объектно-ориентированная парадигма, класс, объект, доступ, поля и методы, проектирование объектно-ориентированных программ, объектно-ориентированные языки, архитектура, а также свободно владеть технологией объектно-ориентированного программирования. Важным навыками должны стать умение программировать, используя объектно-ориентированный подход на языке Java, а также умение описывать и читать архитектуру классов и объектов на языке UML.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Дисциплина «Объектно-ориентированные языки и системы программирования» (ООЯСП) относится к числу дисциплин профессионального цикла. Успешное овладение дисциплиной предполагает предварительные знания операционных систем, сетей ЭВМ и телекоммуникаций, предусмотренном специальностью «Прикладная математика и информатика», а также навыки программирования на языках высокого уровня**.**

Знания, полученные при изучении данной дисциплины, будут в дальнейшем использоваться в научно-исследовательской работе и при изучении современных компьютерных технологий.

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины «Объектно-ориентированные языки и системы программирования» направлен на формирование следующих компетенций:

общепрофессиональные компетенции (ОПК):

- способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям (ОПК-3).

профессиональные компетенции (ПК):

- способностью разрабатывать концептуальные и теоретические модели решаемых научных проблем и задач (ПК-2);

В результате изучения дисциплины студент должен:

Знать:

- принципы объектно-ориентированной разработки программ;
- способы описания программы на языке моделирования.

Уметь:

- пользоваться принципами объектно-ориентированной разработки для написания программ на языке высокого уровня;

Владеть:

- объектно-ориентированной технологией разработки программ;
- языком высокого уровня Java и средой разработки Eclipse;
- языком моделирования систем UML.

Согласована на портале № 6836

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины составляет **6** зачетных единиц.

Вид учебной работы	Всего часов	Семестры			
		1	2	3	4
Аудиторные занятия (всего)	**98**	**30**	**68**		
В том числе:	–		–		
Лекции	18	12	6		
Лабораторные работы (ЛР)	54	18	36		
Практические занятия (ПЗ)	18	–	18		
Семинары (С)	–	–	–		
Коллоквиумы (К)					
Курсовой проект (работа) (аудиторная нагрузка)	8	–	8		
Другие виды аудиторной работы					
Самостоятельная работа (всего)	**82**	**42**	**40**		
В том числе:	–		–		
Курсовой проект (работа) (самостоятельная работа)	20		20		
Расчетно-графические работы	–		–		
Реферат	–				
Другие виды самостоятельной работы					
Проработка лекционного материала	20	20	–		
Подготовка к лабораторным занятиям	30	10	20		
Самостоятельное изучение тем теоретической части	12	12	–		
Подготовка к экзамену	**36**	**36**			
Вид промежуточной аттестации (зачет, экзамен)		экзамен			
Общая трудоемкость час	**216**	**108**	**108**		
зач. ед.	**6**	**3**	**3**		

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

5.1. Разделы дисциплин и виды занятий

Таблица 5.1

№ п/п	Наименование раздела дисциплины	Лекции	Лаборат. занятия	Практич. занятия	Самост. работа студентов	Курсовой проект	Всего часов	Формируемые компетенции (ОПК, ПК)
1	2	3	4	5	7	8	9	10
	Семестр 1.	**12**	**18**		**42**		**72**	
1.	Тема 1. Системы объектно-ориентированного программирования. Технология Java	1	2		5		8	ОПК-3, ПК-2
2.	Тема 2. Интегрированная среда разработки Eclipse	1	2		5		8	ОПК-3, ПК-2
3.	Тема 3. Объектно-ориентированное программирование на Java	1	2		5		8	ОПК-3, ПК-2

Согласована на портале № 6836

4.	Тема 4. Типы данных, переменные, операторы языка Java	1	2		5		8	ОПК-3, ПК-2
5	Тема 5. Создание и использование объектов языка Java. Классы и объекты String. Классы и объекты Java API	1	2		5		8	ОПК-3, ПК-2
6.	Тема 6. Операторы управления ходом программы на языке Java.	2	2		5		9	ОПК-3, ПК-2
7	Тема 7. Разработка методов, инкапсуляция и конструкторы, массивы объектов	2	3		5		10	ОПК-3, ПК-2
8	Тема 8. Наследование, интерфейсы и полиморфизм	3	3		7		13	ОПК-3, ПК-2
	Семестр 2	**6**	**36**	**18**	**40**	**8**	**108**	
9	Тема 9. Язык объектно-ориентированного моделирования UML	2	3		5		10	ОПК-3, ПК-2
10	Тема 10. Механизм исключений и операции ввода/вывода	1	3	3	5		12	ОПК-3, ПК-2
11	Тема 11. Графический интерфейс пользователя на основе библиотеки Swing	1	10	5	10	2	28	ОПК-3, ПК-2
12	Тема 12. Шаблоны проектирования	1	10	5	10	3	29	ОПК-3, ПК-2
13	Тема 13. Web-программирование. Java - апплеты	1	10	5	10	3	29	ОПК-3, ПК-2

5.2. Содержание разделов дисциплины (по лекциям)

Таблица 5.2

№ п/п	Наименование разделов	Содержание разделов	Трудоемкость (час.)	Формируемые компетенции (ОПК, ПК)
1	2	3	4	5
1.	Системы объектно-ориентированного программирования. Технология Java	Обзор систем объектно-ориентированного программирования. История развития Java. Основные понятия Java.	1	ОПК-3, ПК-2
2.	Интегрированная среда разработки Eclipse	Работа со средой Eclipse. Навигация ресурсов. Инструмент разработки Java Development Kit (JDK)	1	ОПК-3, ПК-2
3.	Объектно-ориентированное программирование на Java	Понятие объекта. Понятие класса, отношения между классами. Понятие наследования. Понятие интерфейса. Понятие пакета.	1	ОПК-3, ПК-2
4.	Типы данных, переменные, операторы языка Java	Идентификаторы, типы данных, литералы. Приведение типов. Объявление и инициализация переменных. Массивы примитивных типов.	1	ОПК-3, ПК-2

Согласована на портале № 6836

		Операторы Java.		
5.	Создание и использование объектов языка Java. Классы и объекты String. Классы и объекты Java API	Объявление, создание экземпляров, инициализация, хранение объектов в оперативной памяти. Использование класса String. Спецификация классов Java API.	1	ОПК-3, ПК-2
6.	Операторы управления ходом программы на языке Java	Операторы цикл. Операторы условия. Оператор выбора.	2	ОПК-3, ПК-2
7.	Разработка методов, инкапсуляция и конструкторы, массивы объектов	Разработка методов в классах Java. Инкапсуляция и конструкторы. Массивы и объекты.	2	ОПК-3, ПК-2
8.	Наследование, интерфейсы и полиморфизм	Наследование. Интерфейсы и абстрактные классы. Полиморфизм. Коллекции и генерики в Java.	3	ОПК-3, ПК-2
9	Язык объектно-ориентированного моделирования UML	Основные элементы языка UML. Виды диаграмм: диаграммы классов, объектов, последовательности, состояний, вариантов использования.	2	ОПК-3, ПК-2
10	Механизм исключений и операции ввода/вывода	Механизм исключений. Файловый ввод/вывод. Сетевое взаимодействие.	1	ОПК-3, ПК-2
11	Графический интерфейс пользователя на основе библиотеки Swing	Разработка графического интерфейса с применением компонента Visual Editor среды Eclipse	1	ОПК-3, ПК-2
12	Шаблоны проектирования	Виды шаблонов проектирования. Разработка программ с использованием шаблонов проектирования.	1	ОПК-3, ПК-2
13.	Web-программирование. Java - апплеты	Апплеты на Java.	1	ОПК-3, ПК-2

5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (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
	Предшествующие дисциплины													
1.	Языки и методы программирования		+	+	+	+	+	+		+		+	+	

Согласована на портале № 6836

2.	Операционные системы	+				+					+			
3.	Программное обеспечение ЭВМ и сетей	+				+			+		+			+
	Последующие дисциплины													
1.	Научно-исследовательская работа			+	+	+	+	+	+	+	+	+	+	+
2.	Современные компьютерные технологии	+	+										+	+

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

Перечень компетенций	Л	Лаб	СРС	КП	Формы контроля
ОПК-3		+	+	+	Устный опрос на лекции, устный ответ по практической и лабораторной работе, тест
ПК-2	+	+	+	+	Отчеты по лабораторным и практическим работам, тест, пояснительная записка к курсовому проекту

Л – лекция, Лаб – лабораторные работы, ПЗ- практические занятия, СРС – самостоятельная работа студента, КП – курсовой проект

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

Формы / Методы	Лекции (час)	Лабораторные занятия (час)	Всего (час)
Работа в команде	4		**4**
Пресс-конференция		4	**4**
Поисковый метод		8	**8**
Итого интерактивных занятий			**16**

Примечание.

1. «Работа в команде» происходит при коллективном построении архитектуры системы на языке UML (раздел №9 из табл. 5.1).
2. «Поисковый метод» студенты используют при выборе шаблонов проектирования (лаб. работа № 8).
3. Основные результаты своих лабораторных работ (наиболее интересные исследования) студенты докладывают при помощи презентаций, устраивая подобие пресс-конференции.

7. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

№ п/п	№ раздела дисциплины из табл. 5.1	Наименование лабораторных работ	Трудо-емкость (час.)	ОПК, ПК

Согласована на портале № 6836

1.	1	Компиляция и запуск Java-приложений из командной строки	2	ОПК-3
2.	2, 3	Создание и отладка проекта в среде Ecipse. Элементы объектно-ориентированного программирования Java.	4	ОПК-3, ПК-2
3.	4	Объявление, инициализация и использование переменных	2	ОПК-3
4.	5	Создание и использование объектов	2	ОПК-3, ПК-2
5.	6	Операторы управления ходом программ	2	ОПК-3
6.	7	Разработка методов в классах Java, использование инкапсуляции и конструкторов.	2	ОПК-3, ПК-2
7.	7	Создание и использование массивов	1	ОПК-3, ПК-2
8.	8,9	Наследование, переопределение методов, полиморфизм. Интерфейсы и абстрактные классы	6	ОПК-3, ПК-2
9.	10	Механизм исключений и операции ввода/вывода	3	ОПК-3
10.	11	Разработка графического интерфейса в среде Eclipse	3	ОПК-3, ПК-2
11.	12	Разработка программ с использованием шаблонов проектирования	10	ОПК-3, ПК-2
12.	13	Написание Java-апплетов.	10	ОПК-3, ПК-2

8. ПРАКТИЧЕСКИЕ ЗАНЯТИЯ (СЕМИНАРЫ).

№ п/п	№ раздела дисциплины из табл. 5.1	Наименование практических занятий	Трудо-емкость (час.)	ОПК, ПК
1.	10	Механизм исключений и операции ввода/вывода	3	ОПК-3
2.	11	Графический интерфейс пользователя на основе библиотеки Swing	5	ОПК-3, ПК-2
3.	12	Шаблоны проектирования	5	ОПК-3, ПК-2
4.	13	Web-программирование. Java - апплеты	5	ОПК-3, ПК-2

9. САМОСТОЯТЕЛЬНАЯ РАБОТА

№ п/п	№ раздела дисциплины из табл. 5.1	Тематика самостоятельной работы *(детализация)*	Трудо-емкость (час.)	ОПК, ПК	Контроль выполнения работы (Опрос, тест, дом.задание, и т.д)
1.	1÷13	Проработка лекционного материала	20	ОПК-3, ПК-2	Опрос на занятиях (устно)
2.	1÷13	Подготовка к лабораторным	30	ОПК-3, ПК-2	Отчет, защита лаб.

Согласована на портале № 6836

		занятиям			работы
3	11÷13	Работа над курсовым проектом	20	ОПК-3, ПК-2	Пояснительная записка, защита курсовой работы
4.	9, 11, 12, 13	Самостоятельное изучение тем теоретической части ***Темы для самостоятельного изучения*** 1) Язык UML. Диаграммы обзора взаимодействия, синхронизации, пакетов, компонентов. 2) Классы Java для работы в Интернет.	12	ОПК-3, ПК-2	Дом. задание, тест
5.	1÷13	Подготовка и сдача экзамена	36	ОПК-3, ПК-2	Оценка за экзамен

10. КУРСОВОЙ ПРОЕКТ

№ п/п	**Элементы учебной деятельности**	Трудо-емкость (час.)	ОПК, ПК
1.	Подбор и обзор литературы	2	ОПК-3, ПК-2
2.	Выполнение необходимых заданий по проекту	3	ОПК-3, ПК-2
3.	Полное оформление работы	2	ОПК-3, ПК-2
4.	Защита курсового проекта	1	ОПК-3, ПК-2

11. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ

1. Логическая игра на языке Java с использованием стандартных шаблонов проектирования.
2. Электронная открытка на языке Java с использованием стандартных шаблонов проектирования.
3. Решение производственных задач с помощью программ, разработанных с использованием объектно-ориентированного подхода.
4. Решение научных задач с помощью, разработанных с использованием объектно-ориентированного подхода.
5. Java – апплеты для web-страниц.

Согласована на портале № 6836

11. БАЛЛЬНО-РЕЙТИНГОВАЯ СИСТЕМА

11.1 Курс 1, семестр 1
Контроль обучения – Экзамен.

Максимальный семестровый рейтинг – **100 баллов.**

По дисциплине «Объектно-ориентированные языки и системы программирования» (ООЯСП) проведение экзамена является **обязательным** в 1 семестре. При этом балльная оценка в соотношении **70/30** распределяется на две составляющие: **семестровую** и **экзаменационную.** Т.е. **70 баллов** можно получить за текущую работу в семестре, а **30 баллов** – за ответы на экзамене.

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 (тесты, лабораторные работы, коллоквиумы).

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

Текущий контроль изучения дисциплины состоит из контроля за усвоением теоретического материала – проведение **3** тестов.

В таблице 11.1 содержится распределение баллов в течение 1 семестра для дисциплины «Объектно-ориентированные языки и системы программирования» (ООЯСП), завершающейся экзаменом и содержащей 9 лекций (18 часов), 8 лабораторных работ (16 часов), проводимых в течение 1 семестра и 3 итоговых теста во время проведения двух контрольных точек. В таблице 11.2 представлен пересчет суммы баллов по 1 и 2 контрольной точке в традиционную оценку. В таблице 11.3 – представлен пересчет итоговой суммы баллов в традиционную и международную оценку.

Таблица 11.1 – Дисциплина «Объектно-ориентированные языки и системы программирования» (ООЯСП) **(экзамен)**

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**
Выполнение и защита результатов лабораторных и практических работ	10	10	10	**30**
Тестовый контроль	5	5	5	**15**
Компонент своевременности	4	4	5	**13**
Итого максимум за период:	**23**	**23**	**24**	**70**
Нарастающим итогом	**23**	**46**	**70**	
Экзамен				**30**
ИТОГО				**100**

По результатам текущего контроля формируется допуск студента к итоговому контролю – экзамену по дисциплине. Экзамен осуществляется в форме опроса по теоретической части дисциплины. В составе суммы баллов, полученной студентом по дисциплине, заканчивающейся **экзаменом**, экзаменационная составляющая должна быть не менее 10 баллов. В противном случае экзамен считается не сданным, студент в установленном в ТУСУРе порядке обязан его пересдать.

Методика выставления баллов за ответы на **экзамене** определяется из расчета **10 баллов** за каждый из **3 вопросов в билете**.

Согласована на портале № 6836

Неудовлетворительной сдачей экзамена считается экзаменационная составляющая **менее 10 баллов**. При неудовлетворительной сдаче экзамена (<10 баллов) или неявке на экзамен экзаменационная составляющая приравнивается к нулю (0).

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

Таблица 11.3 – Пересчет суммы баллов в традиционную и международную оценку

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), (не зачтено)	Ниже 60 баллов	F (неудовлетворительно)

11.2 Курс 1, семестр 2
Контроль обучения – курсовой проект

При выполнении курсового проекта по дисциплине «Объектно-ориентированные языки и системы программирования» студент стимулируется к планомерной работе. Студенту указывается распределение баллов, начисляемых за отдельные этапы работы над курсовым проектом. В таблице 11.4 приведено распределение баллов для отдельных этапов выполнения курсового проекта. В таблице 11.5 представлен пересчет суммы баллов по 1 и 2 контрольной точке в традиционную оценку.

Для курсового проекта отчетная составляющая (до **30** баллов) выставляется студенту по результатам защиты проекта. Допуском к защите курсового проекта является обязательная сдача 10 лабораторных работ (20 часов). При защите курсового проекта учитываются следующие важные моменты.

1) Выбор темы курсовой работы (сложность и творческие моменты).
2) Содержание пояснительной записки.
3) Оформление пояснительной записки.
4) Творческие моменты в реализации программы.
5) Защита проекта (ответы на вопросы).

Если студент по всем вышеперечисленным пунктам показал себя хорошо, то по отчетной составляющей он заслуживает **30** баллов. При наборе отчетной составляющей менее 10 баллов, она приравнивается к нулю. В этом случае курсовой проект подлежит повторной защите в установленном университетом порядке.

Таблица 11.4 – Дисциплина Объектно-ориентированные языки и системы программирования (**Курсовой проект**)

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**	**Всего за семестр**

Согласована на портале № 6836

			семестра	
Получение задания на курсовой проект/работу	**4**			**4**
Подбор и обзор литературы	**12**			**12**
Выполнение необходимых заданий по проекту		**22**		**22**
Полное оформление работы			**20**	**20**
Компонент своевременности	**4**	**4**	**4**	**12**
Защита курсового проекта			**30**	**30**
Итого максимум за период	**20**	**26**	**54**	**100**
Нарастающим итогом	**20**	**46**	**100**	

Если курсовая работа не выполнена в положенной срок (до экзаменационной сессии), то выставляется оценка на балл меньше, чем по полученному рейтингу и рейтинг обнуляется.

Таблица 11.5 – Пересчет суммы баллов по 1 и 2 контрольной точке в традиционную оценку

Оценка (ГОС)	**сумма баллов, на 1-ую контрольную точку с начала семестра**	**сумма баллов, на 2-ую контрольную точку за период между 1КТ и 2КТ**
5 (отлично)	**18 – 20**	**22 – 26**
4 (хорошо)	**15 – 17**	**17 – 21**
3 (удовлетворительно)	**10 – 14**	**12 – 16**
2 (неудовлетворительно)	**Ниже 10 баллов**	**Ниже 12 баллов**

Согласована на портале № 6836

13. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

13.1 Основная литература

1. Объектно-ориентированное программирование: Учебное пособие / Романенко В. В. – 2014. 475 с. [Электронный ресурс]. – Режим доступа: https://edu.tusur.ru/training/publications/4872

13.2 Дополнительная литература

1. Павловская Т.А. C/C++. Программирование на языке высокого уровня: учебник для вузов. - СПб. : Питер, 2009. - 464 с. (4 экз.)
2. Лаптев В.В. C++. Объектно-ориентированное программирование: учебное пособие. - СПб. : Питер, 2008. – 457 с. (3 экз.)
3. Кьоу Д. Объектно-ориентированное программирование. - СПб.: Питер, 2005. – 237 с. (20 экз.)
4. Хорев П.Б. Технологии объектно-ориентированного программирования. – М.: Академия, 2004. – 446 с. (30 экз.)
5. Леоненков А.В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose. - М.: БИНОМ. Лаборатория знаний, 2006. – 318 с. (20 экз.)

13.3 Учебно-методические пособия и программное обеспечение

1. Грибанова Е.Б. Объектно-ориентированные языки и системы программирования. Учебное методическое пособие по лабораторным работам. – Томск: ТУСУР, 2016. – 21с. [Электронный ресурс]. – Режим доступа:
http://asu.tusur.ru/learning/mag010400/d11/m010402_d11_labs.doc

2. Грибанова Е.Б. Методические указания к курсовому проекту по дисциплине «Объектно-ориентированные языки и системы программирования». – Томск: ТУСУР, 2016. – 15 с.– [Электронный ресурс]. – Режим доступа:
/http://asu.tusur.ru/learning/books/ gribanova_objectlang_project.doc
3. Грибанова Е.Б. Методические указания по самостоятельной работе по дисциплине «Объектно-ориентированные языки и системы программирования». – Томск: ТУСУР, 2016. – 9 с.– [Электронный ресурс]. – Режим доступа: http://asu.tusur.ru/learning/books/ gribanova_objectlang_work.doc
4. Шельмина, Е. А. Объектно-ориентированное программирование: Методические указания по выполнению лабораторных и практических работ для студентов 09.03.01 «Информатика и вычислительная техника», 09.03.02 «Информационные системы и технологии» [Электронный ресурс] / Шельмина Е. А. — Томск: ТУСУР, 2015. — 21 с. — Режим доступа: https://edu.tusur.ru/publications/6149.
4. Среда разработки Eclipse, виртуальная машина Java.

Internet-ресурсы

1. http://onlinelibrary.wiley.com - научные журналы издательства Wiley&Sons
2. http://www.sciencedirect.com/ - научные журналы издательства Elsevier
3. www.intuit.ru - национальный открытый университет
4. el.asu.tusur.ru – электронные курс по дисциплине Основы технологии Java 2.

14. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для проведения теоретического (лекций) материала по дисциплине используются персональный ПК с проектором. Лабораторные занятия осуществляются в компьютерном классе с использованием среды разработки Eclipse.

Согласована на портале № 6836

Приложение к рабочей программе

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

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

УТВЕРЖДАЮ

Проректор по учебной работе

_____ **П. Е. Троян**

«___» _____ 2016 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

ОБЪЕКТНО-ОРИЕНТИРОВАННЫЕ ЯЗЫКИ И СИСТЕМЫ ПРОГРАММИРОВАНИЯ

Уровень основной образовательной программы магистратура

Направление подготовки 01.04.02 – Прикладная математика и информатика

Форма обучения очная

Факультет систем управления

Кафедра автоматизированных систем управления

Курс 1

Семестр 1-2

Учебный план набора 2015 года и последующих лет

Экзамен 1 семестр

Диф.зачет 2 семестр

Томск 2016

Согласована на портале № 6836

1. ВВЕДЕНИЕ

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины «**Объектно-ориентированные языки и системы программирования»** 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 «**Объектно-ориентированные языки и системы программирования»** компетенций приведен в таблице 1.

Таблица 1 – Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенции
ОПК-3	способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям	**Знать:** – Принципы построения алгоритмов, типы данных и базовые конструкции языка программирования Си. **Уметь:** – Работать в современных средах разработки, составлять блок-схемы алгоритмов, создавать программы на структурном языке программирования высокого уровня Си. **Владеть:** – Основными приемами программирования на современном структурном языке

Согласована на портале № 6836

ПК-2	способность разрабатывать концептуальные и теоретические модели решаемых научных проблем и задач	**Знать:** – методы программирования; – создание алгоритма решения программы; – основную цель и задачи программирования; – применимость основ программирования для дальнейшей деятельности; – порядок выполнения программ. **Уметь:** – применять имеющиеся знания для решения практических задач и тестовых заданий; – связывать программирование с математическими моделями; – пользоваться различными режимами при работе с языками программирования. **Владеть:** – сведениями о других существующих языках программирования; – навыками программирования в конкретных ситуациях и в зависимости от поставленной цели.

Согласована на портале № 6836

2. РЕАЛИЗАЦИЯ КОМПЕТЕНЦИЙ

2.1 Компетенция ОПК-3

ОПК-3: способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.

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

Состав	Знать	Уметь	Владеть
Содержание этапов	- Знает принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	– Умеет разрабатывать алгоритмические и программные решения в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	– Владеет основными приемами разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.
Виды занятий	– Лекции; – Лабораторные работы – Групповые консультации	– Лабораторные работы; – Самостоятельная работа студентов	– Лабораторные работы; – Самостоятельная работа студентов
Используемые средства	– Тест; – Экзамен	– Конспект самостоятельной работы	– Экзамен

Согласована на портале № 6836

оценивания			

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 (низкий уровень)	Обладает низким уровнем общих знаний	Обладает умениями на низком уровне, которые не достаточны для выполнения даже простых задач	Работает только при прямом наблюдении

Согласована на портале № 6836

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)	– Глубоко понимает принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	– Умеет уверенно разрабатывать алгоритмические и программные решения в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	– Владеет основными приемами разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.
ХОРОШО (базовый уровень)	– Хорошо понимает принципы разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств	– Умеет разрабатывать алгоритмические и программные решения в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования	– Хорошо владеет основными приемами разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей,

Согласована на портале № 6836

	тестирования систем и средств на соответствие стандартам и исходным требованиям.	систем и средств на соответствие стандартам и исходным требованиям.	образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	– Имеет общие представления о принципах разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	– Умеет разрабатывать некоторые алгоритмические и программные решения в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.	– Слабо владеет основными приемами разработки алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, создания информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям.

2.2 Компетенция ПК-2

ПК-2: способность разрабатывать концептуальные и теоретические модели решаемых научных проблем и задач.

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

Таблица 5 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть

Согласована на портале № 6836

Содержание этапов	– Знает методы разработки алгоритмических и программных решений в области системного и прикладного программного обеспечения.	– Умеет применять алгоритмические и программные решения в области системного и прикладного программного обеспечения.	– Владеет методами разработки алгоритмических и программных решений в области системного и прикладного программного обеспечения.
Виды занятий	– Лекции; – Лабораторные работы – Групповые консультации	– Лабораторные работы; – Самостоятельная работа студентов	– Лабораторные работы; – Самостоятельная работа студентов
Используе-мые средства оценивания	– Тест; – Экзамен	– Конспект самостоятельной работы	– Экзамен – Диф.зачет

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

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 твор-ческих 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 (низкий уровень)	Обладает низким уровнем общих знаний	Обладает умениями на низком уровне, которые не достаточны для выполнения даже простых задач	Работает только при прямом наблюдении

Согласована на портале № 6836

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

Таблица 7 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	Знать	Уметь	Владеть
ОТЛИЧНО (высокий уровень)	– Знает методы разработки концептуальных и теоретических моделей решаемых научных проблем и задач.	– Умеет уверенно применять концептуальные и теоретические модели для решения научных проблем и задач.	– Владеет сведениями о методах разработки концептуальных и теоретических моделей решаемых научных проблем и задач.
ХОРОШО (базовый уровень)	– Знает основные методы разработки концептуальных и теоретических моделей решаемых научных проблем и задач.	Умеет применять концептуальные и теоретические модели для решения научных проблем и задач.	– Владеет на хорошем уровне методами разработки концептуальных и теоретических моделей решаемых научных проблем и задач.
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	– Имеет общие представления о методах разработки концептуальных и теоретических моделей решаемых научных проблем и задач.	– Неуверенно применяет концептуальные и теоретические модели для решения научных проблем и задач.	– Слабо владеет методами разработки концептуальных и теоретических моделей решаемых научных проблем и задач.

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 работ

1. Компиляция и запуск Java-приложений из командной строки
2. Создание и отладка проекта в среде Ecipse. Элементы объектно-ориентированного программирования Java
3. Объявление, инициализация и использование переменных
4. Создание и использование объектов
5. Операторы управления ходом программ
6. Разработка методов в классах Java, использование инкапсуляции и конструкторов
7. Создание и использование массивов
8. Наследование, переопределение методов, полиморфизм. Интерфейсы и абстрактные классы

Согласована на портале № 6836

9. Механизм исключений и операции ввода/вывода
10. Разработка графического интерфейса в среде Eclipse
11. Разработка программ с использованием шаблонов проектирования
12. Написание Java-апплетов.

3.2 Темы самостоятельных работ

1. Язык UML. Диаграммы обзора взаимодействия, синхронизации, пакетов, компонентов.
2. Классы Java для работы в Интернет.
3. Классы Java для решения задач оптимизации.
4. Классы Java для регрессионного анализа.

3.3 Пример типовых вопросов по тестам

1. Поименованная ячейка памяти
2. Порядковый (идентификационный) номер ячейки
3. Переменная, смысл которой хранить адрес некоторой ячейки
4. Переменная, которая указывает на некоторую ячейку
5. Поименованная последовательность действий, часто используемая в программе, имя используется для улучшения читабельности и уменьшения объема исходного кода программы, а так же, использование такого имени уменьшает исполняемый код программы.
6. Поименованная последовательность действий, часто используемая в программе, имя используется для улучшения читабельности и уменьшения объема исходного кода программы, но использование такого имени не уменьшает исполняемый код программы.
7. Запуск подпрограммы на исполнение
8. Завершение работы подпрограммы
9. Параметры подпрограммы или макроса, указываемые при описании подпрограммы или макроса
10. Параметры подпрограммы или макроса, указываемые при вызове подпрограммы или макроса
11. Операция, копирующая значение вычисленного выражения (константы или переменной) указанного справа, в переменную указанную слева
12. Параметр операции
13. Операция над одним операндом
14. Операция над двумя операндами
15. Унарная операция, для которой операнд указывается слева
16. Унарная операция, для которой операнд указывается справа
17. Множество (ряд) ячеек одинакового типа, которые объявлены под одним именем и проиндексированы
18. Переменные, объявленные внутри подпрограммы
19. Переменные, объявленные вне какой-либо подпрограммы, и видимые из любой точки программы
20. Часть оператора if которая проверяется на истинность
21. Часть оператора цикла, которую необходимо выполнять несколько раз
22. Часть оператора цикла, от которого зависит, будет ли выполняться тело цикла
23. Часть оператора цикла, которая выполняется до цикла
24. Часть оператора цикла, присущая только циклу for согласно его формату
25. Часть оператора цикла, используемая, когда заранее известно, сколько раз надо выполнить тело цикла
26. Определение объекта
27. Определение класса
28. Понятие инкапсуляции
29. Понятие наследования

Согласована на портале № 6836

30. Понятие полиморфизма
31. Классы-хранилища данных
32. Классы для реализации интерфейса
33. Конструктор
34. Абстрактный класс
35. Интерфейс
36. Стековая память и куча

3.4 Примерная тематика курсовых проектов

1. Логическая игра на языке Java с использованием стандартных шаблонов проектирования.
2. Электронная открытка на языке Java с использованием стандартных шаблонов проектирования.
3. Решение производственных задач с помощью программ, разработанных с использованием объектно-ориентированного подхода.
4. Решение научных задач с помощью, разработанных с использованием объектно-ориентированного подхода.
5. Java – апплеты для web-страниц.

3.5 Вопросы для подготовки к экзамену по дисциплине «Объектно-ориентированные языки и системы программирования»

1) Типы данных языка Java, приведение типов, объявление и инициализация переменных.
2) Массивы примитивных типов, операторы языка Java.
3) Объявление, создание экземпляров и инициализация.
4) Оператор цикла while, for.
5) Операторы if-else, switch.
6) Перегрузка методов.
7) Инкапсуляция и конструкторы.
8) Наследование.
9) Полиморфизм.
10) Интерфейсы и абстрактные классы.
11) Механизм исключений.
12) Файловый ввод/вывод.
13) Язык UML: история развития, назначение, виды моделируемых диаграмм (статические и динамические).
14) Отношения между классами. Моделирование классов с помощью UML.
15) Отношения между объектами. UML диаграмма объектов.
16) UML диаграмма последовательности.
17) UML диаграмма состояний.
18) UML диаграмма прецедентов.

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, в составе:

1. Учебное пособие по дисциплине «Объектно-ориентированные языки и системы программирования» приведено в рабочей программе в разделе 12.1 [2].
2. Методические указания по самостоятельной и индивидуальной работе студентов всех форм обучения приведены в рабочей программе в разделе 12.3 [6].

Методические указания по лабораторным работам и по самостоятельной работе приведены в рабочей программе в разделе 12.3 [6].

Согласована на портале № 6836

– Объектно-ориентированное программирование: Учебное пособие / Романенко В. В. – 2014. 475 с. [Электронный ресурс]. – Режим доступа: https://edu.tusur.ru/training/publications/4872
– 1. Грибанова Е.Б. Объектно-ориентированные языки и системы программирования. Учебное методическое пособие по лабораторным работам для магистров, обучающихся по направлению подготовки «Прикладная математика и информатика 01.04.02». – Томск: ТУСУР, 2016. – 21с. [Электронный ресурс]. – Режим доступа: http://www.asu.tusur.ru/learning/mag010400/d11/m010400_d11_labs.doc

– 2. Грибанова Е.Б. Методические указания к курсовому проекту по дисциплине «Объектно-ориентированные языки и системы программирования» для магистров, обучающихся по направлению подготовки «Прикладная математика и информатика 01.04.02». – Томск: ТУСУР, 2016. – 15 с.– [Электронный ресурс]. – Режим доступа: http://www.asu.tusur.ru/learning/mag010400/d11/m010400_d11_project.doc

– 3. Грибанова Е.Б. Методические указания по самостоятельной работе по дисциплине «Объектно-ориентированные языки и системы программирования» для магистров, обучающихся по направлению подготовки «Прикладная математика и информатика 01.04.02». – Томск: ТУСУР, 2016. – 9 с.– [Электронный ресурс]. – Режим доступа: http://www.asu.tusur.ru/learning/mag010400/d11/m010400_d11_work.doc

– Шельмина, Е. А. Объектно-ориентированное программирование: Методические указания по выполнению лабораторных и практических работ для студентов 09.03.01 «Информатика и вычислительная техника», 09.03.02 «Информационные системы и технологии» [Электронный ресурс] / Шельмина Е. А. — Томск: ТУСУР, 2015. — 21 с. — Режим доступа: https://edu.tusur.ru/publications/6149.

Согласована на портале № 6836