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

Объектно-ориентированное программирование

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки (специальность): **09.03.01 Информатика и вычислительная техника**
Направленность (профиль): **Программное обеспечение средств вычислительной техники и автоматизированных систем**
Форма обучения: **очная**
Факультет: **ФСУ, Факультет систем управления**
Кафедра: **АСУ, Кафедра автоматизированных систем управления**
Курс: **2**
Семестр: **3, 4**
Учебный план набора 2016 года

Распределение рабочего времени

№	Виды учебной деятельности	3 семестр	4 семестр	Всего	Единицы
1	Лекции	18	16	34	часов
2	Лабораторные занятия	36	34	70	часов
3	Всего аудиторных занятий	54	50	104	часов
4	Из них в интерактивной форме		18	18	часов
5	Самостоятельная работа	54	58	112	часов
6	Всего (без экзамена)	108	108	216	часов
7	Подготовка и сдача экзамена		36	36	часов
8	Общая трудоемкость	108	144	252	часов
		3.0	4.0	7.0	З.Е

Зачет: 3 семестр
Экзамен: 4 семестр

Томск 2016

Согласована на портале № 4182

Рассмотрена и одобрена на заседании кафедры
протокол № 2 от « 8 » 11 2016 г.

ЛИСТ СОГЛАСОВАНИЙ

Рабочая программа составлена с учетом требований Федерального Государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) 09.03.01 Информатика и вычислительная техника, утвержденного 2016-01-12 года, рассмотрена и утверждена на заседании кафедры «___» _____ 20__ года, протокол №_____.

Разработчики:

доцент каф. АСУ _____ Романенко В. В.

Заведующий обеспечивающей каф.
АСУ _____ Кориков А. М.

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

Декан ФСУ _____ Сенченко П. В.

Заведующий выпускающей каф.
АСУ _____ Кориков А. М.

Эксперты:

доцент каф. АСУ _____ Исакова А. И.

Согласована на портале № 4182

1. Цели и задачи дисциплины

1.1. Цели дисциплины

Целью курса является обучение студентов основам объектно-ориентированного проектирования и программирования в современных средах разработки ПО.

1.2. Задачи дисциплины

– Основой задачей изучения курса является получение знаний и практических навыков в области проектирования и разработки объектно-ориентированных программ. В результате изучения курса студент должен иметь представление о предпосылках возникновения ООП и его месте в эволюции парадигм программирования, знать принципы объектно-ориентрованного проектирования и программирования, а также уметь разрабатывать объектно-ориентированные программы на языках C++, C++ CLI и C# с применением библиотек классов STL и .NET.

2. Место дисциплины в структуре ОПОП

Дисциплина «Объектно-ориентированное программирование» (Б1.В.ОД.7) относится к блоку 1 (вариативная часть).

Предшествующими дисциплинами, формирующими начальные знания, являются следующие дисциплины: Дискретная математика, Информатика, Математика, Математическая логика и теория алгоритмов, Программирование, Структуры и алгоритмы обработки данных в ЭВМ.

Последующими дисциплинами являются: .

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

– ОПК-2 Способность осваивать методики использования программных средств для решения практических задач.;

– ПК-1 Способность разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина».;

В результате изучения дисциплины студент должен:

– **знать** Принципы объектно-ориентированного анализа и проектирования, основы объектно-ориентированного подхода к программированию.

– **уметь** Проводить объектную декомпозицию предметной области, писать в современных средах разработки объектно-ориентрованные программы, использовать объектно-ориентированные библиотеки классов при разработке ПО.

– **владеть** Основными приемами объектно-ориентрованного программирования на языках C++, C++ CLI и C#, навыками использования библиотек классов STL, .NET.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 7.0 зачетных единицы и представлена в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры	
		3 семестр	4 семестр
Аудиторные занятия (всего)	104	54	50
Лекции	34	18	16
Лабораторные занятия	70	36	34
Из них в интерактивной форме	18		18
Самостоятельная работа (всего)	112	54	58
Оформление отчетов по лабораторным работам	70	34	36
Проработка лекционного материала	42	20	22

Согласована на портале № 4182

Всего (без экзамена)	216	108	108
Подготовка и сдача экзамена	36		36
Общая трудоемкость час	252	108	144
Зачетные Единицы Трудоемкости	7.0	3.0	4.0

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	Введение	2	0	2	4	ОПК-2, ПК-1
2	Основные понятия	2	0	2	4	ОПК-2, ПК-1
3	Объектно-ориентированный анализ и проектирование	2	0	2	4	ОПК-2, ПК-1
4	Объектная декомпозиция	2	0	2	4	ОПК-2, ПК-1
5	Объектно-ориентированное программирование на языке C++	2	12	10	24	ОПК-2, ПК-1
6	Классы и структуры. Члены классов. Дружественность	2	8	12	22	ОПК-2, ПК-1
7	Перегрузка стандартных операторов	4	8	12	24	ОПК-2, ПК-1
8	Виртуальный и абстрактный полиморфизм. Наследование	2	8	12	22	ОПК-2, ПК-1
9	Шаблоны функций и классов	2	8	10	20	ОПК-2, ПК-1
10	Объектно-ориентированное программирование на языке C#	2	0	4	6	ОПК-2, ПК-1
11	Библиотека .NET. Основы языка C#	2	0	4	6	ОПК-2, ПК-1
12	Классы, структуры и интерфейсы	2	0	2	4	ОПК-2, ПК-1
13	Свойства и индексаторы	2	10	16	28	ОПК-2, ПК-1
14	Делегаты. События	2	0	2	4	ОПК-2, ПК-1
15	Универсальные типы	2	8	10	20	ОПК-2, ПК-1
16	Документирование кода	2	8	10	20	ОПК-2, ПК-1
	Итого	34	70	112	216	

Согласована на портале № 4182

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
3 семестр			
1 Введение	Ведение в предмет. История развития парадигм программирования. Причины возникновения ООП	2	ОПК-2, ПК-1
	Итого	2	
2 Основные понятия	Принципы ООП. Понятия объекта и класса. Понятия члена класса, поля, метода. Инкапсуляция, полиморфизм, наследование	2	ОПК-2, ПК-1
	Итого	2	
3 Объектно-ориентированный анализ и проектирование	Типы отношений между классами. Понятие предметной области решаемой задачи. Анализ предметной области. Программные средства для описания предметной области	2	ОПК-2, ПК-1
	Итого	2	
4 Объектная декомпозиция	Выделение в предметной области объектов и определение отношений между ними. Программные средства для описания объектной декомпозиции	2	ОПК-2, ПК-1
	Итого	2	
5 Объектно-ориентированное программирование на языке C++	Особенности объектной реализации в языке C++. Переход от языка C к языку C++. Программирование на смешанных объектно-ориентированных языках. Указатели на функции и процедуры. Написание динамических библиотек (DLL)	2	ОПК-2, ПК-1
	Итого	2	
6 Классы и структуры. Члены классов. Дружественность	Синтаксис описания класса. Отличия классов и структур в языке C++. Члены класса. Конструкторы и деструкторы. Поля. Методы. Объявление вложенных типов. Объявление дружественности. Статические поля и методы. Указатели на методы классов	2	ОПК-2, ПК-1
	Итого	2	
7 Перегрузка стандартных операторов	Правила перегрузки операций в языке C++. Перегрузка унарных и бинарных операций. Операторы-члены класса и	4	ОПК-2, ПК-1

Согласована на портале № 4182

	внешние операторы. Перегрузка операторов приведения типа		
	Итого	4	
8 Виртуальный и абстрактный полиморфизм. Наследование	Таблицы виртуальных функций. Виртуальные и абстрактные методы. Абстрактные классы. Статический и виртуальный полиморфизм. Наследование. Поведение классов при наследовании	2	ОПК-2, ПК-1
	Итого	2	
Итого за семестр		18	
4 семестр			
9 Шаблоны функций и классов	Шаблоны процедур и функций. Шаблоны структур и классов. Шаблоны констант и типов	2	ОПК-2, ПК-1
	Итого	2	
10 Объектно-ориентированное программирование на языке C#	Основы программирования на языке C#. Безопасность кода. Управляемые ресурсы. Типы данных по значению и ссылочные типы данных. Полностью объектно-ориентированные языки программирования. Язык C++ CLI	2	ОПК-2, ПК-1
	Итого	2	
11 Библиотека .NET. Основы языка C#	Операторы языка C#. Операторы выражений. Идентификаторы. Форматирование и разбор строк. Консольный ввод-вывод. Файловый ввод-вывод. Сериализация и десериализация	2	ОПК-2, ПК-1
	Итого	2	
12 Классы, структуры и интерфейсы	Пространства имен. Сборки. Синтаксис описания классов, структур и интерфейсов. Наследование и полиморфизм в языке C#. Вложенные типы	2	ОПК-2, ПК-1
	Итого	2	
13 Свойства и индексаторы	Поля класса. Синтаксис описания свойств и индексаторов в классах и интерфейсах. Статические поля	2	ОПК-2, ПК-1
	Итого	2	
14 Делегаты. События	Методы класса. Указатели на методы классов. Синтаксис описания делегатов и событий. Статические и анонимные методы	2	ОПК-2, ПК-1
	Итого	2	
15 Универсальные типы	Параметры типа. Ограничения параметров типа. Универсальные	2	ОПК-2, ПК-1

Согласована на портале № 4182

	методы, классы и интерфейсы. Наследование универсальных типов		
	Итого	2	
16 Документирование кода	Расстановка в коде тегов документирования. Сборка XML-документации. Генерация файлов документации. Разработка проектной документации	2	ОПК-2, ПК-1
	Итого	2	
Итого за семестр		16	
Итого		0	

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															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Предшествующие дисциплины																	
1	Дискретная математика				+	+		+									
2	Информатика	+	+			+					+	+					+
3	Математика							+									
4	Математическая логика и теория алгоритмов					+		+	+	+	+						
5	Программирование					+	+	+	+	+	+	+	+	+	+	+	
6	Структуры и алгоритмы обработки данных в ЭВМ				+	+	+	+			+		+				

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

	Виды занятий	Формы контроля

Согласована на портале № 4182

Компетенции	Лекции	Лабораторные занятия	Самостоятельная работа	
ОПК-2	+	+	+	Экзамен, Конспект самоподготовки, Защита отчета, Отчет по лабораторной работе, Компонент своевременности, Опрос на занятиях
ПК-1	+	+	+	Экзамен, Конспект самоподготовки, Защита отчета, Отчет по лабораторной работе, Компонент своевременности, Опрос на занятиях

6. Интерактивные методы и формы организации обучения

Технологии интерактивного обучения при разных формах занятий в часах приведены в таблице 6.1

Таблица 6.1 – Технологии интерактивного обучения при разных формах занятий в часах

Методы	Интерактивные лабораторные занятия	Интерактивные лекции	Всего
IT-методы	4	1	5
Работа в команде	8		8
Деловые игры	4	1	5
Итого	16	2	18

7. Лабораторный практикум

Содержание лабораторных работ приведено в таблице 7.1.

Таблица 7. 1 – Содержание лабораторных работ

Названия разделов	Содержание лабораторных работ	Трудоемкость, ч	Формируемые компетенции
3 семестр			
5 Объектно-ориентированное программирование на языке C++	Анализ предметной области. Проектирование класса в рамках предметной области	12	ОПК-2, ПК-1
	Итого	12	
6 Классы и структуры. Члены классов. Дружественность	Инкапсуляция объектов линейной алгебры (вектор, матрица) в классе. Перегрузка стандартных операций	8	ОПК-2, ПК-1

Согласована на портале № 4182

	Итого	8	
7 Перегрузка стандартных операторов	Инкапсуляция математических объектов (дробь, полином) в классе. Перегрузка стандартных операций	8	ОПК-2, ПК-1
	Итого	8	
8 Виртуальный и абстрактный полиморфизм. Наследование	Декомпозиция предметной области. Создание иерархии классов	8	ОПК-2, ПК-1
	Итого	8	
Итого за семестр		36	
4 семестр			
9 Шаблоны функций и классов	Создание шаблонов классов	8	ОПК-2, ПК-1
	Итого	8	
13 Свойства и индексаторы	Инкапсуляция объектов линейной алгебры в классе. Перегрузка стандартных операций	10	ОПК-2, ПК-1
	Итого	10	
15 Универсальные типы	Инкапсуляция математических объектов в универсальных классах	8	ОПК-2, ПК-1
	Итого	8	
16 Документирование кода	Документирование кода класса	8	ОПК-2, ПК-1
	Итого	8	
Итого за семестр		34	
Итого		0	

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
3 семестр				
1 Введение	Проработка лекционного материала	2	ОПК-2, ПК-1	Опрос на занятиях, Экзамен
	Итого	2		
2 Основные понятия	Проработка лекционного материала	2	ОПК-2, ПК-1	Опрос на занятиях, Экзамен
	Итого	2		
3 Объектно-ориентированный	Проработка лекционного материала	2	ОПК-2, ПК-1	Опрос на занятиях, Экзамен

Согласована на портале № 4182

анализ и проектирование	Итого	2		
4 Объектная декомпозиция	Проработка лекционного материала	2	ОПК-2, ПК-1	Опрос на занятиях, Экзамен
	Итого	2		
5 Объектно-ориентированное программирование на языке C++	Проработка лекционного материала	2	ОПК-2, ПК-1	Защита отчета, Компонент своевременности, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Оформление отчетов по лабораторным работам	8		
	Итого	10		
6 Классы и структуры. Члены классов. Дружественность	Проработка лекционного материала	4	ОПК-2, ПК-1	Защита отчета, Компонент своевременности, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Оформление отчетов по лабораторным работам	8		
	Итого	12		
7 Перегрузка стандартных операторов	Проработка лекционного материала	4	ОПК-2, ПК-1	Защита отчета, Компонент своевременности, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Оформление отчетов по лабораторным работам	8		
	Итого	12		
8 Виртуальный и абстрактный полиморфизм. Наследование	Проработка лекционного материала	2	ОПК-2, ПК-1	Защита отчета, Компонент своевременности, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Оформление отчетов по лабораторным работам	10		
	Итого	12		
Итого за семестр		54		
4 семестр				
9 Шаблоны функций и классов	Проработка лекционного материала	2	ОПК-2, ПК-1	Защита отчета, Компонент своевременности, Конспект самоподготовки, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Оформление отчетов по лабораторным работам	8		
	Итого	10		
10 Объектно-ориентированное программирование на языке C#	Проработка лекционного материала	4	ОПК-2, ПК-1	Опрос на занятиях, Экзамен
	Итого	4		
11 Библиотека .NET. Основы языка C#	Проработка лекционного материала	4	ОПК-2, ПК-1	Конспект самоподготовки, Опрос на занятиях, Экзамен
	Итого	4		
12 Классы, структуры и интерфейсы	Проработка лекционного материала	2	ОПК-2, ПК-1	Опрос на занятиях, Экзамен
	Итого	2		

Согласована на портале № 4182

13 Свойства и индексаторы	Проработка лекционного материала	4	ОПК-2, ПК-1	Защита отчета, Компонент своевременности, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Оформление отчетов по лабораторным работам	12		
	Итого	16		
14 Делегаты. События	Проработка лекционного материала	2	ОПК-2, ПК-1	Опрос на занятиях, Экзамен
	Итого	2		
15 Универсальные типы	Проработка лекционного материала	2	ОПК-2, ПК-1	Защита отчета, Компонент своевременности, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Оформление отчетов по лабораторным работам	8		
	Итого	10		
16 Документирование кода	Проработка лекционного материала	2	ОПК-2, ПК-1	Защита отчета, Компонент своевременности, Конспект самоподготовки, Опрос на занятиях, Отчет по лабораторной работе, Экзамен
	Оформление отчетов по лабораторным работам	8		
	Итого	10		
Итого за семестр		58		
	Подготовка к экзамену	36		Экзамен
Итого		148		

9.1. Вопросы на проработку лекционного материала

1. Перегрузка унарных и бинарных операций
2. Операторы-члены класса и внешние операторы
3. Перегрузка операторов приведения типа
4. Указатели на функции и процедуры
5. Написание динамических библиотек (DLL)
6. Выделение в предметной области объектов и определение отношений между ними
7. История развития парадигм программирования
8. Принципы ООП
9. Понятия объекта и класса
10. Понятия члена класса, поля, метода
11. Инкапсуляция, полиморфизм, наследование
12. Виртуальные и абстрактные методы
13. Абстрактные классы
14. Статический и виртуальный полиморфизм
15. Наследование
16. Поведение классов при наследовании
17. Синтаксис описания класса
18. Члены класса
19. Конструкторы и деструкторы
20. Поля. Методы
21. Объявление вложенных типов
22. Объявление дружественности
23. Статические поля и методы
24. Указатели на методы классов

Согласована на портале № 4182

25. Типы отношений между классами
26. Понятие предметной области решаемой задачи
27. Анализ предметной области
28. Программные средства для описания предметной области

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
3 семестр				
Защита отчета	10	15	15	40
Компонент своевременности	3	4	4	11
Конспект самоподготовки	3	3	3	9
Отчет по лабораторной работе	10	15	15	40
Итого максимум за период	26	37	37	100
Нарастающим итогом	26	63	100	100
4 семестр				
Защита отчета	5	10	10	25
Компонент своевременности	3	4	4	11
Конспект самоподготовки	3	3	3	9
Отчет по лабораторной работе	5	10	10	25
Итого максимум за период	16	27	27	70
Экзамен				30
Нарастающим итогом	16	43	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

Согласована на портале № 4182

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)
	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. Павловская Т.А. C/C++. Программирование на языке высокого уровня: учебник для вузов. – СПб: Питер, 2013. – 461 с. (наличие в библиотеке ТУСУР - 35 экз.)
2. Павловская Т.А. C#. Программирование на языке высокого уровня: учебник для вузов. – СПб: Питер, 2013. – 432 с. (наличие в библиотеке ТУСУР - 16 экз.)

12.2. Дополнительная литература

1. Орлов С.А. Технологии разработки программного обеспечения: современный курс по программной инженерии. – СПб: Питер, 2012. – 608 с. (наличие в библиотеке ТУСУР - 15 экз.)

12.3. Учебно-методическое пособие и программное обеспечение

1. Программирование на языках высокого уровня: Методические указания по выполнению лабораторных, практических работ и самостоятельной работы для студентов 230100 «Информатика и вычислительная техника» / Панасенко Е. А. - 2012. 12 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/2199, свободный.

12.4. Базы данных, информационно справочные и поисковые системы

1. Библиотека Microsoft Developer Network (http://msdn.microsoft.com/ru-ru).
2. Виртуальная академия Microsoft Virtual Academy (http://www.microsoftvirtualacademy.com/?lang=ru-ru).
3. Интернет-университет ИНТУИТ (http://www.intuit.ru/).

13. Материально-техническое обеспечение дисциплины

Для проведения лекционных занятий предоставляется аудитория с проектором.

Для проведения лабораторных занятий предоставляются два компьютерных класса (18 рабочих мест).

14. Фонд оценочных средств

Фонд оценочных средств приведен в приложении 1.

15. Методические рекомендации по организации изучения дисциплины

Без рекомендаций.

Согласована на портале № 4182

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

Объектно-ориентированное программирование

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки (специальность): **09.03.01 Информатика и вычислительная техника**
Направленность (профиль): **Программное обеспечение средств вычислительной техники и автоматизированных систем**
Форма обучения: **очная**
Факультет: **ФСУ, Факультет систем управления**
Кафедра: **АСУ, Кафедра автоматизированных систем управления**
Курс: **2**
Семестр: **3, 4**

Учебный план набора 2016 года

Разработчики:
- доцент каф. АСУ Романенко В. В.

Зачет: 3 семестр
Экзамен: 4 семестр

Томск 2016

Согласована на портале № 4182

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
ОПК-2	Способность осваивать методики использования программных средств для решения практических задач.	Должен знать Принципы объектно-ориентированного анализа и проектирования, основы объектно-ориентированного подхода к программированию.; Должен уметь Проводить объектную декомпозицию предметной области, писать в современных средах разработки объектно-ориентрованные программы, использовать объектно-ориентированные библиотеки классов при разработке ПО.; Должен владеть Основными приемами объектно-ориентрованного программирования на языках C++, C++ CLI и C#, навыками использования библиотек классов STL, .NET.;
ПК-1	Способность разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина».	

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

Согласована на портале № 4182

2 Реализация компетенций

2.1 Компетенция ОПК-2

ОПК-2: Способность осваивать методики использования программных средств для решения практических задач..

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	Принципы объектно-ориентированного анализа и проектирования	Проводить объектную декомпозицию предметной области, писать в современных средах разработки объектно-ориентированные программы	Основными приемами объектно-ориентированного программирования на языках C++, C++ CLI и C#
Виды занятий	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Интерактивные лабораторные занятия; • Интерактивные лекции; • Подготовка к экзамену;	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Интерактивные лабораторные занятия; • Интерактивные лекции; • Подготовка к экзамену;	• Лабораторные занятия; • Самостоятельная работа; • Интерактивные лабораторные занятия;
Используемые средства оценивания	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки; • Зачет; • Экзамен;	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки; • Зачет; • Экзамен;	• Отчет по лабораторной работе; • Экзамен; • Зачет; • 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)	• Знает все принципы объектно-ориентированного анализа и проектирования;	• Умеет эффективно про-водить объектную декомпозицию предметной области, писать в современных средах разработки объектно-ориентированные программы любого уровня сложности;	• Свободно владеет приемами объектно-ориентированного программирования на языках C++, C++ CLI и C#;

Согласована на портале № 4182

Хорошо (базовый уровень)	• Знает основные принципы объектно-ориентированного анализа и проектирования, достаточные для решения типовых задач;	• Умеет проводить объектную декомпозицию предметной области, писать в современных средах разработки объектно-ориентированные программы среднего уровня сложности;	• В достаточной степени владеет приемами объектно-ориентированного программирования на языках C++ и C#;
Удовлетворительно (пороговый уровень)	• Знает лишь базовые принципы объектно-ориентированного проектирования, достаточные для решения простейших задач;	• Умеет писать в современных средах разработки простые объектно-ориентированные программы;	• На низком уровне владеет приемами объектно-ориентированного программирования на языке C++;

2.2 Компетенция ПК-1

ПК-1: Способность разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина»..

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 объектно-ориентированного подхода к программированию	Использовать объектно-ориентированные библиотеки классов при разработке ПО	Навыками использования библиотек классов STL, .NET
Виды занятий	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Интерактивные лабораторные занятия; • Интерактивные лекции; • Подготовка к экзамену;	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Интерактивные лабораторные занятия; • Интерактивные лекции; • Подготовка к экзамену;	• Лабораторные занятия; • Самостоятельная работа; • Интерактивные лабораторные занятия;
Используемые средства оценивания	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки; • Зачет; • Экзамен;	• Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Конспект самоподготовки; • Зачет; • Экзамен;	• Отчет по лабораторной работе; • Экзамен; • Зачет; • Экзамен;

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

Согласована на портале № 4182

Таблица 6 – Показатели и критерии оценивания компетенции на этапах

Состав	Знать	Уметь	Владеть
Отлично (высокий уровень)	• Знает все аспекты объектно-ориентированного подхода к программированию;	• Умеет использовать различные объектно-ориентированные библиотеки классов при разработке ПО любого уровня сложности;	• Владеет навыками программирования приложений с использованием библиотек классов STL, .NET любого уровня сложности в современных средах программирования;
Хорошо (базовый уровень)	• Знает основные аспекты объектно-ориентированного подхода к программированию;	• Умеет использовать некоторые объектно-ориентированные библиотеки классов при разработке ПО среднего уровня сложности;	• Владеет навыками программирования приложений с использованием библиотек классов STL, .NET среднего уровня сложности в современных средах программирования;
Удовлетворительно (пороговый уровень)	• Знает лишь базовые аспекты объектно-ориентированного подхода к программированию;	• Умеет использовать лишь библиотеки функций при разработке ПО простого уровня;	• Владеет навыками программирования простых приложений с использованием библиотек классов STL хотя бы в одной среде программирования;

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

– Описать класс с двумя полями X и P, инкапсулирующий число X·10^P. Определить операции деления, умножения и возведения в степень таких чисел (/, *, ^).

– Класс инкапсулирует число N, записанное в системе счисления по основанию P ($2 \le P \le 16$). Определить операции вывода числа на консоль (<<) и присваивания строки (=) такому числу.

– Класс инкапсулирует вектор из N элементов. Определить операции сравнения векторов (==, !=, >, >=, <, <=). В качестве критерия сравнения использовать норму векторов.

– Класс инкапсулирует десятичное число, хранящееся в виде строки S, максимальная длина которой равна N. Определить операции сложения (+) и присваивания (=) таких чисел.

– Класс инкапсулирует точку на декартовой плоскости. Определить операции покоординатного сложения и вычитания точек (+, –), а также унарную операцию обращения знака (–).

– Класс инкапсулирует точку на декартовой плоскости. Определить операции поворота точки вокруг центра координат на указанный угол (+=, –=), а также поворота на угол ±π (++, —).

– Класс инкапсулирует двоичное число, хранимое в виде строки S максимальной длины N. Определить операции циклического сдвига двоичного числа вправо или влево, а также инверсии этого числа (<<, >>, ~).

– Класс инкапсулирует прямоугольник со сторонами A и B. Определить операцию «&», соединяющую два прямоугольника горизонтально, если они имеют одинаковую высоту, и

Согласована на портале № 4182

операцию «|», соединяющую два прямоугольника вертикально, если они имеют одинаковую ширину, а также операцию присваивания (=).

– Описать класс с полем P, инкапсулирующий число e^P. Определить операции деления, умножения и возведения в степень таких чисел (/, *, ^), а также их деления и умножения с числами типа double.

– Класс инкапсулирует шар радиуса R. Определить операцию сложения (+), в результате которой получается шар, объем которого равен сумме объемов исходных шаров, а также операцию вычитания (–) по схожему принципу. При получении отрицательного объема выдавать ошибку.

– Класс инкапсулирует дату (в виде номера дня, месяца и года – D, M, Y). Определить операции сравнения дат (<, >), а также увеличения и уменьшения даты на целое количество дней (+=, –=).

– Класс инкапсулирует рациональную дробь (в виде числителя A и знаменателя B). Определить операции сравнения дробей.

– Класс инкапсулирует мнимое число. Определить операции деления, умножения и вывода на экран таких чисел.

– Класс инкапсулирует вектор произвольной размерности. Определить операцию доступа к элементам вектора.

– Описать класс с двумя полями X и P, инкапсулирующий число X, возведенное в степень P (X^P). Определить операции деления, умножения и возведения в степень таких чисел.

3.2 Темы опросов на занятиях

– Перегрузка унарных и бинарных операций
– Операторы-члены класса и внешние операторы
– Перегрузка операторов приведения типа
– Указатели на функции и процедуры
– Написание динамических библиотек (DLL)
– Выделение в предметной области объектов и определение отношений между ними
– История развития парадигм программирования
– Принципы ООП
– Понятия объекта и класса
– Понятия члена класса, поля, метода
– Инкапсуляция, полиморфизм, наследование
– Виртуальные и абстрактные методы
– Абстрактные классы
– Статический и виртуальный полиморфизм
– Наследование
– Поведение классов при наследовании
– Синтаксис описания класса
– Члены класса
– Конструкторы и деструкторы
– Поля. Методы
– Объявление вложенных типов
– Объявление дружественности
– Статические поля и методы
– Указатели на методы классов
– Типы отношений между классами
– Понятие предметной области решаемой задачи
– Анализ предметной области
– Программные средства для описания предметной области

3.3 Экзаменационные вопросы

– Перегрузка унарных и бинарных операций

Согласована на портале № 4182

– Операторы-члены класса и внешние операторы
– Перегрузка операторов приведения типа
– Указатели на функции и процедуры
– Написание динамических библиотек (DLL)
– Выделение в предметной области объектов и определение отношений между ними
– История развития парадигм программирования
– Принципы ООП
– Понятия объекта и класса
– Понятия члена класса, поля, метода
– Инкапсуляция, полиморфизм, наследование
– Виртуальные и абстрактные методы
– Абстрактные классы
– Статический и виртуальный полиморфизм
– Наследование
– Поведение классов при наследовании
– Синтаксис описания класса
– Члены класса
– Конструкторы и деструкторы
– Поля. Методы
– Объявление вложенных типов
– Объявление дружественности
– Статические поля и методы
– Указатели на методы классов
– Типы отношений между классами
– Понятие предметной области решаемой задачи
– Анализ предметной области
– Программные средства для описания предметной области

3.4 Темы лабораторных работ

– Анализ предметной области. Проектирование класса в рамках предметной области
– Инкапсуляция объектов линейной алгебры (вектор, матрица) в классе. Перегрузка стандартных операций
– Инкапсуляция математических объектов (дробь, полином) в классе. Перегрузка стандартных операций
– Декомпозиция предметной области. Создание иерархии классов
– Создание шаблонов классов
– Инкапсуляция объектов линейной алгебры в классе. Перегрузка стандартных операций
– Инкапсуляция математических объектов в универсальных классах
– Документирование кода класса

3.5 Зачёт

– Причины возникновения ООП. Принципы ООП. Плюсы и минусы ООП на языке C++.
– Понятие объекта и класса в языке C++. Типы отношений между классами.
– Объектная декомпозиция.
– Класс и структура в языке C++.
– Синтаксис описания класса на языке C++. Члены класса.
– Объявление локальных типов данных. Пространства имен в языке C++.
– Инкапсуляция. Уровни доступа к членам класса. Объявления дружественности в языке C++.
– Конструкторы и деструкторы в языке C++.
– Поля класса. Инициализация полей. Статические поля в языке C++.
– Методы класса. Виртуальные и статические методы. Абстрактные методы и классы в

Согласована на портале № 4182

языке С++.

– Перегрузка операторов в языке С++.

– Наследование. Поведение и правила использования членов класса при наследовании в языке С++.

– Использование указателей и ссылок в ООП на языке С++. Указатель this.

– Совместное использование членов класса и внешних объектов в языке С++.

– Создание экземпляров классов. Доступ к членам классов в языке С++.

– Указатели на методы классов в языке С++.

– Шаблоны функций и классов в языке С++.

– Преобразование типов данных при работе с классами в языке С++. Явные и неявные преобразования.

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. Павловская Т.А. С/С++. Программирование на языке высокого уровня: учебник для вузов. – СПб: Питер, 2013. – 461 с. (наличие в библиотеке ТУСУР - 35 экз.)
2. Павловская Т.А. С#. Программирование на языке высокого уровня: учебник для вузов. – СПб: Питер, 2013. – 432 с. (наличие в библиотеке ТУСУР - 16 экз.)

4.2. Дополнительная литература

1. Орлов С.А. Технологии разработки программного обеспечения: современный курс по программной инженерии. – СПб: Питер, 2012. – 608 с. (наличие в библиотеке ТУСУР - 15 экз.)

4.3. Учебно-методическое пособие и программное обеспечение

1. Программирование на языках высокого уровня: Методические указания по выполнению лабораторных, практических работ и самостоятельной работы для студентов 230100 «Информатика и вычислительная техника» / Панасенко Е. А. - 2012. 12 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/2199, свободный.

4.4. Базы данных, информационно справочные и поисковые системы

1. Библиотека Microsoft Developer Network (http://msdn.microsoft.com/ru-ru).
2. Виртуальная академия Microsoft Virtual Academy (http://www.microsoftvirtualacademy.com/?lang=ru-ru).
3. Интернет-университет ИНТУИТ (http://www.intuit.ru/).

Согласована на портале № 4182