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

Программирование

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки (специальность): **09.03.01 Информатика и вычислительная техника**
Направленность (профиль): **Автоматизированное управление бизнес-процессами и финансами**
Форма обучения: **очная**
Факультет: **ФВС, Факультет вычислительных систем**
Кафедра: **ЭМИС, Кафедра экономической математики, информатики и статистики**
Курс: **1**
Семестр: **1, 2**
Учебный план набора 2016 года

Распределение рабочего времени

№	Виды учебной деятельности	1 семестр	2 семестр	Всего	Единицы
1	Лекции	36	34	70	часов
2	Практические занятия	18		18	часов
3	Лабораторные занятия	36	34	70	часов
4	Контроль самостоятельной работы (курсовой проект / курсовая работа)		17	17	часов
5	Всего аудиторных занятий	90	85	175	часов
6	Самостоятельная работа	54	59	113	часов
7	Всего (без экзамена)	144	144	288	часов
8	Подготовка и сдача экзамена		36	36	часов
9	Общая трудоемкость	144	180	324	часов
		4.0	5.0	9.0	З.Е

Зачет: 1 семестр
Экзамен: 2 семестр
Курсовое проектирование / Курсовая работа: 2 семестр

Согласована на портале № 4660

Рассмотрена и одобрена на заседании кафедры
протокол № 2 от «15» 10 2016 г.

Томск 2016

Согласована на портале № 4660

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. ЭМИС _____ Зариковская Н. В.

Заведующий обеспечивающей каф. ЭМИС _____ Боровской И. Г.

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

Декан ФВС _____ Козлова Л. А.

Заведующий выпускающей каф. ЭМИС _____ Боровской И. Г.

Эксперты:

доцент кафедра ЭМИС _____ Шельмина Е. А.

Согласована на портале № 4660

1. Цели и задачи дисциплины

1.1. Цели дисциплины

Формирование у студентов навыка разработки программ на языке С++

1.2. Задачи дисциплины

– Формирование у студентов системы знаний по структурному программированию, основным методам построения и анализа алгоритмов, средствам разработки, отладки и тестирования программ;

– Расширение и углубление у студентов знаний по программированию средствами языка программирования С++;

– Формирование у студентов опыта применения различных методов решения задач, реализуемых на языке С++;

– Развитие алгоритмического и логического стилей мышления, внимания, трудолюбия, настойчивости и терпения;

– Совершенствовать умение классифицировать и систематизировать учебный материал, сравнивать, анализировать и правильно применять на практике;

– Учиться работать, в группе, самостоятельно, дистанционно, вести дискуссию, корректно формулировать вопросы.

2. Место дисциплины в структуре ОПОП

Дисциплина «Программирование» (Б1. Дисциплины (модули)) относится к блоку 1 (вариативная часть).

Предшествующими дисциплинами, формирующими начальные знания, являются следующие дисциплины: Алгоритмы и структуры данных, Информатика.

Последующими дисциплинами являются: Математическая логика и теория алгоритмов, Моделирование систем, Объектно-ориентированное программирование, Технология программирования.

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

– ОПК-1 Способность инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем.;

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

– ОПК-5 Способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.;

В результате изучения дисциплины студент должен:

– **знать** отличия структурного и объектно-ориентированного программирования; понятия объект, свойства объекта, событие; назначение и использование классов; понятия метод, наследование, дружественность; понятие пространства имен; назначение и использование перегрузки операторов; механизм возникновения исключений и методы их обработки; понятие структуры языка С++; понятие массива структур С++; принцип создания указателей на структуры. понятие библиотеки; отличия статических и динамических библиотек; способы связывания для динамических библиотек; функции для работы с динамической памятью в С++.

– **уметь** использовать пространство имен std для ввода и вывода данных; создавать обработчики событий; распределять память динамически; разрабатывать собственные динамические библиотеки и выполнять связывание несколькими способами; создавать обработчики исключений.

– **владеть** отношениями дружественности и наследования между классами; перегрузкой операций; способностью использовать возможности образовательной среды разработки; способностью создания собственных классов и структур на языке С++.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 9.0 зачетных единицы и представлена в

Согласована на портале № 4660

таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры	
		1 семестр	2 семестр
Аудиторные занятия (всего)	175	90	85
Лекции	70	36	34
Практические занятия	18	18	
Лабораторные занятия	70	36	34
Контроль самостоятельной работы (курсовой проект / курсовая работа)	17		17
Самостоятельная работа (всего)	113	54	59
Оформление отчетов по лабораторным работам	80	34	46
Проработка лекционного материала	24	11	13
Подготовка к практическим занятиям, семинарам	9	9	
Всего (без экзамена)	288	144	144
Подготовка и сдача экзамена	36		36
Общая трудоемкость час	324	144	180
Зачетные Единицы Трудоемкости	9.0	4.0	5.0

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Разделы дисциплины и виды занятий приведены в таблице 5.1.

Таблица 5.1 – Разделы дисциплины и виды занятий

№	Названия разделов дисциплины	Лекции	Практические занятия	Лабораторные работы	Самостоятельная работа	Курсовая работа	Всего часов (без экзамена)	Формируемые компетенции
1	Общая характеристика и элементы языка С++	2	2	0	2	0	6	ОПК-2, ОПК-5
2	Операторы ветвления	4	2	0	2	0	8	ОПК-2, ОПК-5
3	Циклы и массивы	4	2	6	6	0	18	ОПК-2, ОПК-5
4	Файлы	4	0	4	5	0	13	ОПК-2, ОПК-5
5	Функции	4	0	4	5	0	13	ОПК-2, ОПК-5
6	Указатели и динамическая память	4	0	6	8	0	18	ОПК-2, ОПК-5

Согласована на портале № 4660

7	Структуры и связные списки	4	4	6	9	0	23	ОПК-2, ОПК-5
8	Рекурсия в языке С++	6	4	4	7	0	21	ОПК-2, ОПК-5
9	Классы	4	4	6	10	0	24	ОПК-2, ОПК-5
10	Конструкторы	4	0	6	12	0	22	ОПК-2, ОПК-5
11	Дружественные функции и классы	4	0	6	12	0	22	ОПК-2, ОПК-5
12	Наследование	6	0	8	12	0	26	ОПК-2, ОПК-5
13	Обработка исключительных ситуаций	4	0	0	2	0	6	ОПК-2, ОПК-5
14	Базы данных	6	0	8	10	0	24	ОПК-2, ОПК-5
15	Консольные приложения	6	0	6	9	0	21	ОПК-2, ОПК-5
16	Создание установочного диска	4	0	0	2	0	6	ОПК-1, ОПК-2, ОПК-5
	Итого	70	18	70	113	17	288	

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
1 семестр			
1 Общая характеристика и элементы языка С++	История и назначение языка С++;Алфавит, идентификаторы, литералы, служебные слова языка С++;Пример программы на С++	2	ОПК-2, ОПК-5
	Итого	2	
2 Операторы ветвления	Логические выражения;Условный оператор if;Оператор выбора switch.	4	ОПК-2, ОПК-5
	Итого	4	
3 Циклы и массивы	Цикл;Оператор цикла с параметром (for);Оператор цикла с предусловием (while);Оператор цикла с постусловием (do while);Одномерные массивы;Способы инициализации массива;Вывод массива;Обработка элементов массива.	4	ОПК-2, ОПК-5

Согласована на портале № 4660

	Итого	4	
4 Файлы	Операторы работы с файлами	4	ОПК-2, ОПК-5
	Итого	4	
5 Функции	Определение и вызов функций;Локальные и глобальные переменные.	4	ОПК-2, ОПК-5
	Итого	4	
6 Указатели и динамическая память	Указатели и адреса;Указатели как аргументы функций;Указатели и массивы;Массивы указателей, указатели на указатели;Динамическое распределение памяти.	4	ОПК-2, ОПК-5
	Итого	4	
7 Структуры и связные списки	Основные сведения о структурах;Структуры и функции;Массивы структур;Указатели на структуры. Структуры со ссылками на себя.	4	ОПК-2, ОПК-5
	Итого	4	
8 Рекурсия в языке C++	Рекурсия;Использование рекурсивных функций.	6	ОПК-2, ОПК-5
	Итого	6	
9 Классы	Абстракции и классы;Реализация функций-членов класса;Использование классов.	4	ОПК-2, ОПК-5
	Итого	4	
Итого за семестр		36	
2 семестр			
10 Конструкторы	Конструкторы и деструкторы параметры по умолчанию:Перегрузка конструкторов и деструкторов	4	ОПК-2, ОПК-5
	Итого	4	
11 Дружественные функции и классы	Дружественные классы;Дружественные функции;Вложенные классы.	4	ОПК-2, ОПК-5
	Итого	4	
12 Наследование	Наследование классов	6	ОПК-2, ОПК-5
	Итого	6	
13 Обработка исключительных ситуаций	Исключения;Механизм исключений;Способы повышения отказоустойчивости программы;Блок try;Раскручивание стека.	4	ОПК-2, ОПК-5
	Итого	4	
14 Базы данных	Назначение баз данных;Пример построения базы данных на C++.	6	ОПК-2, ОПК-5

Согласована на портале № 4660

	Итого	6	
15 Консольные приложения	Создание консольного приложения	6	ОПК-2, ОПК-5
	Итого	6	
16 Создание установочного диска	Создание установочного диска средствами C++	4	ОПК-1, ОПК-2, ОПК-5
	Итого	4	
Итого за семестр		34	
Итого		70	

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	Алгоритмы и структуры данных	+						+									
2	Информатика	+	+														
Последующие дисциплины																	
1	Математическая логика и теория алгоритмов		+	+		+		+	+								
2	Моделирование систем				+	+	+	+	+							+	+
3	Объектно-ориентированное программирование				+		+			+	+	+	+	+			
4	Технология программирования														+	+	+

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 № 4660

Компетенции	Лекции	Практические занятия	Лабораторные занятия	Контроль самостоятельной работы (курсовой проект / курсовая работа)	Самостоятельная работа	
ОПК-1	+			+		Защита отчета, Опрос на занятиях, Отчет по курсовой работе, Дифференцированный зачет
ОПК-2	+	+	+	+	+	Домашнее задание, Отчет по индивидуальному заданию, Защита отчета, Отчет по лабораторной работе, Опрос на занятиях, Тест, Отчет по курсовой работе, Дифференцированный зачет

Согласована на портале № 4660

ОПК-5	+	+	+	+	+	Домашнее задание, Отчет по индивидуальному заданию, Защита отчета, Отчет по лабораторной работе, Опрос на занятиях, Тест, Отчет по курсовой работе, Дифференцированный зачет

6. Интерактивные методы и формы организации обучения

Не предусмотрено РУП

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
1 семестр			
3 Циклы и массивы	Цикл; Оператор цикла с параметром (for); Оператор цикла с предусловием (while); Оператор цикла с постусловием (do while); Одномерные массивы; Способы инициализации массива; Вывод массива; Обработка элементов массива.	6	ОПК-2, ОПК-5
	Итого	6	
4 Файлы	Операторы работы с файлами	4	ОПК-2, ОПК-5
	Итого	4	
5 Функции	Определение и вызов функций;Локальные и глобальные переменные	4	ОПК-2, ОПК-5
	Итого	4	
6 Указатели и динамическая память	Указатели и адреса; Указатели как аргументы функций; Указатели и массивы; Массивы указателей, указатели на указатели; Динамическое распределение памяти.	6	ОПК-2, ОПК-5
	Итого	6	
7 Структуры и связные списки	Основные сведения о структурах; Структуры и функции; Массивы структур; Указатели на структуры. Структуры со ссылками на себя.	6	ОПК-2, ОПК-5
	Итого	6	

Согласована на портале № 4660

8 Рекурсия в языке С++	Рекурсия; Использование рекурсивных функций.	4	ОПК-2, ОПК-5
	Итого	4	
9 Классы	Абстракции и классы; Реализация функций-членов класса; Использование классов.	6	ОПК-2, ОПК-5
	Итого	6	
Итого за семестр		36	
2 семестр			
10 Конструкторы	Конструкторы и деструкторы параметры по умолчанию: Перегрузка конструкторов и деструкторов	6	ОПК-2, ОПК-5
	Итого	6	
11 Дружественные функции и классы	Дружественные классы; Дружественные функции; Вложенные классы.	6	ОПК-2, ОПК-5
	Итого	6	
12 Наследование	Наследование классов	8	ОПК-2, ОПК-5
	Итого	8	
14 Базы данных	Назначение баз данных; Пример построения базы данных на С++.	8	ОПК-2, ОПК-5
	Итого	8	
15 Консольные приложения	Создание консольного приложения	6	ОПК-2, ОПК-5
	Итого	6	
Итого за семестр		34	
Итого		70	

8. Практические занятия

Содержание практических работ приведено в таблице 8.1.

Таблица 8. 1 – Содержание практических работ

Названия разделов	Содержание практических занятий	Трудоемкость, ч	Формируемые компетенции
1 семестр			
1 Общая характеристика и элементы языка С++	История и назначение языка С++; Алфавит, идентификаторы, литералы, служебные слова языка С++; Пример программы на С++	2	ОПК-2, ОПК-5
	Итого	2	
2 Операторы ветвления	Логические выражения; Условный оператор if; Оператор выбора switch.	2	ОПК-2, ОПК-5
	Итого	2	

Согласована на портале № 4660

3 Циклы и массивы	Цикл; Оператор цикла с параметром (for); Оператор цикла с предусловием (while); Оператор цикла с постусловием (do while); Одномерные массивы; Способы инициализации массива; Вывод массива; Обработка элементов массива.	2	ОПК-2, ОПК-5
	Итого	2	
7 Структуры и связные списки	Основные сведения о структурах; Структуры и функции; Массивы структур; Указатели на структуры. Структуры со ссылками на себя.	4	ОПК-2, ОПК-5
	Итого	4	
8 Рекурсия в языке С++	Рекурсия;Использование рекурсивных функций.	4	ОПК-2, ОПК-5
	Итого	4	
9 Классы	Абстракции и классы; Реализация функций-членов класса; Использование классов.	4	ОПК-2, ОПК-5
	Итого	4	
Итого за семестр		18	
Итого		18	

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
1 семестр				
1 Общая характеристика и элементы языка С++	Подготовка к практическим занятиям, семинарам	1	ОПК-2, ОПК-5	Домашнее задание, Опрос на занятиях, Отчет по индивидуальному заданию
	Проработка лекционного материала	1		
	Итого	2		
2 Операторы ветвления	Подготовка к практическим занятиям, семинарам	1	ОПК-2, ОПК-5	Домашнее задание, Опрос на занятиях, Отчет по индивидуальному заданию, Тест
	Проработка лекционного материала	1		
	Итого	2		
3 Циклы и массивы	Подготовка к практическим занятиям,	1	ОПК-2, ОПК-5	Домашнее задание, Опрос на занятиях,

Согласована на портале № 4660

	семинарам			Отчет по индивидуальному заданию, Отчет по лабораторной работе, Тест
	Проработка лекционного материала	1		
	Оформление отчетов по лабораторным работам	4		
	Итого	6		
4 Файлы	Проработка лекционного материала	1	ОПК-2, ОПК-5	Опрос на занятиях, Отчет по лабораторной работе
	Оформление отчетов по лабораторным работам	4		
	Итого	5		
5 Функции	Проработка лекционного материала	1	ОПК-2, ОПК-5	Опрос на занятиях, Отчет по лабораторной работе
	Оформление отчетов по лабораторным работам	4		
	Итого	5		
6 Указатели и динамическая память	Проработка лекционного материала	2	ОПК-2, ОПК-5	Опрос на занятиях, Отчет по лабораторной работе
	Оформление отчетов по лабораторным работам	6		
	Итого	8		
7 Структуры и связные списки	Подготовка к практическим занятиям, семинарам	2	ОПК-2, ОПК-5	Домашнее задание, Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе, Тест
	Проработка лекционного материала	1		
	Оформление отчетов по лабораторным работам	6		
	Итого	9		
8 Рекурсия в языке С++	Подготовка к практическим занятиям, семинарам	2	ОПК-2, ОПК-5	Домашнее задание, Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе
	Проработка лекционного материала	1		
	Оформление отчетов по лабораторным работам	4		
	Итого	7		
9 Классы	Подготовка к практическим занятиям, семинарам	2	ОПК-2, ОПК-5	Домашнее задание, Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе
	Проработка лекционного материала	2		
	Оформление отчетов по лабораторным работам	6		

Согласована на портале № 4660

	Итого	10		
Итого за семестр		54		
2 семестр				
10 Конструкторы	Проработка лекционного материала	2	ОПК-2, ОПК-5	Опрос на занятиях, Отчет по лабораторной работе
	Оформление отчетов по лабораторным работам	10		
	Итого	12		
11 Дружественные функции и классы	Проработка лекционного материала	2	ОПК-2, ОПК-5	Опрос на занятиях, Отчет по лабораторной работе
	Оформление отчетов по лабораторным работам	10		
	Итого	12		
12 Наследование	Проработка лекционного материала	2	ОПК-2, ОПК-5	Опрос на занятиях, Отчет по лабораторной работе
	Оформление отчетов по лабораторным работам	10		
	Итого	12		
13 Обработка исключительных ситуаций	Проработка лекционного материала	2	ОПК-2, ОПК-5	Опрос на занятиях
	Итого	2		
14 Базы данных	Проработка лекционного материала	2	ОПК-2, ОПК-5	Опрос на занятиях, Отчет по лабораторной работе
	Оформление отчетов по лабораторным работам	8		
	Итого	10		
15 Консольные приложения	Проработка лекционного материала	1	ОПК-2, ОПК-5	Опрос на занятиях, Отчет по лабораторной работе
	Оформление отчетов по лабораторным работам	8		
	Итого	9		
16 Создание установочного диска	Проработка лекционного материала	2	ОПК-2, ОПК-5	Опрос на занятиях
	Итого	2		
Итого за семестр		59		
	Подготовка к экзамену	36		Экзамен
Итого		149		

9.1. Тематика практики

1. Цикл;
2. Оператор цикла с параметром (for);
3. Оператор цикла с предусловием (while);
4. Оператор цикла с постусловием (do while);
5. Одномерные массивы;
6. Способы инициализации массива;

Согласована на портале № 4660

7. Вывод массива;
8. Обработка элементов массива.
9. Основные сведения о структурах;
10. Структуры и функции;
11. Массивы структур;
12. Указатели на структуры. Структуры со ссылками на себя.
13. Логические выражения;
14. Условный оператор if;
15. Оператор выбора switch.
16. Рекурсия;
17. Использование рекурсивных функций.
18. История и назначение языка С++;
19. Алфавит, идентификаторы, литералы, служебные слова языка С++;
20. Пример программы на С++

9.2. Вопросы на проработку лекционного материала

21. Определение и вызов функций;
22. Локальные и глобальные переменные.
23. Операторы работы с файлами
24. Создание консольного приложения
25. Абстракции и классы;
26. Реализация функций-членов класса;
27. Использование классов.
28. Назначение баз данных;
29. Пример построения базы данных на С++.
30. Указатели и адреса;
31. Указатели как аргументы функций;
32. Указатели и массивы;
33. Массивы указателей, указатели на указатели;
34. Динамическое распределение памяти.
35. Наследование классов

10. Курсовая работа

Содержание курсовой работы (проекта), трудоемкость и формируемые компетенции представлены в таблице 10.1.

Таблица 10. 1 – Содержание курсовой работы (проекта), трудоемкость и формируемые компетенции

Содержание курсовой работы	Трудоемкость (час.)	Формируемые компетенции
2 семестр		
Основные задачи и цели курсового проектирования:- приобретение дополнительных навыков и изучение программирования, в частности методов программирования сложных задач.В курсовой работе должна быть разработана тема в соответствии с заданием, одобренным кафедрой.ПЗ к работе должна содержать следующие разделы:1) титульный лист;2) реферат;3) задание на проектирование;4) содержание;5) введение;6) основная часть работы;7) заключение;8) список литературы;9) приложения.	17	ОПК-1, ОПК-2, ОПК-5
Итого за семестр	17	

Согласована на портале № 4660

10.1 Темы курсовых работ

Примерная тематика курсовых работ (проектов):

- Логическая игра «Обратный тетрис»
- Логическая игра «Тетрис»
- Задача Прима-Краскала («жадный» алгоритм)
- Логическая игра «Цепь»
- Логическая игра «Крестики-нолики» на неограниченном поле
- Построитель графиков функций
- Советник по транспорту
- Логическая игра «Сокобан»
- Эмулятор Total Commander
- Одномерные клеточные автоматы
- Оптимальные стратегии для игры с угадыванием («Быки и коровы»)
- Функция Аккермана
- Вычислитель арифметических выражений
- «Ханойская башня»
- Игра в слова
- «Задача о раскрое»
- Семь лунок
- Прыгающие шарики
- Касса аэрофлота
- Электронная сваха
- Форматирование текста
- Перенос слов
- Шифр Цезаря
- База данных «классный журнал»
- База данных «Дни рождения»
- База данных «поручения»
- Тестирующая система
- «Тир слов»
- «100 спичек»
- «Ним»
- «Черный квадрат»
- Круги на воде
- «Крестики-нолики»
- Склад
- Поздравления
- Счеты
- Отсев
- Тест по национальным сказкам
- Генератор любовного романа
- Игра «Нажми Клавишу»

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

Согласована на портале № 4660

1 семестр				
Домашнее задание	10	8	8	26
Опрос на занятиях	8	4	4	16
Отчет по индивидуальному заданию	8	6	6	20
Отчет по лабораторной работе	8	6	6	20
Тест	6	6	6	18
Итого максимум за период	40	30	30	100
Нарастающим итогом	40	70	100	100
2 семестр				
Опрос на занятиях	8	4	4	16
Отчет по лабораторной работе	24	12	12	48
Тест	6			6
Итого максимум за период	38	16	16	70
Экзамен				30
Нарастающим итогом	38	54	70	100

11.2. Пересчет баллов в оценки за контрольные точки

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

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

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

11.3. Пересчет суммы баллов в традиционную и международную оценку

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

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

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

Согласована на портале № 4660

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

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основная литература

1. Павловская, Татьяна Александровна. C/C++. Программирование на языке высокого уровня : учебник для вузов / Т. А. Павловская. - СПб. : Питер, 2007. - 464 с. : ил., портр., табл. - (Учебник для вузов) (300 лучших учебников для высшей школы). - Алф. указ.: с. 450-460. - ISBN 978-5-94723-568-5 (наличие в библиотеке ТУСУР - 47 экз.)

12.2. Дополнительная литература

1. Павловская, Татьяна Александровна. C++. Объектно-ориентированное программирование. Практикум : учебное пособие для вузов / Т. А. Павловская, Ю. А. Щупак. - СПб. : Питер, 2006. - 264[8] с. : ил., табл. - (300 лучших учебников для высшей школы в честь 300-летия Санкт-Петербурга) (Учебное пособие). - Библиогр.: с. 260. - Алф. указ.: с. 261-264. - ISBN 5-94723-842-X (наличие в библиотеке ТУСУР - 1 экз.)
2. Павловская, Татьяна Александровна. C/C++. Структурное программирование. Практикум : Учебное пособие для вузов / Т. А. Павловская, Ю. А. Щупак. - СПб. : Питер, 2005. - 238[2] с. : ил. - (300 лучших учебников для высшей школы в честь 300-летия Санкт-Петербурга) (Учебное пособие). - Алф. указ.: с. 236-238. - ISBN 5-94723-967-1 (наличие в библиотеке ТУСУР - 15 экз.)

12.3. Учебно-методическое пособие и программное обеспечение

1. Объектно-ориентированное программирование на C++: Учебное пособие / Егоров И. М. - 2007. 180 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/870, свободный.
2. Объектно-ориентированное программирование на C++: Руководство к организации самостоятельной работы / Егоров И. М. - 2007. 47 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/875, свободный.
3. 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» / Шельмина Е. А. - 2015. 21 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/6149, свободный.
4. Объектно-ориентированное программирование: методические указания к самостоятельной работе по выполнению лабораторных и практических работ для студентов 09.03.02 «Информационные системы и технологии»: Учебное-методическое пособие / Шельмина Е. А. - 2015. 6 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/6216, свободный.
5. Программирование: Методические указания к выполнению курсового проекта для обучающихся по направлению подготовки бакалавров 09.03.02 «Информационные системы и технологии» и 09.03.01 «Информатика и вычислительная техника» / Зариковская Н. В. - 2016. 21 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/6320, свободный.

12.4. Базы данных, информационно справочные и поисковые системы

1. https://cpp.zeef.com/michael.tkach
2. http://www.quizful.net/test
3. https://code-live.ru/tag/cpp-manual/
4. https://www.youtube.com/watch?v=x5MB9RfJFlU&ab_channel=NextPro

13. Материально-техническое обеспечение дисциплины

Требования к аудиториям (помещениям, местам) для проведения занятий:

- компьютерный класс (20 компьютеров);
- проектор (интерактивная доска) для демонстрации презентаций.

Требования к оборудованию рабочих мест преподавателя и обучающихся:

Видеопроектор, ноутбук, переносной экран. В компьютерном классе должны быть установлены операционные системы, Microsoft® Visual Studio® 2005 Express Edition, средства MS

Согласована на портале № 4660

Office; Word,PowerPoint.

14. Фонд оценочных средств

Фонд оценочных средств приведен в приложении 1.

15. Методические рекомендации по организации изучения дисциплины

Без рекомендаций.

Согласована на портале № 4660

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

Программирование

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки (специальность): **09.03.01 Информатика и вычислительная техника**
Направленность (профиль): **Автоматизированное управление бизнес-процессами и финансами**
Форма обучения: **очная**
Факультет: **ФВС, Факультет вычислительных систем**
Кафедра: **ЭМИС, Кафедра экономической математики, информатики и статистики**
Курс: **1**
Семестр: **1, 2**

Учебный план набора 2016 года

Разработчики:

- доцент каф. ЭМИС Зариковская Н. В.

Зачет: 1 семестр
Экзамен: 2 семестр
Курсовое проектирование / Курсовая работа: 2 семестр

Томск 2016

Согласована на портале № 4660

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-1	Способность инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем.	Должен знать отличия структурного и объектно-ориентированного программирования; понятия объект, свойства объекта, событие; назначение и использование классов; понятия метод, наследование, дружественность; понятие пространства имен; назначение и использование перегрузки операторов; механизм возникновения исключений и методы их обработки; понятие структуры языка C++; понятие массива структур C++; принцип создания указателей на структуры. понятие библиотеки; отличия статических и динамических библиотек; способы связывания для динамических библиотек; функции для работы с динамической памятью в C++.; Должен уметь использовать пространство имен std для ввода и вывода данных; создавать обработчики событий; распределять память динамически; разрабатывать собственные динамические библиотеки и выполнять связывание несколькими способами; создавать обработчики исключений.; Должен владеть отношениями дружественности и наследования между классами; перегрузкой операций; способностью использовать возможности образовательной среды разработки; способностью создания собственных классов и структур на языке C++.;
ОПК-2	Способность осваивать методики использования программных средств для решения практических задач.	
ОПК-5	Способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	

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	Обладает диапазоном практических умений,	Контролирует работу, проводит оценку,

Согласована на портале № 4660

	знаниями в пределах изучаемой области с пониманием границ применимости	требуемых для развития творческих решений, абстрагирования проблем	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

2 Реализация компетенций

2.1 Компетенция ОПК-1

ОПК-1: Способность инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем..

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

Таблица 3 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	принципы организации, установки, конфигурирования системных программных средств, сетевого ПО	выбирать и квалифицированно применять современные аппаратные и программные средства для реализации информационных и автоматизированных систем	системными инструментальными и сервисными средствами, встроенными средствами управления в ОС Microsoft Windows и Unix
Виды занятий	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену; • Контроль самостоятельной работы (курсовой проект / курсовая работа);	• Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену; • Контроль самостоятельной работы (курсовой проект / курсовая работа);	• Лабораторные занятия; • Самостоятельная работа; • Контроль самостоятельной работы (курсовой проект / курсовая работа);
Используемые средства оценивания	• Опрос на занятиях; • Отчет по курсовой работе; • Дифференцированный зачет;	• Опрос на занятиях; • Отчет по курсовой работе; • Дифференцированный зачет;	• Отчет по курсовой работе; • Дифференцированный зачет; • Зачет;

Согласована на портале № 4660

	• Зачет; • Экзамен; • Курсовое проектирование / Курсовая работа;	• Зачет; • Экзамен; • Курсовое проектирование / Курсовая работа;	• Экзамен; • Курсовое проектирование / Курсовая работа;

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)	• принципы организации, установки, конфигурирования системных программных средств, сетевого ПО ; • принципы и методы организации системных программных средств. Информационные технологии в системах управления;	• выбирать и квалифицированно применять современные аппаратные и программные средства для реализации информационных и автоматизированных систем ; • использовать технологии виртуализации, операционные системы, программное обеспечение вычислительных сетей;	• системными инструментальными и сервисными средствами, встроенными средствами управления в ОС Microsoft Windows и Unix ; • профессиональными умениями инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем;
Хорошо (базовый уровень)	• не в полной мере принципы организации, установки, конфигурирования системных программных средств, сетевого ПО ; • не в полной мере принципы и методы организации системных программных средств. Информационные технологии в системах управления;	• не в полной мере выбирать и квалифицированно применять современные аппаратные и программные средства для реализации информационных и автоматизированных систем ; • не в полной мере использовать технологии виртуализации, операционные системы, программное обеспечение вычислительных сетей;	• не в полной мере системными инструментальными и сервисными средствами, встроенными средствами управления в ОС Microsoft Windows и Unix ; • не в полной мере профессиональными умениями инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем;
Удовлетворительно (пороговый уровень)	• иметь представление об основных принципах и методах организации системных программных средств. Информационные технологии в системах	• иметь начальные навыки по использованию технологии виртуализации, операционные системы, программное	• общим представлением об инсталлировании программного и аппаратного обеспечения для информационных и

Согласована на портале № 4660

	управления;	обеспечение вычислительных сетей;	автоматизированных систем;

2.2 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 5.

Таблица 5 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	современные объектно-ориентированные алгоритмические языки, их области применения и особенности основные приемы сборочного программирования методики оценки качества объектно-ориентированного проектирования программных систем основные проектные процедуры и технологии при объектно-ориентированом анализе и реализации программных систем ознакомление с дополнительной литературой и передовыми научно-техническими достижениями в области программирования	использовать объектно-ориентированные методы и средства разработки алгоритмов и программ, способы отладки, испытания и документирования программ использовать современные готовые библиотеки классов использовать основные модели, методы и средства информационных технологий и способы их применения для решения задач в предметных областях использовать современные системные программные средства, технологии и инструментальные средства	навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ навыками объектно-ориентированной разработки программного обеспечения навыками адаптации существующих проектных решений к разрабатываемой или модернизируемой программной системе навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ навыками использования инструментальных средств оценки качества объектных моделей программных систем
Виды занятий	• Практические занятия; • Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену; • Контроль самостоятельной работы (курсовой	• Практические занятия; • Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену; • Контроль самостоятельной работы (курсовой	• Лабораторные занятия; • Самостоятельная работа; • Контроль самостоятельной работы (курсовой проект / курсовая работа);

Согласована на портале № 4660

	проект / курсовая работа);	проект / курсовая работа);	
Используемые средства оценивания	• Домашнее задание; • Отчет по лабораторной работе; • Отчет по индивидуальному заданию; • Опрос на занятиях; • Тест; • Отчет по курсовой работе; • Дифференцированный зачет; • Зачет; • Экзамен; • Курсовое проектирование / Курсовая работа;	• Домашнее задание; • Отчет по лабораторной работе; • Отчет по индивидуальному заданию; • Опрос на занятиях; • Тест; • Отчет по курсовой работе; • Дифференцированный зачет; • Зачет; • Экзамен; • Курсовое проектирование / Курсовая работа;	• Домашнее задание; • Отчет по лабораторной работе; • Отчет по индивидуальному заданию; • Отчет по курсовой работе; • Дифференцированный зачет; • Зачет; • Экзамен; • Курсовое проектирование / Курсовая работа;

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

Таблица 6 – Показатели и критерии оценивания компетенции на этапах

Состав	Знать	Уметь	Владеть
Отлично (высокий уровень)	• современные объектно-ориентированные алгоритмические языки, их области применения и особенности; • основные приемы сборочного программирования; • методики оценки качества объектно-ориентированного проектирования программных систем; • основные проектные процедуры и технологии при объектно-ориентированом анализе и реализации программных систем; • ознакомление с дополнительной литературой и передовыми научно-техническими достижениями в области	• использовать объектно-ориентированные методы и средства разработки алгоритмов и программ, способы отладки, испытания и документирования программ; • использовать современные готовые библиотеки классов; • использовать основные модели, методы и средства информационных технологий и способы их применения для решения задач в предметных областях; • использовать современные системные программные средства, технологии и инструментальные средства;	• навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ; • навыками объектно-ориентированной разработки программного обеспечения; • навыками адаптации существующих проектных решений к разрабатываемой или модернизируемой программной системе; • навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ; • навыками использования инструментальных

Согласована на портале № 4660

	программирования;		средств оценки качества объектных моделей программных систем;
Хорошо (базовый уровень)	• методики оценки качества объектно-ориентированного проектирования программных систем; • основные проектные процедуры и технологии при объектно-ориентированом анализе и реализации программных систем; • наличие не искажающих существа ответа погрешностей и пробелов при изложении материала;	• использовать объектно-ориентированные методы и средства разработки алгоритмов и программ, способы отладки, испытания и документирования программ; • выделять и повторно использовать проектные решения и программный код в объектной парадигме; • использовать современные готовые библиотеки классов; • осуществлять объектно-ориентированный анализ предметной области и объектное проектирование структуры программной системы;	• навыками восприятия документации и исходного кода объектно-ориентированных программных систем; • навыками объектно-ориентированной разработки программного обеспечения; • навыками адаптации существующих проектных решений к разрабатываемой или модернизируемой программной системе; • навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ;
Удовлетворительно (пороговый уровень)	• основные проектные процедуры и технологии при объектно-ориентированом анализе и реализации программных систем; • основные теоретические и практические вопросы программного материала;	• использовать объектно-ориентированные методы и средства разработки алгоритмов и программ, способы отладки, испытания и документирования программ;	• навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ; • навыками объектно-ориентированной разработки программного обеспечения;

2.3 Компетенция ОПК-5

ОПК-5: Способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности..

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

Таблица 7 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	ознакомление с дополнительной	творчески подтвердить теоретические	глубоким усвоением программного материала

Согласована на портале № 4660

	литературой и передовыми научно-техническими достижениями в области программирования современные объектно-ориентированные алгоритмические языки, их области применения и особенности основные приемы сборочного программирования методики оценки качества объектно-ориентированного проектирования программных систем основные проектные процедуры и технологии при объектно-ориентированом анализе и реализации программных систем	положения процессов и расчета аппаратов соответствующими примерами, умелое применение теоретических знаний при решении практических задач читать программы, написанные на объектно-ориентированных языках программирования осуществлять объектно-ориентированный анализ предметной области и объектное проектирование структуры программной системы; использовать основные модели, методы и средства информационных технологий и способы их применения для решения задач в предметных областях использовать современные системные программные средства, технологии и инструментальные средства рассчитывать метрики качества объектно-ориентированного анализа и проектирования программных систем выделять и повторно использовать проектные решения и программный код в объектной парадигме использовать современные готовые библиотеки классов использовать объектно-ориентированные методы и средства разработки алгоритмов и программ, способы отладки, испытания и документирования программ	по всем разделам курса, изложение его на высоком научно-техническом уровне навыками восприятия документации и исходного кода объектно-ориентированных программных систем навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ навыками объектно-ориентированной разработки программного обеспечения навыками использования инструментальных средств оценки качества объектных моделей программных систем навыками адаптации существующих проектных решений к разрабатываемой или модернизируемой программной системе навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ
Виды занятий	• Практические	• Практические	• Лабораторные

Согласована на портале № 4660

	занятия; • Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену; • Контроль самостоятельной работы (курсовой проект / курсовая работа);	занятия; • Лабораторные занятия; • Лекции; • Самостоятельная работа; • Подготовка к экзамену; • Контроль самостоятельной работы (курсовой проект / курсовая работа);	занятия; • Самостоятельная работа; • Контроль самостоятельной работы (курсовой проект / курсовая работа);
Используемые средства оценивания	• Домашнее задание; • Отчет по лабораторной работе; • Отчет по индивидуальному заданию; • Опрос на занятиях; • Тест; • Отчет по курсовой работе; • Дифференцированный зачет; • Зачет; • Экзамен; • Курсовое проектирование / Курсовая работа;	• Домашнее задание; • Отчет по лабораторной работе; • Отчет по индивидуальному заданию; • Опрос на занятиях; • Тест; • Отчет по курсовой работе; • Дифференцированный зачет; • Зачет; • Экзамен; • Курсовое проектирование / Курсовая работа;	• Домашнее задание; • Отчет по лабораторной работе; • Отчет по индивидуальному заданию; • Отчет по курсовой работе; • Дифференцированный зачет; • Зачет; • Экзамен; • Курсовое проектирование / Курсовая работа;

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

Таблица 8 – Показатели и критерии оценивания компетенции на этапах

Состав	Знать	Уметь	Владеть
Отлично (высокий уровень)	• современные объектно-ориентированные алгоритмические языки, их области применения и особенности; • основные приемы сборочного программирования; • методики оцнки качества объектно-ориентированного проектирования программных систем; • основные проектные процедуры и технологии при	• осуществлять объектно-ориентированный анализ предметной области и объектное проектирование структуры программной системы; • творчески подтвердить теоретические положения процессов и расчета аппаратов соответствующими примерами, умелое применение теоретических знаний	• глубоким усвоением программного материала по всем разделам курса, изложение его на высоком научно-техническом уровне; • навыками восприятия документации и исходного кода объектно-ориентированных программных систем; • навыками применения объектно-ориентированных подходов и

Согласована на портале № 4660

	объектно-ориентированном анализе и реализации программных систем; • ознакомление с дополнительной литературой и передовыми научно-техническими достижениями в области программирования;	при решении практических задач; • читать программы, написанные на объектно-ориентированных языках программирования; • использовать объектно-ориентированные методы и средства разработки алгоритмов и программ, способы отладки, испытания и документирования программ; • выделять и повторно использовать проектные решения и программный код в объектной парадигме; • использовать современные готовые библиотеки классов; • использовать основные модели, методы и средства информационных технологий и способы их применения для решения задач в предметных областях; • использовать современные системные программные средства, технологии и инструментальные средства; • рассчитывать метрики качества объектно-ориентированного анализа и проектирования программных систем;	инструментальных средств разработки алгоритмов и программ; • навыками объектно-ориентированной разработки программного обеспечения; • навыками адаптации существующих проектных решений к разрабатываемой или модернизируемой программной системе; • навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ; • навыками использования инструментальных средств оценки качества объектных моделей программных систем;
Хорошо (базовый уровень)	• методики оценки качества объектно-ориентированного проектирования программных систем; • основные проектные	• осуществлять объектно-ориентированный анализ предметной области и объектное проектирование	• полным усвоением программного материала в объеме обязательной литературы по курсу; • терминологией и

Согласована на портале № 4660

	процедуры и технологии при объектно-ориентированом анализе и реализации программных систем; • наличие не искажающих существа ответа погрешностей и пробелов при изложении материала;	структуры программной системы; • увязывать теоретические знания с решением практических задач; • читать программы, написанные на объектно-ориентированных языках программирования; • использовать объектно-ориентированные методы и средства разработки алгоритмов и программ, способы отладки, испытания и документирования программ; • выделять и повторно использовать проектные решения и программный код в объектной парадигме; • использовать современные готовые библиотеки классов;	символикой изучаемой дисциплины при изложении материала; • навыками восприятия документации и исходного кода объектно-ориентированных программных систем; • навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ; • навыками объектно-ориентированной разработки программного обеспечения; • навыками адаптации существующих проектных решений к разрабатываемой или модернизируемой программной системе; • навыками применения объектно-ориентированных подходов и инструментальных средств разработки алгоритмов и программ;
Удовлетворительно (пороговый уровень)	• основные проектные процедуры и технологии при объектно-ориентированом анализе и реализации программных систем; • основные теоретические и практические вопросы программного материала;	• осуществлять объектно-ориентированный анализ предметной области и объектное проектирование структуры программной системы; • читать программы, написанные на объектно-ориентированных языках программирования; • использовать объектно-ориентированные методы и средства разработки алгоритмов	• допущение незначительных ошибок и неточностей, нарушение логической последовательности изложения материала, недостаточную аргументацию теоретических положений; • навыками восприятия документации и исходного кода объектно-ориентированных программных систем; • навыками применения объектно-ориентированных

Согласована на портале № 4660

		и программ, способы отладки, испытания и документирования программ;	подходов и инструментальных средств разработки алгоритмов и программ; • навыками объектно-ориентированной разработки программного обеспечения;

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 Тестовые задания

– 1. Какому зарезервированному слову программа передаёт управление в случае, если значение переменной или выражения оператора switch не совпадает ни с одним константным выражением? o other o contingency o all o default

– 2. Какие среды программирования (IDE) предназначены для разработки программных средств? o MVS, Code::Blocks, Qt Creator, AutoCAD, Eclipse o MVS, NetBeans, Qt Creator, RAD Studio, Dev-C++ o MVS, Code::Blocks, Qt Creator, RAD Studio, MathCAD

– 3. Какой из перечисленных типов данных не является типом данных в C++? o double o int o float o real

– 4. Каков результат работы следующего фрагмента кода? int x = 0; switch(x) { case 1: cout << "Один"; case 0: cout << "Нуль"; case 2: cout << "Привет мир"; } o Привет мир o Нуль o НульПривет мир o Один

– 5. Какая из следующих записей - правильный комментарий в C++? o /* комментарий */ o ** Комментарий ** o */ Комментарии */ o {комментарий}

– 6. Программа, переводящая входную программу на исходном языке в эквивалентную ей выходную программу на результирующем языке, называется: o сканер o интерпретатор o компилятор o транслятор

– 7. Укажите объектно-ориентированный язык программирования o C++ o Java o Все варианты ответов o Eiffel

– 8. Какое значение, по соглашению, возвращает программа операционной системе в случае успешного завершения? o -1 o Программа не возвращает значение. o 1 o 0

– 9. Что будет напечатано? int main() { for (int i = 0; i < 4; ++i) { switch (i) { case 0 : std::cout << "0"; case 1 : std::cout << "1"; continue; case 2 : std::cout << "2"; break; default : std::cout << "D"; break; } std::cout << "."; } return 0; } o 011.2.D o 0112.D. o 0.1.2. o Ошибка компиляции в строке 10 o 01.2.D.

– 10. Какой служебный знак ставится после оператора case ? o - o : o ; o .

– 11. В приведённом коде измените или добавьте один символ чтобы код напечатал 20 звёздочек - *. int i, N = 20; for(i = 0; i < N; i--) printf("*"); o int i, N = 40; for(i = 0; i < N; i--) printf("*"); o int i, N = 20; for(i = 0; i < N; N--) printf("*"); o int i, N = 20; for(i = 19; i < N; i--) printf("*"); o int i, N = 20; for(i = 20; i < N; i--) printf("*");

– 12. Какую функцию должны содержать все программы на C++, которые используют стандартную библиотеку C? o system() o program() o start() o main()

– 13. Язык программирования C++ разработал o Бьерн Страуструп o Никлаус Вирт o Дональд Кнут o Кен Томпсон

– 14. Цикл с постусловием? o do while o for o while

– 15. Общий формат оператора множественного выбора - switch o switch (switch_expression) { case constant1: statement1; [break;] case constant2: statement2; [break;] case constantN: statementN; [break;] [default: statement N+l;] } o switch (switch_expression) { case constant1, case constant2:

Согласована на портале № 4660

statement1; [break;] case constantN: statementN; [break;] [default: statement N+l;] } o switch (switch_expression) { case constant1: statement1; [break;] case constant2: statement2; [break;] case constantN: statementN; [break;] [else: statement N+l;] }

– 16. Какой оператор не допускает перехода к следующему варианту в конструкции switch? o точка с запятой o end; o Stop; o break;

– 17. Чему будет равна переменная a, после выполнения этого кода int a; for(a = 0; a < 10; a++) {}? o 9 o 1 o 10

– 18. Какой из ниже перечисленных операторов, не является циклом в C++? o while o do while o repeat until o for

– 19. Чтобы подключить заголовочный файл в программу на C++, например, iostream необходимо написать: o include #iostream,h; o include (iostreamh) o #include <> c iostream внутри скобок o #include <>; c iostream.h внутри скобок

– 20. Простые типы данных в C++. o целые – int, вещественные – float или real, символьные – char o целые – int, вещественные – float или double, символьные – char o целые – int, вещественные – float или double, символьные – string o целые – bool, вещественные – float или double, символьные – string

– 21. Какие служебные символы используются для обозначения начала и конца блока кода? o begin end o ( ) o < > o { }

– 22. Название C++ предложил o Кэн Томпсон o Бьёрн Страуструп o Рик Масситти o Дональд Кнут

– 23. Цикл с предусловием? o do while o for o while

– 24. Какой из следующих операторов - оператор сравнения двух переменных? o = o equal o := o ==

– 25. Тело любого цикла выполняется до тех пор, пока его условие ... o ложно o у цикла нет условия o истинно

– 26. До каких пор будут выполняться операторы в теле цикла while (x < 100)? o Пока x меньше или равен стам o Пока x строго меньше ста o Пока x больше ста o Пока x равен стам

– 27. Укажите правильное определение функции main в соответствии со спецификацией стандарта ANSI o void main(void) o int main(void) o int main() o void main()

– 28. Какими знаками заканчивается большинство строк кода в Си++? o , (запятая) o ; (точка с запятой) o . (точка) o : (двоеточие)

– 29. Выберите правильный вариант объявления константной переменной в C++, где type - тип данных в C++ variable- имя переменной value - константное значение o const variable = value; o const type variable := value; o const type variable = value;

– 30. Укажите правильную форму записи цикла do while o // форма записи оператора цикла do while: do // начало цикла do while { /*блок операторов*/; } while {/*условие выполнения цикла*/} // конец цикла do while o // форма записи оператора цикла do while: do // начало цикла do while { /*блок операторов*/; } while (/*условие выполнения цикла*/); // конец цикла do while o // форма записи оператора цикла do while: do // начало цикла do while { /*блок операторов*/; } while (/*условие выполнения цикла*/)// конец цикла do while

3.2 Темы домашних заданий

– Создание установочного диска средствами C++
– Исключения
– Механизм исключений
– Способы повышения отказоустойчивости программы
– Блок try
– Дружественные классы
– Дружественные функции
– Вложенные классы.
– Конструкторы и деструкторы параметры по умолчанию:
– Перегрузка конструкторов и деструкторов
– Основные сведения о структурах

Согласована на портале № 4660

– Структуры и функции
– Массивы структур
– Указатели на структуры. Структуры со ссылками на себя.
– Логические выражения
– Условный оператор if
– Оператор выбора switch.
– Использование рекурсивных функций.
– Оператор цикла с параметром (for)
– Оператор цикла с предусловием (while)
– Оператор цикла с постусловием (do while)
– Одномерные массивы
– Способы инициализации массива
– Обработка элементов массива.
– История и назначение языка C++

3.3 Темы индивидуальных заданий

– Создание установочного диска средствами C++
– Исключения
– Механизм исключений
– Способы повышения отказоустойчивости программы
– Блок try
– Дружественные классы
– Дружественные функции
– Вложенные классы.
– Конструкторы и деструкторы параметры по умолчанию:
– Перегрузка конструкторов и деструкторов
– Основные сведения о структурах
– Структуры и функции
– Массивы структур
– Указатели на структуры. Структуры со ссылками на себя.
– Логические выражения
– Условный оператор if
– Оператор выбора switch.
– Использование рекурсивных функций.
– Оператор цикла с параметром (for)
– Оператор цикла с предусловием (while)
– Оператор цикла с постусловием (do while)
– Одномерные массивы
– Способы инициализации массива
– Обработка элементов массива.
– История и назначение языка C++

3.4 Темы опросов на занятиях

– Абстракции и классы
– Реализация функций-членов класса
– Использование классов.
– Указатели и адреса
– Указатели как аргументы функций
– Указатели и массивы
– Массивы указателей, указатели на указатели
– Динамическое распределение памяти.
– Создание консольного приложения

Согласована на портале № 4660

– Назначение баз данных
– Пример построения базы данных на C++.
– Наследование классов
– Определение и вызов функций
– Локальные и глобальные переменные
– Операторы работы с файлами

3.5 Экзаменационные вопросы

– 1. Классификация языков программирования (компилируемые, исполняемые на виртуальных машинах и интерпретируемые ЯП). Стандарт языка C++. Стандартная библиотека.
– 2. История появления и развития ЯП C++.
– 3. Структура простейшей программы на C++. Процесс компиляции.
– 4. Базовые типы C++. Константы и переменные.
– 5. Алфавит языка. Операции, выражения, операторы и идентификаторы.
– 6. Оператор присваивания. Арифметические операторы. Операторы сравнения и логические операторы. Порядок действий (приоритет операторов).
– 7. Условный оператор и оператор множественного выбора.
– 8. Операторы для организации циклов. Префиксная и постфиксная итерация.
– 9. Стандартные потоки ввода/вывода. Средства работы с потоками ввода/вывода. Специальные символы (символ перевода строки, символ табуляции, символ конца строки).
– 10. Псевдослучайные числа. Генерация псевдослучайных чисел на C++.
– 11. Массивы в C++. Алгоритмы сортировки. C-строки (символьные массивы). Многомерные массивы.
– 12. Указатели в C++. Операция разыменования. Константные указатели и указатели на константы. Ссылки в C++.
– 13. Указатели и массивы (одномерные и многомерные). Операции над указателями.
– 14. Статическая, автоматическая и динамическая память. Типичные ошибки, возникающие при использовании динамической памяти («утечка памяти»).
– 15. Функции в C++. Прототип и описание функции. Формальные и фактические параметры. Передача параметров в функции по значению, по ссылке, по указателю.
– 16. Параметры функций со значениями по умолчанию. Перегрузка функций. Рекурсия. Шаблоны функций.
– 17. Графика в C++.
– 18. Объектно-ориентированный подход. Абстракция данных. Классы.
– 19. Контроль доступа к элементам класса. Функции-члены классов. Конструкторы и деструкторы. Инкапсуляция.
– 20. ООП в C++. Наследование.
– 21. ООП в C++. Полиморфизм и шаблоны классов

3.6 Вопросы дифференцированного зачета

– Защита курсового проекта

3.7 Темы лабораторных работ

– Цикл
– Оператор цикла с параметром (for)
– Оператор цикла с предусловием (while)
– Оператор цикла с постусловием (do while)
– Одномерные массивы
– Способы инициализации массива
– Вывод массива
– Обработка элементов массива.
– Операторы работы с файлами
– Определение и вызов функций

Согласована на портале № 4660

– Локальные и глобальные переменные
– Указатели и адреса
– Указатели как аргументы функций
– Указатели и массивы
– Массивы указателей, указатели на указатели
– Динамическое распределение памяти.
– Основные сведения о структурах
– Структуры и функции
– Массивы структур
– Указатели на структуры. Структуры со ссылками на себя.
– Рекурсия
– Использование рекурсивных функций.
– Абстракции и классы
– Реализация функций-членов класса
– Использование классов
– Конструкторы и деструкторы параметры по умолчанию: Перегрузка конструкторов и деструкторов
– Дружественные классы
– Дружественные функции
– Вложенные классы.
– Наследование классов
– Назначение баз данных
– Пример построения базы данных на C++
– Создание консольного приложения

3.8 Зачёт

– 1. Элементы языка Си.
– 2. Константы, идентификаторы, ключевые слова.
– 3. Типы данных и их объявление. Целые и вещественные типы. Перечисляемый тип.
– 4. Типы данных и их объявление. Указатели.
– 5. Выражения. Операнды и операции. Правила преобразования типов.
– 6. Операторы языка Си. Оператор выражение, составной оператор, операторы условного перехода.
– 7. Организация циклических вычислительных процессов с помощью операторов for, while, do while.
– 8. Организация ввода-вывода в языке Си. Форматный ввод-вывод.
– 9. Массивы. Индексные выражения. Хранение в памяти одномерных и многомерных массивов.
– 10. Массивы. Основные алгоритмы их обработки. Ввод-вывод, поиск экстремума, сортировка.
– 11. Правила определения переменных и типов. Инициализация данных.
– 12. Определение и вызов функций. Фактические и формальные параметры.
– 13. Определение и вызов функций. Передача массивов и указателей на функции.
– 14. Определение и вызов функций. Предварительная инициализация параметров, функции с переменным числом параметров. Передача параметров функции main.
– 15. Время жизни и область видимости программных объектов. Классы памяти.
– 16. Инициализация глобальных и локальных переменных
– 17. Динамические массивы. Особенности выделения и освобождения памяти для многомерных массивов.

3.9 Темы курсовых проектов (работ)

– Логическая игра «Обратный тетрис»

Согласована на портале № 4660

– Логическая игра «Тетрис»
– Задача Прима-Краскала («жадный» алгоритм)
– Логическая игра «Цепь»
– Логическая игра «Крестики-нолики» на неограниченном поле
– Построитель графиков функций
– Советник по транспорту
– Логическая игра «Сокобан»
– Эмулятор Total Commander
– Одномерные клеточные автоматы
– Оптимальные стратегии для игры с угадыванием («Быки и коровы»)
– Функция Аккермана
– Вычислитель арифметических выражений
– «Ханойская башня»
– Игра в слова
– «Задача о раскрое»
– Семь лунок
– Прыгающие шарики
– Касса аэрофлота
– Электронная сваха
– Форматирование текста
– Перенос слов
– Шифр Цезаря
– База данных «классный журнал»
– База данных «Дни рождения»
– База данных «поручения»
– Тестирующая система
– «Тир слов»
– «100 спичек»
– «Ним»
– «Черный квадрат»
– Круги на воде
– «Крестики-нолики»
– Склад
– Поздравления
– Счеты
– Отсев
– Тест по национальным сказкам
– Генератор любовного романа
– Игра «Нажми Клавишу»

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, Татьяна Александровна. C/C++. Программирование на языке высокого уровня : учебник для вузов / Т. А. Павловская. - СПб. : Питер, 2007. - 464 с. : ил., портр., табл. - (Учебник для вузов) (300 лучших учебников для высшей школы). - Алф. указ.: с. 450-460. - ISBN 978-5-94723-568-5 (наличие в библиотеке ТУСУР - 47 экз.)

Согласована на портале № 4660

4.2. Дополнительная литература

1. Павловская, Татьяна Александровна. C++. Объектно-ориентированное программирование. Практикум : учебное пособие для вузов / Т. А. Павловская, Ю. А. Щупак. - СПб. : Питер, 2006. - 264[8] с. : ил., табл. - (300 лучших учебников для высшей школы в честь 300-летия Санкт-Петербурга) (Учебное пособие). - Библиогр.: с. 260. - Алф. указ.: с. 261-264. - ISBN 5-94723-842-X (наличие в библиотеке ТУСУР - 1 экз.)
2. Павловская, Татьяна Александровна. C/C++. Структурное программирование. Практикум : Учебное пособие для вузов / Т. А. Павловская, Ю. А. Щупак. - СПб. : Питер, 2005. - 238[2] с. : ил. - (300 лучших учебников для высшей школы в честь 300-летия Санкт-Петербурга) (Учебное пособие). - Алф. указ.: с. 236-238. - ISBN 5-94723-967-1 (наличие в библиотеке ТУСУР - 15 экз.)

4.3. Учебно-методическое пособие и программное обеспечение

1. Объектно-ориентированное программирование на C++: Учебное пособие / Егоров И. М. - 2007. 180 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/870, свободный.
2. Объектно-ориентированное программирование на C++: Руководство к организации самостоятельной работы / Егоров И. М. - 2007. 47 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/875, свободный.
3. 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» / Шельмина Е. А. - 2015. 21 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/6149, свободный.
4. Объектно-ориентированное программирование: методические указания к самостоятельной работе по выполнению лабораторных и практических работ для студентов 09.03.02 «Информационные системы и технологии»: Учебное-методическое пособие / Шельмина Е. А. - 2015. 6 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/6216, свободный.
5. Программирование: Методические указания к выполнению курсового проекта для обучающихся по направлению подготовки бакалавров 09.03.02 «Информационные системы и технологии» и 09.03.01 «Информатика и вычислительная техника» / Зариковская Н. В. - 2016. 21 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/publications/6320, свободный.

4.4. Базы данных, информационно справочные и поисковые системы

1. https://cpp.zeef.com/michael.tkach
2. http://www.quizful.net/test
3. https://code-live.ru/tag/cpp-manual/
4. https://www.youtube.com/watch?v=x5MB9RfJFlU&ab_channel=NextPro

Согласована на портале № 4660