МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
Томский государственный университет систем управления и радиоэлектроники

У Т В Е Р Ж Д А Ю

Проректор по учебной работе

_____ П.Е. Троян

«____» _____ 2017 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
ОСНОВЫ АЛГОРИТМИЗАЦИИ И ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Уровень основной образовательной программы: бакалавриат

Направление(я) подготовки (специальность): 09.03.03 – Прикладная информатика

Профиль: Прикладная информатика в экономике

Форма обучения: заочная

Факультет: ЗиВФ, заочный и вечерний

Кафедра: АСУ, Кафедра автоматизированных систем управления

Курс **1,2**

Семестр 1,2,3,4

Учебный план набора 2016 года набора и последующих лет

Распределение рабочего времени:

Виды учебной работы	Семестр 1	Семестр 2	Семестр 3	Семестр 4	Всего	Единицы
Лекции	4	2	6	–	12	часов
Практические занятия	2	4	4	2	12	часов
Лабораторные работы	4	12	12	4	32	часов
Курсовой проект/работа (КРС) (аудиторная)	–	–	–	–	–	–
Всего аудиторных занятий	10	18	22	6	56	часов
Из них в интерактивной форме	–	3	–	6	9	часов
Самостоятельная работа студентов (СРС)	–	76	102	118	296	часов
Всего (без экзамена)	10	94	124	124	352	часов
Самост. работа на подготовку и сдачу зачета	–	4	–	4	8	часов
Общая трудоемкость	10	98	124	128	360	часов
(в зачетных единицах)					10	ЗЕТ

Контрольная работа 2,4 семестры

Зачет 2 семестр

Диф. зачет 4 семестр

Томск 2017

Согласована на портале № 17434

Рассмотрена и одобрена на заседании кафедры
протокол № 1 от «12» 1 2017 г.

Рабочая программа по дисциплине составлена с учетом требований Федерального Государственного образовательного стандарта высшего профессионального образования (ФГОС ВО) по направлению подготовки 09.03.03 Прикладная информатика (квалификация (степень) "бакалавр"), утвержденного Приказом Министерства образования и науки Российской Федерации от 12 марта 2015 г. № 207, рассмотрена и утверждена на заседании кафедры 12 января 2017 г., протокол № 1.

Разработчик доцент каф. АСУ _____ М.В. Григорьева

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

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами специальности.

Декан ЗиВФ, к.физ.-мат.н., доцент _____ И.В.Осипов

Заведующий профилирующей и выпускающей
кафедрой АСУ, д.т.н., профессор _____ А.М. Кориков

Эксперт:
Кафедра АСУ, доцент _____ А.И. Исакова

Согласована на портале № 17434

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Дисциплина «Основы алгоритмизации и языки программирования» ставит своей **целью** изучение основ алгоритмизации и обучение студентов навыкам программирования для решения задач на современных ЭВМ. Изучение дисциплины направлено на освоение теоретических основ алгоритмизации **задач**, практических приемов программирования на алгоритмических языках высокого уровня, основ организации вычислительного процесса в ЭВМ, проектирование программ. При проведении практических и лабораторных занятий упор делается на интенсификацию обучения, выражающуюся в требовании написания законченных программ.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Учебная дисциплина «Основы алгоритмизации и языки программирования» относится к базовым дисциплинам блока «Дисциплины (модули)» учебного плана бакалавров направления 09.03.03 «Прикладная информатика» профиля «Прикладная информатика в экономике».

Для изучения дисциплины необходимы знания и навыки, полученные ими в рамках дисциплины «Информатика и программирование».

Освоение данной дисциплины позволяет использовать полученные в ней знания в последующих предметах, определяемым учебным планом, в частности: «Объектно-ориентированное программирование», «Базы данных», «Программная инженерия», «Информационные системы в бухгалтерском учете».

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины направлен на формирование следующих компетенций:

профессиональные компетенции (ПК):

– способностью разрабатывать, внедрять и адаптировать прикладное программное обеспечение (**ПК-2**);

– способностью программировать приложения и создавать программные прототипы решения прикладных задач (**ПК-8**).

В результате изучения дисциплины студент должен:

Знать: методы структурного программирования, понятие данных,

Уметь: разрабатывать и отлаживать эффективные алгоритмы и программы с использованием современных технологий программирования

Владеть*:* навыками программирования в современных средах.

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины составляет 6 зачетных единиц.

Вид учебной работы	Всего часов	Семестры			
		1	2	3	4
Аудиторные занятия (всего)	56	10	18	22	6
В том числе:					
Лекции (Л)	12	4	2	6	–
Практические занятия (ПЗ)	12	2	4	4	2
Лабораторные работы (ЛР)	32	4	12	12	4
Самостоятельная работа студента (СРС) (всего)	296	–	76	102	118
В том числе:					
Проработка лекционного материала		–	2	6	–
Подготовка к практическим занятиям		–	4	4	2
Подготовка к лабораторным занятиям		–	12	12	4
Самостоятельное изучение тем теоретической части		–	38	80	72
Контрольная работа	–	–	20	–	40
Подготовка к экзамену/зачету	8	–	4	–	4
Вид промежуточной аттестации (зачет, экзамен)			зачет		диф. зачет
Общая трудоемкость час	360	10	98	124	128
зач. ед. до сотых долей	10				

Согласована на портале № 17434

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

5.1. Разделы дисциплин и виды занятий

№ п/п	Наименование раздела дисциплины	Л	ПЗ	ЛР	СРС	Всего час.	Формируемые компетенции
	Семестр 1						
1.	Основы алгоритмизации	4	2	4	–	10	ПК-2, ПК-8
	Семестр 2						
2.	Рекуррентные алгоритмы	2	2	4	8	16	ПК-2, ПК-8
3.	Использование функций в алгоритмических языках	–	2	4	16	22	ПК-2, ПК-8
4.	Алгоритмы нахождения корней функции	–	–	–	18	18	ПК-2, ПК-8
5.	Работа с массивами в языке Си	–	–	4	34	38	ПК-2, ПК-8
	ИТОГО за семестр 2	**2**	**4**	**12**	**76**	**94**	
	Семестр 3						
6.	Алгоритмы информационного поиска и сортировки	2	2	4	18	26	ПК-2, ПК-8
7.	Проверка правильности алгоритмов	–	–	2	22	24	
8.	Рекурсивные алгоритмы	–	–	–	20	20	ПК-2, ПК-8
9.	Упорядочение нечисловых массивов	2	–	2	24	28	ПК-2, ПК-8
10.	Структурированные типы данных	2	2	4	18	26	ПК-2, ПК-8
	ИТОГО за семестр 3	**6**	**4**	**12**	**102**	**124**	
	Семестр 4						
11.	Динамические информационные структуры	–	–	–	56	56	ПК-2, ПК-8
12.	Интерфейс пользователь-компьютер и его составные части	–	2	4	62	68	ПК-2, ПК-8
	ИТОГО за семестр 4	**–**	**2**	**4**	**118**	**124**	
	ИТОГО по дисциплине	**12**	**12**	**32**	**296**	**352**	

5.2. Содержание разделов дисциплины (по лекциям)

№ п/п	Наименование разделов	Содержание разделов	Трудоемкость (час.)	ПК
		Семестр 1		
1.	Основы алгоритмизации	Понятие алгоритма. Способы записи алгоритмов. Понятие типа данных, приведение типов. Константы. Операции. Форматированный ввод-вывод. Препроцессорные средства. Операторы цикла и ветвления.	4	ПК-2, ПК-8
		Семестр 2		
2.	Рекуррентные алгоритмы	Понятие рекуррентной последовательности и рекуррентного алгоритма. Задача вычисления элемента последовательности с заданным номером. Задача вычисления суммы конечного числа элементов. Вычисление бесконечных сумм.	2	ПК-2, ПК-8
		Семестр 3		
6.	Алгоритмы поиска и сортировки	Общая постановка задачи сортировки. Простые методы сортировки массива: сортировка включением, сортировкам выбором, сортировка обменом.	2	ПК-2, ПК-8
9.	Упорядочение нечисловых массивов	Работа со строками в языке Си. Лексикографический принцип упорядочения символьных строк. Перекодировка символов.	2	
10.	Структурированные типы данных	Структурные типы и структуры в языке Си. Структуры, массивы и указатели. Файловая структура данных. Основные процедуры и функции работы с потоком. Особенности работы с текстовыми и стандартными текстовыми файлами.	2	
		ИТОГО	**12**	

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

№ п/п	Наименование обеспечивающих (предыдущих) дисциплин	№ № разделов данной дисциплины, для которых необходимо изучение обеспечивающих (предыдущих) дисциплин											
		1	2	3	4	5	6	7	8	9	10	11	12
1.	Информатика и программирование	+	+	+	+	+	+	+	+	+	+	+	+

№ п/п	Наименование последующих дисциплин	№ № разделов данной дисциплины, которые необходимы для изучения последующих дисциплин											
		1	2	3	4	5	6	7	8	9	10	11	12
1.	Объектно-ориентированное программирование		+	+			+				+	+	+
3.	Базы данных					+	+			+	+	+	+
4.	Информационные системы в БУ	+		+		+				+	+	+	+

Согласована на портале № 17434

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

Перечень компетенций	Виды занятий				Формы контроля
	Л	Пр	Лаб	СРС	
ПК-2	+	+	+	+	Устный опрос на лекции, Тест, домашнее задание, устная защита практической работы, письменный отчет по лабораторной работе
ПК-8	+	+	+	+	Тест, устная защита практической работы, письменный отчет по лабораторной работе

6. МЕТОДЫ И ФОРМЫ ОРГАНИЗАЦИИ ОБУЧЕНИЯ

Для успешного освоения дисциплины применяются различные образовательные технологии, которые обеспечивают достижение планируемых результатов обучения согласно основной образовательной программе, с учетом требований к объему занятий в интерактивной форме.

Технологии интерактивного обучения при разных формах занятий

Методы \ Формы	Лекции (час)	Практические/семинарские занятия (час)	Лабораторные (час)занятия	Всего (час)
IT-методы	2			**2**
Работа в команде			5	**5**
Поисковый метод		2		**2**
Итого интерактивных занятий	**2**	**2**	**5**	**9**

IT-методы: на экран выводится несложный алгоритм на тему лекции, от студентов требуется найти ошибку.

Работа в команде используется при совместном выполнении одной задачи во время выполнения лабораторных работ несколькими студентами.

Игра: при объяснении алгоритмов применяются игровые ситуации в качестве демонстрации, например, сортировки группы студентов по росту, обмена данными двух ячеек оперативной памяти демонстрируется на примере кочек на болоте.

Поисковый метод используется для поиска логических ошибок алгоритма.

7. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

№ п/п	№ раздела дисциплины	Наименование лабораторных работ	Трудоемкость(час.)	ОК, ПК
		Семестр 1		
1.	**1**	Основы алгоритмизации. Циклические алгоритмы	4	ПК-2, ПК-8
		Семестр 2		
2	**2**	Алгоритмы нахождения бесконечных сумм	4	ПК-2, ПК-8
3	**3**	Прототипы функций, формальные и фактические параметры	4	
4	**4**	Алгоритмы на одномерных массивах	4	
		Семестр 3		
5.	**6**	Алгоритмы сортировки массивов методами выбора и вставки	4	ПК-2, ПК-8
6.	**8, 9**	Алгоритмы обработки строковых данных в Си	4	
7.	**10**	Алгоритмы с использование структур на языке Си	4	
		Семестр 4		
8	**12**	Алгоритмы создания простейших пользовательских интерфейсов	4	ПК-2, ПК-8
		ИТОГО	**32**	

8. ПРАКТИЧЕСКИЕ ЗАНЯТИЯ (СЕМИНАРЫ)

№ п/п	№ раздела дисциплины	Наименование лабораторных работ	Трудоемкость(час.)	ОК, ПК
		Семестр 1		
1.	**1**	Линейные алгоритмы. Форматированный ввод/вывод	2	ПК-2, ПК-8
		Семестр 2		
2.	**2**	Циклические алгоритмы на примере вычисления членов рекуррентных последовательностей	2	ПК-2, ПК-8
3.	**3**	Стандартные функции Си. Создание простейших функций на языке Си, вызов функций.	2	
		Семестр 3		
4.	**6**	Алгоритм сортировки одномерных массивов методом обмена	2	ПК-2, ПК-8
5.	**10**	Указатели на структуры, структуры в качестве параметров	2	
		Семестр 4		
6.	**12**	Алгоритмы создания простейших пользовательских интерфейсов	2	ПК-2, ПК-8
		ИТОГО	**12**	

Согласована на портале № 17434

9. САМОСТОЯТЕЛЬНАЯ РАБОТА

№ п/п	№ раздела дисциплины из табл. 5.1	Тематика самостоятельной работы *(детализация)*	Трудоемкость в семестр (час.)			ОК, ПК	Контроль выполнения работы (Опрос, тест, дом.задание, и т.д)
			2	**3**	**4**		
1.	1 (1 сем), 2 (2сем), 6, 9,10 (3 сем)	Проработка лекционного материала	2	6	–	ПК-2, ПК-8	Опрос на занятиях (устно)
2.	1 (1 сем), 2,3 (2сем), 6, 10 (3 сем), 12 (4сем)	Подготовка к практическим занятиям	4	4	2		Устная защита практической работы
3.	1 (1 сем), 2,3,4 (2сем), 6, 8,9,10 (3 сем), 12 (4сем)	Подготовка к лабораторным занятиям	12	12	4		Отчет, защита лаб. работы
4.	3,4,5 (2сем), 6÷10 (3 сем), 11,12 (4сем)	Самостоятельное изучение тем теоретической части	38	80	72		Конспект
5.	5 (2сем), 12 (4сем	Выполнение контрольной работы	20	–	40		Устная защита контрольной работы
6.	1 ÷ 10	Подготовка и сдача экзамена	4	–	4		Оценка за экзамен
ИТОГО			**76**	**102**	**118**		

Темы для самостоятельного изучения

№ п/п	№ раздела дисциплины	Содержание тем для самостоятельного изучения	Трудоемкость (час.)
Семестр 2			
2.	3	Определение функции в языке Си. Формальные и фактические параметры, их разновидность. Классы памяти. Функции с переменным количеством параметров	10
3.	4	Принципы численного нахождения корня функции. Метод дихотомии. Метод _ассательных (Ньютона). Метод хорд (линейной интерполяции). Оценка трудоемкости алгоритма.	18
5.	5	Объявление массивов, их инициализация. Вложенные циклы при работе с массивами. Адреса и указатели.	10
		Итого за семестр	**38**
Семестр 3			
6.	7	Задача информационного поиска и ее разновидности. Поиск в неупорядоченном и упорядоченном массивах. Сложные сортировки.	10
		Основные источники ошибок и методы борьбы с ними. Тестирование. Методы верификации алгоритмов. Защита от неправильных данных.	20
7.	8	Понятие рекурсии, рекурсивного спуска, рекурсивного подъема. Примеры рекурсивных алгоритмов. Прямая и косвенная рекурсия. Преимущества и недостатки рекурсивного описания алгоритмов.	20
8.	9	Работа со строками в языке Си. Лексикографический принцип упорядочения символьных строк. Перекодировка символов. Особенности работы с текстовыми и стандартными текстовыми файлами.	20
9.	10	Структурные типы и структуры в языке Си. Структуры, массивы и указатели. Файловая структура данных. Основные процедуры и функции работы с потоком.	10
		Итого за семестр	**80**
Семестр 4			
10.	11	Статическое и динамическое представление данных. Списковые структуры данных. Односвязный список. Очередь, стек. Основные операции над ними.	36
11.	12	Понятие интерфейса. Типы диалога. Методы и приемы разработки интерфейса программы.	36
		Итого за семестр	**72**

Темы контрольных работ

В соответствии с учебным планом студентами ЗиВФ выполняется 3 контрольных работы по дисциплине: 1 контрольная работа во 2-ом семестре и две конрольных работы в 4-м семестре. Примеры вариантов контрольных работ приведены в приложении к рабочей программе.

Семестр 2

1) Работа с одномерными массивами в языке Си.
2) Работа с двумерными массивами в языке Си.

Семестр 4

3) Создание динамических структур данных типа стек, очередь, дек. Операции на ними.
4) Организация пользовательского интерфейса.

Согласована на портале № 17434

10. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ)

Курсовые работы в учебном плане отсутствуют.

11. БАЛЛЬНО-РЕЙТИНГОВАЯ СИСТЕМА

Не предусмотрена для заочной формы обучения

12. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

12.1 Основная литература

1. С/С++. Программирование на языке высокого уровня : учебник для вузов / Т. А. Павловская. - СПб. : ПИТЕР, 2013. - 461 с. (35 экз.)

2. Языки и методы программирования : учебник для вузов / И. Г. Головин, И. А. Волкова. - М. : Академия, 2012. - 304 с. (30 экз.)

12.2 Дополнительная литература

1. Давыдов, В.Г.Программирование и основы алгоритмизации : Учебное пособие для вузов / В. Г. Давыдов. - 2-е изд., стереотип. - М. : Высшая школа, 2005. - 448 с. (69 экз.)

2. Борисенко, В.В. Основы программирования / В. В. Борисенко. - М. : Интернет-Университет Информационных Технологий, 2005. - 314[4] (55 экз.)

3. Технологии программирования [Электронный ресурс] : учебное пособие / В. В. Кручинин ; Томский государственный университет систем управления и радиоэлектроники (Томск). - Электрон. текстовые дан. - Томск : [б. и.], 2013. – Режим доступа : http://edu.tusur.ru/training/publications/2834

12.3 Учебно-методические пособия

12.3.1. Обязательные учебно-методические пособия

По лабораторным работам

1. Программирование на языках высокого уровня: Методические указания по выполнению лабораторных работ и самостоятельной работе для студентов 230200 «Информационные системы» / Панасенко Е. А. – 2012. 12 с - 12 с. - Режим доступа: https://edu.tusur.ru/training/publications/2198

По практическим занятиям

2. Программирование и основы алгоритмизации [Электронный ресурс] : лабораторный практикум / Е. В. Истигечева, А. В. Мельников ; Томский государственный университет систем управления и радиоэлектроники - Томск : 2015. - 31. - Режим доступа: https://edu.tusur.ru/training/publications/5024

По самостоятельной работе

3. Программирование и основы алгоритмизации [Электронный ресурс] : методические указания по самостоятельной работе / Е. В. Истигечева, А. В. Мельников ; Томский государственный университет систем управления и радиоэлектроники - Томск : 2015. - 11 с. - Режим доступа: https://edu.tusur.ru/training/publications/5023

12.3.2 Учебно-методические пособия для лиц с ограниченными возможностями здоровья

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа инвалидов предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

Для лиц с нарушениями зрения:

– в форме электронного документа;

– в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

– в форме электронного документа;

– в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

– в форме электронного документа;

– в печатной форме.

12.4. Базы данных, информационно-справочные, поисковые системы и требуемое программное обеспечение

Информационно-справочные и поисковые системы сети Интернет.

Согласована на портале № 17434

13. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

13.1. Общие требования к материально-техническому обеспечению дисциплины

13.1.1. Материально-техническое обеспечение для лекционных занятий

Для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используется учебная аудитория, с количеством посадочных мест не менее 22-24, оборудованная доской и стандартной учебной мебелью. Имеются наглядные пособия в виде презентаций по лекционным разделам дисциплины.

13.1.2. Материально-техническое обеспечение для лабораторных работ

Для проведения лабораторных занятий используется учебно-исследовательская вычислительная лаборатория, расположенная по адресу 634034, Томская область, г. Томск, Вершинина улица, д. 74, 4 этаж, ауд. 437, 438, 439. Состав оборудования: Учебная мебель; Экран с электроприводом DRAPER BARONET – 1 шт.; Мультимедийный проектор TOSHIBA – 1 шт.; Компьютеры класса не ниже Intel Pentium G3220 (3.0GHz/4Mb)/4GB RAM/ 500GB с широкополосным доступом в Internet, с мониторами типа Samsung 18.5" S19C200N– 10 шт.; Используется лицензионное программное обеспечение, пакеты версией не ниже: Microsoft Windows XP Professional with SP3; Visual Studio 2008 EE with SP1; Microsoft Office Visio 2010; Microsoft SQL-Server 2005; Matlab v6.5.

13.1.3. Материально-техническое обеспечение для самостоятельной работы

Для самостоятельной работы используется учебная аудитория (компьютерный класс), расположенная по адресу 634034, г. Томск, ул. Вершинина, 74, 1 этаж, ауд. 100. Состав оборудования: учебная мебель; компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 4 шт.; компьютеры подключены к сети ИНТЕРНЕТ и обеспечивают доступ в электронную информационно-образовательную среду университета.

13.2. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения.

При обучении студентов **с нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах для студентов с нарушениями слуха, мобильной системы обучения для студентов с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой обучаются студенты с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При обучении студентов **с нарушениями зрениями** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для удаленного просмотра.

При обучении студентов **с нарушениями опорно-двигательного аппарата** используются альтернативные устройства ввода информации и другие технические средства приема/передачи учебной информации в доступных формах для студентов с нарушениями опорно-двигательного аппарата, мобильной системы обучения для людей с инвалидностью.

Согласована на портале № 17434

14. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

14.1. Основные требования к фонду оценочных средств и методические рекомендации

Фонд оценочных средств и типовые контрольные задания, используемые для оценки сформированности и освоения закрепленных за дисциплиной компетенций при проведении текущей, промежуточной аттестации по дисциплине приведен в приложении к рабочей программе.

14.2 Требования к фонду оценочных средств для лиц с ограниченными возможностями здоровья

Для студентов с инвалидностью предусмотрены дополнительные оценочные средства, перечень которых указан в таблице 14.1.

Таблица 14.1 – Дополнительные средства оценивания для студентов с инвалидностью

Категории студентов	Виды дополнительных оценочных средств	Формы контроля и оценки результатов обучения
С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно- двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами, исходя из состояния обучающегося на момент проверки

14.3 Методические рекомендации по оценочным средствам для лиц с ограниченными возможностями здоровья

Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной форме;
- в печатной форме с увеличенным шрифтом;
- в форме электронного документа;
- методом чтения ассистентом задания вслух;
- предоставление задания с использованием сурдоперевода.

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге;
- набор ответов на компьютере;
- набор ответов с использованием услуг ассистента;
- представление ответов устно.

Процедура оценивания результатов обучения инвалидов по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

- в форме электронного документа;
- в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

- в форме электронного документа;
- в печатной форме.

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 17434

ПРИЛОЖЕНИЕ к рабочей программе

Федеральное государственное бюджетное образовательное учреждение высшего образования

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

УТВЕРЖДАЮ

Проректор по учебной работе

_____ **П. Е. Троян**

«___» _____ 2017 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Уровень основной образовательной программы бакалавриат

Направление подготовки 09.03.03 – Прикладная информатика

Профиль(и) Прикладная информатика в экономике

Форма обучения заочная

Факультет систем управления

Кафедра автоматизированных систем управления

Курс 1,2

Семестр 1,2,3,4

Учебный план набора 2016г набора и последующих лет

Контрольные работы во 2 (1 штука), 4 (2 штуки) семестрах

Зачет 2 семестр

Диф. зачет 4 семестр

Томск 2017

Согласована на портале № 17434

1. 1. ВВЕДЕНИЕ

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины «**Основы алгоритмизации и языки программирования»** и представляет собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.) и методов их использования, предназначенных для измерения уровня достижения студентом установленных результатов обучения.

ФОС по дисциплине (практике) используется при проведении текущего контроля успеваемости и промежуточной аттестации студентов. Перечень закрепленных за дисциплиной «**Основы алгоритмизации и языки программирования»** компетенций приведен в таблице 1.

Таблица 1 – Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенции
ПК-2	Способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	***Знать:*** методы структурного программирования, понятие данных и их структуры; ***Уметь:*** разрабатывать и отлаживать эффективные алгоритмы и прикладное программное обеспечние с использованием современных технологий программирования; ***Владеть****:* навыками разработки и отладки программ в современных средах.
ПК-8	Способность программировать приложения и создавать программные прототипы решения прикладных задач	***Знать:*** методы структурного программирования программных приложений, понятие данных и их структуры, ***Уметь:*** программировать приложения; ***Владеть****:* навыками разработки программных приложений в современных средах.

2. РЕАЛИЗАЦИЯ КОМПЕТЕНЦИЙ

2.1 Компетенция ПК-2

ПК-2: способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение.

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

Таблица 2 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	Методы структурного программирования, понятие данных и их структуры, этапы разработки программного обеспечения.	Разрабатывать и отлаживать эффективные алгоритмы и прикла–дное программное обеспечние с использованием современных технологий программирования	Навыками разработки и отладки программного оеспечения в современных средах.
Виды занятий	Лекции, практические занятия, СРС	Практические занятия, лабораторные работы, СРС	Практические занятия, лабораторные работы, СРС
Используемые средства оценивания	– Контрольная работа; – Устный опрос; – Контроль выполнения домашнего задания; – Экзамен.	– Проверка правильности выполнения практических заданий; – Контрольная работа; – Отчеты по ЛР; – Экзамен	– Проверка правильности выполнения практических заданий; – Контрольная работа; – Отчеты по ЛР.

Согласована на портале № 17434

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

Таблица 3 – Общие характеристики показателей и критериев оценивания компетенции по этапам

Показатели и критерии	Знать	Уметь	Владеть
ОТЛИЧНО (высокий уровень)	Обладает фактическими и теоретическими знаниями в пределах изучаемой области с пониманием границ применимости	Обладает диапазоном практических умений, требуемых для развития творческих решений, абстрагирования проблем	Контролирует работу, проводит оценку, совершенствует действия работы
ХОРОШО (базовый уровень)	Знает факты, принципы, процессы, общие понятия в пределах изучаемой области	Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования	Берет ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем
УДОВЛЕТВОРИТЕЛЬНО (низкий уровень)	Обладает низким уровнем общих знаний	Обладает умениями на низком уровне, которые не достаточны для выполнения даже простых задач	Работает только при прямом наблюдении

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

Таблица 4 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	Знать	Уметь	Владеть
ОТЛИЧНО (высокий уровень)	Знает **все** методы структурного программирования, **все** типы данных и их структуры, **динамические и статические структуры данных,** этапы разработки программного обеспечения.	Умеет разрабатывать и отлаживать **эффективные** алгоритмы и прикладное программное обеспечние с использованием современных технологий программирования **любого уровня** сложности.	**Сводобно** владеет навыками разработки и отладки программного обеспечения в современных средах.
ХОРОШО (базовый уровень)	Знает все методы структурного программирования, **основные** типы данных и их структуры, этапы разработки программного обеспечения.	Умеет разрабатывать и отлаживать алгоритмы и прикладное программное обеспечние с использованием современных технологий программирования **среднего уровня** сложности.	Владеет навыками разработки **и отладки** программного обеспечения в современных средах.
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	Знает основные методы структурного программирования и типы данных, имеет понятие о структуре данных	Умеет разрабатывать **простые** алгоритмы и прикладное программное обеспечние с использованием современных технологий программирования.	Владеет навыками разработки несложного программного обеспечения в одной определенной среде.

Согласована на портале № 17434

2.2 Компетенция ПК-8

ПК-8: способность программировать приложения и создавать программные прототипы решения прикладных задач.

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

Таблица 5 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	Все типы данных, статические и динамические структуры данных	Программировать программные приложения	Навыками программирования программных приложений в современных средах
Виды занятий	Лекции, ПЗ, СРС	ПЗ, ЛР, СРС	ПЗ, ЛР, СРС
Используемые средства оценивания	– Контрольная работа; – Устный опрос; – Контроль выполнения домашнего задания; – Экзамен.	– Проверка правильности выполнения практических заданий; – Контрольная работа; – Отчеты по ЛР; – Экзамен	– Проверка правильности выполнения практических заданий; – Контрольная работа; – Отчеты по ЛР.

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

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

Таблица 6 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	Знать	Уметь	Владеть
ОТЛИЧНО (высокий уровень)	Знает **все** типы данных и их структуры, **все** методы структурного программирования	Умеет программировать приложения **любого уровня** сложности.	Навыками программирования приложений **любого уровня сложности** в современных средах
ХОРОШО (базовый уровень)	Знает **основные** типы данных и их структуры, **все** методы структурного программирования	Умеет программировать приложения **среднего уровня** сложности.	Навыками программирования приложений **среднего уровня сложности** в современных средах
УДОВЛЕТВО-РИТЕЛЬНО (низкий уровень)	Знает основные типы данных, имеет понятие о методах структурного программирования	Умеет программировать **простые** приложения.	Навыками программирования **простых** приложений в одной среде.

Согласована на портале № 17434

3. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ

Для реализации вышеперечисленных задач обучения используются следующие материалы: типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы, в составе, приведенном ниже.

3.1 Темы практических занятий

1-ый семестр

1. Линейные алгоритмы. Форматированный вво/вывод.

2-й семестр

2. Циклические алгоритмы на примере вычсиления членов рекуррентной последовательности.
3. Стандартные функции в Си. Создание простейших функций на языке Си, вызов функций.

3-й семестр

4. Простые алгоритмы сортировки и поиска в одномерном массиве.
5. Указатели на структуры, структуры в качетсве параметров.

Примеры практических заданий:

1. Напишите программу, выводящую ваше имя и адрес.
2. При условии, что переменная *value* имеет тип *int,* определите, какой будет получен в результате выполнения следующего цикла:

```
for ( value = 36; value > O; value /= 2)
  printf("%3d", value);
```

Какие проблемы могли бы возникнуть, если бы переменная *value* имела тип *double* вместо *int*?

3. Найдите все оошибки в следующей программе?

```
#include <stdio.h>
int main (void)
{
 char ch;
 int lc =0; /* счетчик строчных символов
 int lc = 0; /* счетчик прописных символов
 int lc = 0; /* счетчик других символов
 while ( (ch = getchar()) != '#')
 {
  if ( ' a ' <= ch >= ' z ' )
   lc + +;
   else if (! (ch < 'A' ) 11 !(ch> ' Z')
    uc++;
 oc++;
 }
 printf(%d строчных, %d прописных, %d других, lc, uc, oc);
 return ( );
}
```

4. На языке Си напишите алгоритм решения задачи. Ввести с клавиатуры n целых чисел a_1, a_2, .., a_n. Получить сумму тех чисел, которые удовлетворяют условию $|a_i|<i$. Задачу решить, используя тип-массив, для организации цикла применить оператор FOR.
   К каждой строке программы написать комментарий.

5. Параметры, передаваемые по ссылке, параметры, передаваемые по значению. Объясните подробно, чем отличаются их использование.
6. Дан массив целых чисел: 50 60 15 90 11 35 5. Отсортируйте массив методом простых вставок. Покажите состояние массива на каждом шаге. Объясните необходимость элемента-барьера.
7. Написать программу на язые Си, которая читает бинарный файл вещественных чисел, выводит на экран самый большой по значению компонент файла и его номер.
8. Описать структуру с именем STUDENT. Содержащую следующие поля:
   – Name – фамилия и инициалы;
   – Group – номер группы;
   – Ses – успеваемость (массив из пяти элементов).
   Написать программу, выполняющую следующие действия:
   – ввод данных из текстового файла в массив Students[10];
   – вывод на экран фамилий и номеров групп студентов, чей средний балл больше 4.0;
   – если таких студентов нет, вывести соответсвующее сообщение

Согласована на портале № 17434

3.2 Темы лабораторных работ

1-ый семестр

1. Разработка циклических алгоритмов на языке Си.

2-й семестр

2. Алгоритмы нахождения бесконечных сумм членов рекуррентных последовательностей.
3. Прототипы функций. Формальны и фактические параметры.
4. Алгоритмы на одномерных массивах.

3-й семестр

5. Алгоритмы сортировки методом выбора и вставки.
6. Разработка программы нам языке Си с использованием строковых данных.
7. Разработка программы на языке Си с использованием структур.

4-й семестр

8. Алгоритмы создания простейших пользовательскитх интерфейсов.

3.3 Самостоятельная работы

Темы для самостоятельного изучения

№ п/п	№ раздела дисциплины	Содержание тем для самостоятельного изучения	Трудоемкость (час.)
		Семестр 2	
2.	3	Определение функции в языке Си. Формальные и фактические параметры, их разновидность. Классы памяти. Функции с переменным количеством параметров	10
3.	4	Принципы численного нахождения корня функции. Метод дихотомии. Метод _ассательных (Ньютона). Метод хорд (линейной интерполяции). Оценка трудоемкости алгоритма.	18
5.	5	Объявление массивов, их инициализация. Вложенные циклы при работе с массивами. Адреса и указатели.	10
		Итого за семестр	**38**
		Семестр 3	
6.	7	Задача информационного поиска и ее разновидности. Поиск в неупорядоченном и упорядоченном массивах. Сложные сортировки.	10
		Основные источники ошибок и методы борьбы с ними. Тестирование. Методы верификации алгоритмов. Защита от неправильных данных.	20
7.	8	Понятие рекурсии, рекурсивного спуска, рекурсивного подъема. Примеры рекурсивных алгоритмов. Прямая и косвенная рекурсия. Преимущества и недостатки рекурсивного описания алгоритмов.	20
8.	9	Работа со строками в языке Си. Лексикографический принцип упорядочения символьных строк. Перекодировка символов. Особенности работы с текстовыми и стандартными текстовыми файлами.	20
9.	10	Структурные типы и структуры в языке Си. Структуры, массивы и указатели. Файловая структура данных. Основные процедуры и функции работы с потоком.	10
		Итого за семестр	**80**
		Семестр 4	
10.	11	Статическое и динамическое представление данных. Списковые структуры данных. Односвязный список. Очередь, стек. Основные операции над ними.	36
11.	12	Понятие интерфейса. Типы диалога. Методы и приемы разработки интерфейса программы.	36
		Итого за семестр	**72**

Согласована на портале № 17434

3.4 Темы контрольных работ

В соответствии с учебным планом студентами ЗиВФ выполняется 3 контрольных работы по дисциплине: 1 контрольная работа во 2-ом семестре и две конрольных работы в 3-м семестре. Примеры задач контрольных работ приведены ниже.

Семестр 2

Примеры задач по теме **«Основы алгоритмизации. Операторы цикла и ветвления»**

1. Даны натуральное число *n,* целые числа $a_1, ..., a_n$. Найти количество и сумму тех членов данной последовательности, которые делятся на 5 и не делятся на 7.

2. Ввести действительные числа *x, y* ($x \neq y$). Меньшее заменить их полусуммой, а большее – их удвоенным произведением. Вывести новые значения *x* и *y* на экран. Решение задачи повторить для нескольких вариантов *x, y*, для этого после каждого прогона задачи, спросит пользователя, повторить ли решение задачи.

3. Дано натуральное число *n*. Найти наибольшее среди чисел $ke^{\sin/k+1}$ *(k=l, ...,n),* а также сумму всех этих чисел. Решение задачи повторить для нескольких вариантов *x, y*, для этого после каждого прогона задачи, спросит пользователя, повторить ли решение задачи.

4. Если сумма трех различных действительных чисел *x, y, z* меньше единицы, то наименьшее из этих чисел заменить полусуммой двух других. В противном случае заменить меньшее из чисел *x* и *y* полусуммой двух оставшихся значений. Решение задачи повторить для нескольких вариантов *x, y*, *z* для этого после каждого прогона задачи, спросит пользователя, повторить ли решение задачи.

5. Дано натуральное число *n.* Выбросить из записи числа *n* цифры 0 и 5, оставив прежним порядок остальных цифр. Например, из числа 59015509 должно получиться 919.

6. Даны натуральное число *n*, целые числа $a_1,...,a_n$ (последовательности идущих подряд членов), состоящие из степеней пятерки. Определить для каждого числа степень пятерки.

Примеры задач по темам **«Рекуррентные алгоритмы»** и **«Использование функций в алгоритмических языках»**

1. Для введенного значения *x* вычислить значение функции:

$$f(x)=\begin{cases} x^2+7, x<0 \\ 0, x=0 \\ x^3+9, x>0 \end{cases}$$

2. Напечатать таблицу значений функции для n=5...8 с шагом 1, для a=-1...2 с шагом 1:

$$f(a,n)=\begin{cases} \sum_{i=1}^{n}\frac{a}{i}, |a|\le 1 \\ \sum_{i=1}^{n}\frac{a}{i^2}, |a|>1 \end{cases}$$

Вычисление *f(a,n)* оформить в виде функции.

3. Вычислить значение функции с точностью eps=0.00001 для *x* от 0 до 10 с шагом 1:

$$a^x=\sum_{k=0}^{\infty}\frac{(x\ln a)^k}{k!}$$

где *a* — задать с клавиатуры.

Вычисление a^x оформить функцией *ax( )* , используя рекуррентный алгоритм.

Вывысти на экран таблицу значений a^x, вычисленных с помошью стандартной функции *pow()* (прототип описан в заголовочном файле math.h) и с помощью *ax( ).* Дополнительное задание: вычислить число итерация для каждого *x.*

Примеры задач по теме **«Работа с массивами в языке Си» и «Алгоритмы информационного поиска и сортировки».**

Вариант 1

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1) сумму отрицательных элементов массива;

2) произведение элементов массива, расположенных между максимальным и минимальным элементами.

Упорядочить элементы массива по возрастанию.

Вариант 2

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1) сумму положительных элементов массива;

2) произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.

Упорядочить элементы массива по убыванию.

Вариант 3

В одномерном массиве, состоящем из n целых элементов, вычислить:

1) произведение элементов массива с четными номерами;

2) сумму элементов массива, расположенных между первым и последним нулевыми элементами.

Преобразовать массив таким образом, чтобы сначала располагались все положительные элементы, а потом — все отрицательные (элементы, равные 0, считать положительными).

Согласована на портале № 17434

Примеры задач по теме **«Работа со строковыми данными»**.

1. Дано натуральное число n и последовательность символов S1, …, Sn (строка). Группы символов, разделенные одним или несколькими пробелами, будем называть словами. Подсчитать количество букв а в последнем слове данной строки.

2. Дано натуральное число n и последовательность символов S1, …, Sn (строка). Найти количество слов в строке, начинающихся с буквы б (слова – группы символов, разделенные одним или несколькими пробелами).

3. Дано натуральное число n и последовательность символов S1, …, Sn (строка). Найти количество слов в строке, у которых первый и последний символы совпадают (слова – группы символов, разделенные одним или несколькими пробелами).

4. Дано натуральное число n и последовательность символов S1, …, Sn (строка). Найти длину самого короткого слова в строке (слова – группы символов, разделенные одним или несколькими пробелами).

5. Дан массив символьных строк (количество символов в строке равно m, количество строк равно n). Исключить из массива все строки, для которых есть равные среди строк с меньшими номерами.

Семестр 4

Примеры задач по темам **«Интерфейс пользователь-компьютер и его составные части»**, **«Структурированные типы данных».**

Создать пользовательский интерфейс для задачи, выполняющей следующие функции:

1. просмотр содержимого бинарного файла;
2. добавление информации в конец бинарного файла;
3. поиск информации в бинарном файле (в каждом варианте указано, что надо найти);
4. вывод информации из бинарного файла в текстовый файл.

Текстовый файл должен выглядеть, как отчет.

Бинарный файл создать из структур по варианту заданий.

Каждый пункт задания выполнить в отдельной функции. Использовать переключатель **switch** для выбора функций.

Пример содержимого текстового файла:

Список товаров в файле tovar.dat

Наименование товара	Ед. изм.	Цена товара, руб.	Количество товара
Помидоры	кг.	100,50	120
Огурцы	кг.	119,40	180

Итого 2 позиции

3.5 Вопросы и задачи для подготовки к теоретическому зачету (для студентов, которые не выполнили все контрольные работы и СРС)

Экзаменационный вопросы:

1 Структурное программирование, метод пошаговой детализации. Примеры.
2 Структурное программирование, его базовые конструкции. Примеры.
3 Структурное программирование, принцип сквозного контроля. Примеры.
4 Структура и компоненты программы на языке Си.
5 Функции в языке Си. Фактические и формальные параметры.
6 Функции распределения памяти. Динамические массивы
7 Препроцессорные средства.
8 Адреса и указатели. Операции над указателями в языке Си.
9 Потоковый Ввод-вывод в языке Си.
10 Массивы в языке Си. Объявление, инициализация, примеры алгоритмов.
11 Тип данных строка. Работа со строками в языке Си.
12 Структурные типы в языке Си. Примеры объявления и использования.
13 Задача сортировки массивов. Алгоритм простого выбора.
14 Задача сортировки массивов. Алгоритм простого обмена.
15 Задача сортировки массивов. Алгоритм простых вставок.
16 Открытие файла в потоке.
17 Стандартные файлы и функции для работы с ними.
18 Работа с файлами на диске.
19 Указатели в языке Си. Простейшие действия с ними
20 Алгоритмы сортировки файлов.
21 Улучшенные методы сортировки.

Согласована на портале № 17434

3.6 Вопросы и задачи для подготовки к дифференцированному зачету (для студентов, которые не выполнили все контрольные работы и СРС)

1. Дано натуральное число *n* и последовательность символов *S*1, …, *Sn* (строка). Группы символов, разделенные одним или несколькими пробелами, будем называть словами. Подсчитать количество букв *a* в последнем слове данной строки.
2. Для двух массивов a[5] и b[5], элементами которых являются вещественные числа, составить еще два массива x[5] и y[5] с элементами, содержащими соответственно разность и сумму элементов исходных массивов.
3. Используя рекуррентный алгоритм вычислить $a^x = \sum_{k=0}^{\infty} \frac{(x \ln a)^k}{k!}$,

   где a — задать с клавиатуры, x=0...10.
4. Задать с помощью счетчика случайных чисел элементы двух векторов *a(p)* и *b(p)*. Построить из них матрицу c(p×p) по правилу:

   *C[i][j] = a[i] / (*1 + *b[j]),* если *a[i]=b[j],*

   C[i][j] = b[i] / (1 + a[j]), если *a[i]=b[j].*
5. Ввести значения элементов двумерного массива (матрицы) с клавиатуры. Упорядочить столбцы матрицы по следующему закону: переставить столбцы матрицы так, чтобы элементы в первой строке матрицы были расположены по убыванию. Использовать алгоритм сотрировки простого выбора.

4 МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Для обеспечения процесса обучения и решения задач обучения используются следующие материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций, в составе:

1. Учебное пособие по дисциплине «Основы алгоритмизации и языки программирования» приведено в рабочей программе в разделе 12.1 [1].
2. Методические указания по лабораторным работам студентов всех форм обучения приведены в рабочей программе в разделе 12.3 [1].
3. Методические указания к практическим занятиям приведены в рабочей программе в разделе 12.3 [2].
4. Методические указания по самостоятельной работе студентов всех форм обучения приведены в рабочей программе в разделе 12.3 [3].

Основная литература:

– Кирнос, В.Н. Основы программирования на языке С++ : учебное пособие / В. Н. Кирнос ; Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники, Кафедра комплексной информационной безопасности электронно-вычислительных систем. - 2-е изд., перераб. и доп. - Томск : В-Спектр, 2007. - 129 с. (51 экз.)

– Касимов, В.З. Объектно-ориентированное программирование [Электронный ресурс] : методические рекомендации по выполнению лабораторных работ / В. З. Касимов ; Томский государственный университет систем управления и радиоэлектроники, Кафедра экономической математики, информатики и статистики. - Томск : 2011. - 28 с. - Режим доступа: http://edu.tusur.ru/training/publications/280

– Практикум по программированию на языке программирования Си / В. В. Кручинин ; Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники, Кафедра промышленной электроники. - Томск : ТУСУР, 2006. - 160 с. (45 экз.)

– Егоров, И.М. Объектно-ориентированное программирование на С++ : Руководство по организации самостоятельной работы студентов специальности 210106 / И. М. Егоров ; Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники, Кафедра промышленной электроники. - Томск : Томский государственный университет систем управления и радиоэлектроники, 2007. - 47 с. (190 экз.)

Дополнительная литература

– Программирование на языке Си : Учебное пособие для вузов / В. В. Подбельский, С. С. Фомин. - 2-е изд., доп. - М. : Финансы и статистика, 2007. - 600 с. (2 экз.)

– С++: для начинающих : самоучитель. Пер. с англ. / Г. Шилдт ; пер. К. Г. Финогенов. - М. : ЭКОМ, 2007. – 639 с. (1 экз.)

– Программирование на языке С : пер. с англ. / С. Кочан. - 3-е изд. - М. : Вильямс, 2007. - 489 с. (2экз.)

– Кент, Джефф. С++. Основы программирования : самоучитель / Д. Кент ; пер. Ю. В. Кирпичев. - М. : НТ Пресс, 2010. - 368 с. (1 экз.)

Согласована на портале № 17434