

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра теоретической и прикладной информатики

“УТВЕРЖДАЮ”
ДЕКАН ФПМИ

д.т.н. Тимофеев В. С.

“ ____ ” _____ г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Низкоуровневое программирование

Образовательная программа: 02.03.03 Математическое обеспечение и администрирование информационных систем, профиль: Математическое и программное обеспечение информационных технологий

Курс: 3, семестр : 5

Факультет прикладной математики и информатики,

		Семестр
№	Вид деятельности	5
1	Всего зачетных единиц (кредитов)	3
2	Всего часов	108
3	Всего занятий в контактной форме, час.	43
4	Лекции, час.	18
5	Практические занятия, час.	0
6	Лабораторные занятия, час.	18
7	из них в активной и интерактивной форме, час.	14
8	Аттестация, час.	2
9	Консультации, час.	5
10	Самостоятельная работа, час.	65
11	Виды самостоятельной работы (курсовой проект, курсовая работа, РГЗ, подготовка к контрольной работе)	РГЗ
12	Вид аттестации	ДЗ

Рабочая программа составлена на основании федерального государственного образовательного стандарта высшего образования по направлению (специальности): 02.03.03 Математическое обеспечение и администрирование информационных систем

ФГОС введен в действие приказом №222 от 12.03.2015 г. , дата утверждения: 07.04.2015 г.

Место дисциплины в структуре учебного плана: Блок 1, базовая

Рабочая программа разработана на основе компетентностной модели выпускника по направлению (специальности): 02.03.03 Математическое обеспечение и администрирование информационных систем

Рабочая программа обсуждена на заседании кафедры

ТПИ, протокол заседания кафедры №4 от 20.06.2017

Утверждена на совете факультета прикладной математики и информатики, протокол № 6 от 21.06.2017

Программу разработал:

профессор, д.т.н. Лисицин Д. В.

Заведующий кафедрой:

доцент, д.т.н. Чубич В. М.

Ответственный за образовательную программу:

заведующий кафедрой Чубич В. М.

1. Внешние требования

Таблица 1.1

Компетенция ФГОС: ОПК.8 способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения (далее - ПО; в части следующих результатов обучения:
з8. Знать основные способы разработки программ на ассемблере и работы в системе Visual C++
у6. Уметь использовать средства Visual C++ для разработки программ на ассемблере и двуязычных программ на C++ и ассемблере

2. Требования НГТУ к результатам освоения дисциплины

Таблица 2.1

Результаты изучения дисциплины по уровням освоения (иметь представление, знать, уметь, владеть)	Формы организации занятий
<i>Низкоуровневое программирование</i>	
ОПК.8.з8 Знать основные способы разработки программ на ассемблере и работы в системе Visual C++	
1.Знать команды, директивы и операции языка ассемблера	Лекции; Лабораторные работы; Самостоятельная работа
2.Знать основы взаимодействия языка ассемблера и языка C++	Лекции; Лабораторные работы; Самостоятельная работа
3.Знать средства операционной системы для ввода-вывода информации в консольных приложениях	Лабораторные работы; Самостоятельная работа
4.Знать основы работы на уровне макроязыка ассемблера	Лекции; Самостоятельная работа
5.Знать средства Visual C++ для разработки программ на языке ассемблера	Лабораторные работы; Самостоятельная работа
ОПК.8.у6 Уметь использовать средства Visual C++ для разработки программ на ассемблере и двуязычных программ на C++ и ассемблере	
6.Уметь пользоваться средствами Visual C++ для низкоуровневого программирования	Лабораторные работы; Самостоятельная работа
7.Уметь использовать при программировании команды, директивы и операции языка ассемблера	Лабораторные работы; Самостоятельная работа
8.Уметь пользоваться средствами операционной системы для ввода-вывода информации в консольных приложениях	Лабораторные работы; Самостоятельная работа
9.Уметь организовать интерфейс между модулями на языке ассемблера и языке C++	Лабораторные работы; Самостоятельная работа

3. Содержание и структура учебной дисциплины

Таблица 3.1

Темы лекций	Активные формы, час.	Часы	Ссылки на результаты обучения
Семестр: 5			
Дидактическая единица: Основные элементы языка ассемблера			
1. Язык ассемблера, задачи, решаемые с его помощью. Регистры процессора. Процесс разработки программы, структура программы. Понятия команд, директив и операций. Формат записи команд. Директивы segment, proc, assume, end, align, упрощенные директивы сегментации, директивы определения данных. Особенности разработки 16-разрядных программ под MS-DOS.	0	3	1

2. Команды языка ассемблера. Команды пересылки данных. Арифметические команды. Режимы адресации. Команды передачи управления и сравнения. Команды манипулирования битами. Команды для работы с двоично-десятичными числами.	0	3	1
3. Директивы определения идентификаторов. Операции. Директива label.	0	2	1
Дидактическая единица: Взаимодействие языков С++ и ассемблера			
4. Использование функций на языке ассемблера в программах на языке С++. Вызов функций на языке С++ из программ на языке ассемблера. Использование локальных данных. Использование библиотечных функций языка Си в программах/подпрограммах на языке ассемблера. Использование вставок на языке ассемблера в программе на языке С++.	0	2	2
Дидактическая единица: Программирование сопроцессора			
5. Программирование сопроцессора	0	4	1
Дидактическая единица: Макросредства языка ассемблера			
6. Условные директивы. Макросы. Блоки повторения. Директивы EXITM и GOTO. Макрофункции. Макрооперации.	0	2	1, 4
Дидактическая единица: Сложные структуры данных			
7. Цепочечные команды. Структуры. Объединения. Работа с динамической памятью.	0	2	1

Таблица 3.2

Темы лабораторных работ	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 5				
Дидактическая единица: Основные элементы языка ассемблера				
1. Основные элементы языка ассемблера	6	7	1, 3, 5, 6, 7, 8	Изучить и приобрести практические навыки работы с основными командами языка ассемблера, функциями ввода-вывода, регистрами и символьными данными, активная форма проведения занятий: дискуссия на тему "Основные элементы языка ассемблера".
Дидактическая единица: Взаимодействие языков С++ и ассемблера				
2. Интерфейс с языком С++	3	5	1, 2, 5, 6, 7, 9	Изучить правила взаимодействия модулей на языках С++ и ассемблера; разработать программу, которая включает вызов подпрограммы на языке ассемблера из функции на языке С++, активная форма проведения занятий: дискуссия на тему "Взаимодействие языков С++ и ассемблера".
Дидактическая единица: Программирование сопроцессора				

3. Программирование сопроцессора	5	6	1, 2, 5, 6, 7, 9	Изучить архитектуру и средства программирования сопроцессора (модуля операций с плавающей точкой) на языке ассемблера и приобрести практические навыки работы с ними, активная форма проведения занятий: дискуссия на тему "Программирование сопроцессора".
----------------------------------	---	---	------------------	---

4. Самостоятельная работа обучающегося

№	Виды самостоятельной работы	Ссылки на результаты обучения	Часы на выполнение	Часы на консультации
Семестр: 5				
1	РГЗ	1, 2, 4, 5, 6, 7, 9	9	1
Тема "Макросредства". Соответствует дидактической единице "Макросредства языка ассемблера". : Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана.				
2	РГЗ	1, 2, 4, 5, 6, 7, 9	1	1
Подготовка к защите РГЗ производится по контрольным вопросам: Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана.				
3	подготовка к лекциям	1, 2, 4	8	2
Изучение материала предшествующих лекций.: Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана.				
4	подготовка к лабораторным работам	1, 2, 3, 5, 6, 7, 8, 9	35	0
Изучение соответствующего теоретического материала и методических указаний, анализ задачи, проектирование и разработка необходимого программного обеспечения, разработка системы тестов, оформление отчета, подготовка к защите.: Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана.				
5	Подготовка к аттестации	1, 2, 3, 4, 5	12	1
Изучение теоретического материала с использованием конспекта лекций, методических указаний к лабораторным работам.: Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана.				

5. Технология обучения

Для организации и контроля самостоятельной работы обучающихся, а также проведения консультаций применяются информационно-коммуникационные технологии (табл. 5.1).

Таблица 5.1

Деятельность	Информационно-коммуникационные технологии
Информирование	Портал НГТУ: http://ciu.nstu.ru/kaf/persons/842/about
Размещение учебных материалов	ЭБС

Таблица 5.2

Активные и интерактивные формы проведения занятий

№	Наименование активных форм	Коды формируемых компетенций
1	Дискуссия	ОПК.8;
Формируемые умения: з8. Знать основные способы разработки программ на ассемблере и работы в системе Visual C++; у6. Уметь использовать средства Visual C++ для разработки программ на ассемблере и двуязычных программ на C++ и ассемблере		
Краткое описание применения: Беседа студента с преподавателем на предложенные преподавателем темы (контрольные вопросы).		

6. Правила аттестации обучающихся по учебной дисциплине

Для аттестации обучающихся по дисциплине используется балльно-рейтинговая система (БРС), позволяющая выставять оценки по традиционной шкале и 15-уровневой ECTS.

Краткая информация о БРС приведена в табл. 6.1.

Подробная информация о БРС приводится в Приложении №4 к рабочей программе.

Таблица 6.1

Оцениваемые виды деятельности обучающихся	Мин. балл	Максимальный балл
Семестр: 5		
<i>Лекция:</i>	0	8
<i>Лабораторная №1: Выполнение</i>	6	9
Контролирующие материалы (список заданий) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		
<i>Лабораторная №1: Защита</i>	5	9
Контролирующие материалы (список вопросов) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		
<i>Лабораторная №2: Выполнение</i>	6	9
Контролирующие материалы (список заданий) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		
<i>Лабораторная №2: Защита</i>	5	9
Контролирующие материалы (список вопросов) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		
<i>Лабораторная №3: Выполнение</i>	6	9
Контролирующие материалы (список заданий) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		

Лабораторная №3: Защита	5	9
Контролирующие материалы (список вопросов) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		
РГЗ: Выполнение	6	9
Контролирующие материалы (список заданий) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		
РГЗ: Защита	5	9
Контролирующие материалы (список вопросов) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		
Зачет:	6	20
Контролирующие материалы (список вопросов) приводятся в "Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782 . - Загл. с экрана."		

В таблице 6.2 представлено соответствие форм контроля заявляемым требованиям к результатам освоения дисциплины.

Таблица 6.2

Коды компетенций ФГОС	Результаты обучения	Формы контроля		
		Защита ЛР	Защита РГЗ	Зачет
ОПК.8	з8. Знать основные способы разработки программ на ассемблере и работы в системе Visual C++	+	+	+
	уб. Уметь использовать средства Visual C++ для разработки программ на ассемблере и двуязычных программ на C++ и ассемблере	+	+	+

Фонд оценочных средств по дисциплине представлен в приложении № 1 к рабочей программе.

7. Литература

Основная литература

1. Юров В. И. Assembler : учебное пособие для вузов по направлению подготовки специалистов " Информатика и вычислительная техника" / В. И. Юров. - СПб. [и др.], 2007. - 636 с. : ил., табл.
2. Хабибуллин И. . Программирование на языке высокого уровня C/C++ : учебное пособие для вузов по направлению 654600 "Информатика и вычислительная техника" / И. Ш. Хабибуллин. - СПб, 2006. - 485 с. : ил.
3. Архитектура ЭВМ и вычислительных систем: Учебник / Н.В. Максимов, Т.Л. Партыка, И.И. Попов. - 5-е изд., перераб. и доп. - М.: Форум: НИЦ ИНФРА-М, 2015 - 512 с.: ил.; 60х90 1/16. - (Профессиональное образование). (п) ISBN 978-5-91134-742-0 - Режим доступа: <http://znanium.com/catalog.php?bookinfo=492687> - Загл. с экрана.

Интернет-ресурсы

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>
2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>
3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>
4. ЭБС "Znanium.com" : <http://znanium.com/>
5. :

8. Методическое и программное обеспечение

8.1 Методическое обеспечение

1. Лисицин Д. В. Программирование на языке Ассемблера [Электронный ресурс] : учебно-методическое пособие / Д. В. Лисицин, Р. В. Петров, И. А. Полетаева ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: http://ciu.nstu.ru/fulltext/unofficial/2011/lib_842_1320653325.doc. - Загл. с экрана.
2. Программирование на языке Ассемблера : методические указания к лабораторным работам по курсам "Языки программирования и методы трансляции" (для 3 курса ФПМИ по направлению 010400 - Прикладная математика и информатика) и "Теория вычислительных процессов и структур" (для 3 курса ФПМИ по направлению 010500 - Математическое обеспечение и администрирование информационных систем) / Новосиб. гос. техн. ун-т ; [сост.: Д. В. Лисицин, Р. В. Петров, И. А. Полетаева]. - Новосибирск, 2011. - 34, [2] с. : табл.
3. Лисицин Д. В. Низкоуровневое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Д. В. Лисицин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234782. - Загл. с экрана.

8.2 Специализированное программное обеспечение

1 Microsoft Visual C++

9. Материально-техническое обеспечение

Презентационное оборудование

№	Наименование	Назначение
1	Презентационное оборудование (мультимедиа-проектор, экран, компьютер для управления)	Для сопровождения лекций

Компьютерный класс

№	Наименование	Назначение
1	Компьютерный класс (Компьютеры объединены в локальную сеть с выходом в Internet)	Выполнение лабораторных работ и РГЗ.

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра теоретической и прикладной информатики

“УТВЕРЖДАЮ”
ДЕКАН ФПМИ
д.т.н., доцент В.С. Тимофеев
“ ” _____ г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

УЧЕБНОЙ ДИСЦИПЛИНЫ

Низкоуровневое программирование

Образовательная программа: 02.03.03 Математическое обеспечение и администрирование информационных систем, профиль: Математическое и программное обеспечение информационных технологий

1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине Низкоуровневое программирование приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
ОПК.8 способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальным и средствами, поддерживающими создание программного обеспечения (далее - ПО	з8. Знать основные способы разработки программ на ассемблере и работы в системе Visual C++	Директивы определения идентификаторов. Операции. Директива label. Интерфейс с языком C++ Использование функций на языке ассемблера в программах на языке C++. Вызов функций на языке C++ из программ на языке ассемблера. Использование локальных данных. Использование библиотечных функций языка Си в программах/подпрограммах на языке ассемблера. Использование вставок на языке ассемблера в программе на языке C++. Команды языка ассемблера. Команды пересылки данных. Арифметические команды. Режимы адресации. Команды передачи управления и сравнения. Команды манипулирования битами. Команды для работы с двоично-десятичными числами. Основные элементы языка ассемблера Программирование сопроцессора Условные директивы. Макросы. Блоки повторения. Директивы EXITM и GOTO. Макрофункции. Макрооперации. Цепочные команды. Структуры. Объединения. Работа с динамической памятью. Язык ассемблера, задачи, решаемые с его помощью. Регистры процессора. Процесс разработки программы, структура программы. Понятия команд, директив и операций. Формат записи команд. Директивы segment, proc, assume, end, align, упрощенные директивы сегментации, директивы определения данных. Особенности разработки 16-разрядных программ под MS-DOS.	Отчет по лабораторной работе № 1, контрольные вопросы 1–16, Отчет по лабораторной работе № 2, контрольные вопросы 1–11, Отчет по лабораторной работе № 3, контрольные вопросы 1–24 РГЗ, задания 1-5	Зачет, вопросы 1, 3-59.

ОПК.8	уб. Уметь использовать средства Visual C++ для разработки программ на ассемблере и двуязычных программ на C++ и ассемблере	Интерфейс с языком C++ Основные элементы языка ассемблера Программирование сопроцессора	Отчет по лабораторной работе № 1, контрольные вопросы 1–5, Отчет по лабораторной работе № 2, контрольные вопросы 1–11, Отчет по лабораторной работе № 3, задания 2–3, РГЗ, задания 1-5	Зачет, вопросы 2, 18-24, 60
-------	--	---	---	-----------------------------

2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 5 семестре - в форме дифференцированного зачета, который направлен на оценку сформированности компетенций ОПК.8.

Зачет проводится в письменной форме, по комплектам заданий, комплекты заданий состояются из вопросов, приведенных в паспорте зачета, позволяющих оценить показатели сформированности соответствующих компетенций.

На подготовку к зачету отводится 13 часов.

Кроме того, сформированность компетенции проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

В 5 семестре обязательным этапом текущей аттестации является расчетно-графическое задание (РГЗ). Требования к выполнению РГЗ, состав и правила оценки сформулированы в паспорте РГЗ.

В 5 семестре этапами текущей аттестации являются лабораторные работы. Требования к их выполнению, состав и правила оценки сформулированы в паспорте лабораторных работ.

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенции ОПК.8, за которые отвечает дисциплина, на разных уровнях.

Общая характеристика уровней освоения компетенций.

Ниже порогового. Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

Пороговый. Уровень выполнения работ отвечает большинству основных требований, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые виды заданий выполнены с ошибками.

Базовый. Уровень выполнения работ отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки.

Продвинутый. Уровень выполнения работ отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.

Паспорт зачета

по дисциплине «Низкоуровневое программирование», 5 семестр

1. Методика оценки

Зачет проводится в письменной форме, по комплекту заданий. Комплект заданий состоит из двух вопросов и формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-30, второй вопрос из диапазона вопросов 31-60 (список вопросов приведен ниже).

Форма комплекта заданий для зачета

Комплект заданий
к зачету по дисциплине «Низкоуровневое программирование»

1. Директива PROC, команды CALL, RET, JMP
2. Команды передачи данных: FLD, FST(P), FXCH, FILD, FIST(P), FBLD, FBSTP

2. Критерии оценки

- Ответ на комплект заданий для зачета считается **неудовлетворительным**, если студент ответил на вопросы слишком неполно, показал слишком неполное владение теоретическими и практическими аспектами, относящихся к вопросу элементов языка ассемблера, оценка составляет 4 балла
- Ответ на комплект заданий для зачета засчитывается на **пороговом** уровне, если студент лишь частично ответил на вопросы, показал некоторое владение теоретическими и практическими аспектами, относящихся к вопросу элементов языка ассемблера, оценка составляет 6 баллов
- Ответ на комплект заданий для зачета засчитывается на **базовом** уровне, если студент, отвечая на вопросы, не дал развернутых ответов, показал владение теоретическими и практическими аспектами, относящихся к вопросу элементов языка ассемблера, оценка составляет 16 баллов
- Ответ на комплект заданий для зачета засчитывается на **продвинутом** уровне, если студент дал развернутые ответы на вопросы, показал свободное владение теоретическими и практическими аспектами, относящихся к вопросу элементов языка ассемблера, оценка составляет 20 баллов

3. Шкала оценки

Зачет считается сданным, если сумма баллов по всем заданиям комплекта оставляет не менее 6 баллов (из 20 возможных).

В общей оценке по дисциплине баллы за зачет учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Вопросы к зачету по дисциплине «Низкоуровневое программирование»

1. Особенности языка ассемблера и задачи, решаемые с его использованием
2. Разработка программы на языке ассемблера в среде Visual C++
3. Регистры микропроцессора
4. Директивы SEGMENT, ASSUME, END, упрощенные директивы сегментации
5. Директивы определения данных, директива ALIGN
6. Команды пересылки данных
7. Команды ADD, ADC, XADD, INC
8. Команды SUB, SBB, DEC, NEG
9. Команды умножения и деления
10. Команды преобразования типа
11. Регистровая, непосредственная и прямая адресация
12. Косвенная регистровая адресация и адресация по базе
13. Прямая адресация с индексированием, адресация по базе с индексированием
14. Директива PROC, команды CALL, RET, JMP
15. Команды условной передачи управления, команда CMP
16. Команды SETx, CMOVx
17. Команды управления циклами
18. Основы взаимодействия языков C++ и ассемблера. Передача управления в подпрограмму и обратно
19. Передача данных из программы на языке C++ в подпрограмму на языке ассемблера: Использование глобальных данных
20. Передача данных из программы на языке C++ в подпрограмму на языке ассемблера: Использование аргументов функции
21. Возвращение данных из подпрограммы на языке ассемблера в программу на языке C++
22. Вызов функций на языке C++ из программ на языке ассемблера, использование локальных данных
23. Использование библиотечных функций языка Си в программах или подпрограммах на языке ассемблера
24. Использование вставок на языке ассемблера в программе на языке C++
25. Команды AND, OR, XOR, NOT, TEST
26. Команды сканирования битов, команды проверки и модификации битов
27. Команды сдвига
28. Команды циклического сдвига, команды сдвига двойной точности
29. Форматы хранения двоично-десятичных чисел
30. Коррекция результатов арифметических команд для двоично-десятичных чисел
31. Форматы и способы задания двоичных, двоично-десятичных и вещественных чисел в сопроцессоре
32. Специальные численные значения сопроцессора
33. Регистровый стек сопроцессора, регистр тегов TWR сопроцессора
34. Регистры состояния (SWR) и управления (CWR) сопроцессора
35. Особенности вычисления выражений в сопроцессоре с использованием постфиксной записи
36. Команды передачи данных: FLD, FST(P), FXCH, FILD, FIST(P), FBLD, FBSTP
37. Команды сравнения данных: FCOM, FCOMP(P), FICOM(P), FTST, FCOMI(P)
38. Арифметические команды: целочисленные (FIADD, FISUB(R), FIMUL, FIDIV(R)) и вещественные (Fx(P), FxR(P))
39. Дополнительные арифметические команды: FSQRT, FABS, FCHS, FXTRACT, FSCALE, FRNDINT, FPREM, FPREMI
40. Команды тригонометрических функций: FCOS, FSIN, FSINCOS, FPTAN, FPATAN

41. Команды вычисления логарифмов и степеней: F2XM1, FYL2X, FYL2XP1
42. Команды управления сопроцессором: FINIT, FINCSTP, FDECSTP, FFREE
43. Исключения сопроцессора и их обработка
44. Директивы определения идентификаторов
45. Операции: арифметические, логические, отношения
46. Операции, возвращающие значения, операции присваивания атрибутов, директива LABEL
47. Условные директивы
48. Сравнение макросов и процедур, определение и вызов макросов
49. Директивы LOCAL, IFB, IFNB
50. Блоки повторения
51. Директивы EXITM и GOTO, макрофункции
52. Макрооперации
53. Общая характеристика цепочечных команд, префиксы повторения
54. Команды пересылки и сравнения цепочек
55. Команды сканирования, загрузки и сохранения цепочек
56. Определение структуры, объявление структурных переменных
57. Обращение к структурным переменным, вложенные структуры
58. Объединения
59. Работа с динамической памятью
60. Средства операционной системы Windows для ввода-вывода информации в консольных приложениях

Паспорт расчетно-графического задания

по дисциплине «Низкоуровневое программирование», 5 семестр

1. Методика оценки

Задание 1. Изучить состав и средства задания макроопределений.

Задание 2. Написать программу с использованием необходимого макроопределения, реализующую функцию заданного варианта.

Задание 3. Отладить программу, убедиться в правильности ее работы на тестовых примерах.

Задание 4. Оформить отчет. Отчет должен содержать постановку задачи, алгоритм, текст разработанной программы и результаты тестирования.

Задание 5. Защитить работу, ответив на вопросы преподавателя.

Список контрольных вопросов.

1. Сравнение макроопределений и процедур.
2. Макрокоманды, макроопределения и макрорасширения.
3. Директива LOCAL.
4. Директивы повторения WHILE и REPT.
5. Директивы повторения IRP и IRPC.
6. Условные директивы IFE, IF, IFDEF, IFNDEF.
7. Условные директивы IFB, IFNB, IFIDN, IFIDNI, IFDIF, IFDIFI.
8. Директивы определения идентификаторов.
9. Операции в макроопределениях.

2. Критерии оценки

- Работа считается **не выполненной**, если макросредства языка ассемблера не освоены, необходимые практические навыки работы с макросами не сформированы, выполнены не все предусмотренные задания или задания выполнены с ошибками, отчет оформлен с недостатками, оценка составляет 4 балла
- Работа считается выполненной **на пороговом** уровне, если макросредства языка ассемблера освоены частично, но пробелы не носят существенного характера, необходимые практические навыки работы с макросами в основном сформированы, все предусмотренные задания выполнены, но некоторые задания выполнены с ошибками, отчет оформлен с недостатками, оценка составляет 7 баллов, если работа выполнена в срок, и 6 баллов, если график работы нарушен
- Работа считается выполненной **на базовом** уровне, если макросредства языка ассемблера освоены полностью, без пробелов, некоторые практические навыки работы с макросами сформированы недостаточно, все предусмотренные задания выполнены, некоторые из выполненных заданий, возможно, содержат несущественные ошибки, отчет оформлен без недостатков, оценка составляет 8 баллов, если работа выполнена в срок, и 6 баллов, если график работы нарушен
- Работа считается выполненной **на продвинутом** уровне, если макросредства языка ассемблера освоены полностью, без пробелов, необходимые практические навыки работы с макросами сформированы, все предусмотренные задания выполнены с качеством,

близким к максимальному, отчет оформлен без недостатков, оценка составляет 9 баллов, если работа выполнена в срок, и 7 баллов, если график работы нарушен

- Работа считается **не защищенной**, если студент очень неполно ответил на вопросы, показал недостаточное владение теоретическими аспектами изучаемых элементов языка ассемблера, оценка составляет 3 балла
- Работа считается защищенной **на пороговом** уровне, если студент лишь частично ответил на вопросы, показал некоторое владение теоретическими аспектами изучаемых элементов языка ассемблера, оценка составляет 7 баллов, если работа защищена в срок, и 5 баллов, если график защиты работы нарушен
- Работа считается защищенной **на базовом** уровне, если студент, отвечая на вопросы, не дал развернутых ответов, показал владение основными теоретическими аспектами изучаемых элементов языка ассемблера, оценка составляет 8 баллов, если работа защищена в срок, и 6 баллов, если график защиты работы нарушен
- Работа считается защищенной **на продвинутом** уровне, если студент дал развернутые ответы на вопросы, показал свободное владение основными теоретическими аспектами изучаемых элементов языка ассемблера, оценка составляет 9 баллов, если работа защищена в срок, и 7 баллов, если график защиты работы нарушен

3. Шкала оценки

В общей оценке по дисциплине баллы за РГЗ(Р) учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Примерный перечень тем РГЗ

1. Подсчитать количество вхождений заданного символа в каждую строку текста. Макрокоманда должна обрабатывать одну строку исходного текста.

2. Заменить все вхождения заданного символа в тексте на указанный новый символ. Макрокоманда должна обрабатывать одну строку исходного текста.

3. Удалить все вхождения заданного символа из текста. Макрокоманда должна обрабатывать одну строку исходного текста.

4. Подсчитать количество слов в каждой строке исходного текста, считая словом последовательность знаков между пробелами. Макрокоманда должна обрабатывать одну строку исходного текста.

5. Задан текст, состоящий из строк. Переместить заданный символ, если он содержится в строке, в начало строки. Макрокоманда должна обрабатывать одну строку исходного текста.

6. Задан текст, состоящий из строк. Переместить заданный символ, если он содержится в строке, в конец строки. Макрокоманда должна обрабатывать одну строку исходного текста.

7. Задан текст, состоящий из строк. Удалить все пробелы из строки символов. Макрокоманда должна обрабатывать одну строку исходного текста.

8. Задана строка, состоящая из слов. Словом считается последовательность символов между пробелами. В исходной строке оставить между словами только по одному пробелу. Макрокоманда должна обрабатывать одно слово исходного текста.

9. Задана строка, состоящая из слов. Словом считается последовательность символов между пробелами. В исходной строке изменить порядок следования слов на инверсный. Макрокоманда должна обрабатывать одно слово исходного текста.

10. Задана строка, состоящая из слов. Словом считается последовательность символов между пробелами. Из исходной строки удалить слова, начинающиеся с заданного символа. Макрокоманда должна обрабатывать одно слово исходного текста.

11. Задана строка, состоящая из слов. Словом считается последовательность символов между пробелами. В исходной строке слова, начинающиеся с заданного символа, заменить знаком \$. Макрокоманда должна обрабатывать одно слово исходного текста.

12. Задана строка, состоящая из слов. Словом считается последовательность символов между пробелами. В исходной строке удалить слова, содержащие хотя бы одну десятичную цифру. Макрокоманда должна обрабатывать одно слово исходного текста.

Паспорт лабораторных работ

по дисциплине «Низкоуровневое программирование», 5 семестр

1. Методика оценки

ЛАБОРАТОРНАЯ РАБОТА №1

Задание 1. Изучить основные команды языка Ассемблера, функции операционной системы, осуществляющие ввод-вывод, работу символьными данными.

Задание 2. По предложенному преподавателем варианту разработать программу на языке Ассемблера, решающую поставленную задачу:

- 1) ввод с клавиатуры 2-х чисел в заданной системе счисления;
- 2) выполнение арифметической операции над этими числами (в предположении, что размер чисел не вызывает переполнения регистров);
- 3) вывод результата в заданной системе счисления.

Все промежуточные данные должны сохраняться в памяти. При выводе результата не использовать функцию `wsprintfA`.

Задание 3. Отладить программу, убедиться в правильности ее работы на тестовых примерах.

Задание 4. Оформить отчет по лабораторной работе. Отчет должен содержать постановку задачи, алгоритм, текст разработанной программы и результаты тестирования.

Задание 5. Защитить лабораторную работу, ответив на вопросы преподавателя.

Контрольные вопросы

1. Функция `GetStdHandle`: назначение, параметры, примеры использования.
2. Функция `WriteConsoleA`: назначение, параметры, примеры использования.
3. Функция `CharToOem`: назначение, параметры, примеры использования.
4. Функция `ReadConsoleA`: назначение, параметры, примеры использования.
5. Функция `wsprintfA`: назначение, параметры, примеры использования.
6. Команда `mov`: назначение, операнды, примеры использования.
7. Стек и команды работы со стеком `push`, `pop`: назначение, операнды, примеры использования.
8. Команды сохранения и извлечения флагов и регистров из стека.
9. Команды сложения.
10. Команды вычитания.
11. Команды деления.
12. Команды умножения.
13. Директивы определения данных.
14. Регистровая, непосредственная и прямая адресация.
15. Косвенная регистровая адресация и адресация по базе.
16. Прямая адресация с индексированием и адресация по базе с индексированием.

ЛАБОРАТОРНАЯ РАБОТА № 2

Задание 1. Изучить способы вызова процедуры на языке Ассемблера, возврата из процедуры.

Задание 2. Изучить условия взаимодействия функции на языке C++ с процедурой на языке Ассемблера. Изучить правила передачи управления в процедуру и обратно. Изучить способы обмена данными между вызывающей функцией на языке C++ и

процедурой на языке Ассемблера.

Задание 3. Написать процедуру на языке Ассемблера, реализующую функцию заданного варианта. Написать вызывающую функцию на языке C++, осуществляющую ввод исходных данных и вывод результатов.

Задание 4. Отладить программу, убедиться в правильности ее работы на тестовых примерах.

Задание 5. Оформить отчет по лабораторной работе. Отчет должен содержать постановку задачи, алгоритм, текст разработанной программы и результаты тестирования.

Задание 6. Защитить лабораторную работу, ответив на вопросы преподавателя.

Контрольные вопросы

1. Определение процедуры.
2. Команда вызова процедуры CALL.
3. Команда возврата из процедуры.
4. Косвенный вызов процедуры.
5. Основы взаимодействия языков C++ и Ассемблера.
6. Использование глобальных данных для передачи данных процедуре на языке Ассемблера.
7. Использование аргументов функции для передачи данных процедуре на языке Ассемблера.
8. Возвращение значений из процедуры на языке Ассемблера.
9. Вызов функций на языке C++ из процедур на языке Ассемблера.
10. Использование локальных данных.
11. Использование вставок на языке Ассемблера в программе на языке C++.

ЛАБОРАТОРНАЯ РАБОТА № 3

Задание 1. Изучить архитектуру и средства программирования сопроцессора на языке ассемблера. Изучить численный метод решения нелинейного уравнения, заданный в варианте.

Задание 2. Написать программу, реализующую метод заданного варианта.

2.1. Программа должна состоять из модулей на C++ и ассемблере, причем в модуле на C++ осуществляется ввод-вывод, а все вычисления – в модуле на ассемблере.

2.2. Следует оформить отдельными процедурами на ассемблере:

- вычисление функции, задающей левую часть уравнения;
- вычисление (если требуется) производных функции;
- вычисления, связанные с методом решения.

2.3. Входными данными является точность решения. Выходными данными являются решение и число итераций.

2.4. Для функции и (если требуется) ее производных необходимо использовать алгоритм вычисления выражений в постфиксной записи (любые отклонения от него должны быть обоснованы повышением эффективности). Необходимо обращать особое внимание на грамотное использование стека сопроцессора.

Задание 3. Отладить программу, убедиться в правильности ее работы на тестовых примерах.

Задание 4. Оформить отчет по лабораторной работе. Отчет должен содержать постановку задачи, график функции с указанием начального приближения и искомого корня, алгоритм, текст разработанной программы (на этапах вычислений в комментариях указывать состояние стека сопроцессора) и результаты тестирования.

Задание 5. Защитить лабораторную работу, ответив на вопросы преподавателя.

Контрольные вопросы

1. Форматы и способы задания целых и двоично-десятичных чисел.
2. Форматы и способы задания вещественных чисел.
3. Денормализованные вещественные числа, нуль, бесконечность.
4. Нечисла (SNAN, QNAN).

5. Регистровый стек: принципы работы.
6. Регистр состояния SWR: назначение, основные поля.
7. Регистр управления CWR: назначение, основные поля.
8. Регистр тегов TWR: назначение, принцип использования.
9. Особенности вычисления выражений с использованием постфиксной записи.
10. Инициализация сопроцессора: команда FINIT.
11. Команды передачи данных в вещественном формате: FLD, FST(P), команда обмена FXCH.
12. Команды передачи данных в целочисленном и двоично-десятичном формате: FILD, FIST(P), FBLD, FBSTP.
13. Команды сравнения данных: FCOM, FCOMP(P), FICOM(P), FTST, FCOMI(P).
14. Целочисленные арифметические команды: FIADD, FISUB(R), FIMUL, FIDIV(R).
15. Вещественные арифметические команды: Fx(P), FxR(P).
16. Дополнительные арифметические команды: FSQRT, FABS, FCHS, FXTRACT, FSCALE, FRNDINT.
17. Дополнительные арифметические команды: FPREM, FPREM1
18. Команды тригонометрических функций: FCOS, FSIN, FSINCOS, FPTAN, FPATAN.
19. Команды вычисления логарифмов и степеней: F2XM1, FYL2X, FYL2XP1.
20. Команды управления сопроцессором: FINCSTP, FDECSTP, FFREE.
21. Исключения сопроцессора и их обработка: недействительная операция.
22. Исключения сопроцессора и их обработка: деление на ноль, денормализация операнда.
23. Исключения сопроцессора и их обработка: переполнение и антипереполнение, неточный результат.
24. Исключения сопроцессора и их обработка: приоритет особых случаев.

2. Критерии оценки

- Лабораторная работа считается **не выполненной**, если студент выполнил задание частично, отчет оформил с недостатками, показал недостаточное владение практическими навыками, оценка составляет 4 балла
- Лабораторная работа считается выполненной **на пороговом** уровне, если студент выполнил задание не полностью, отчет оформил с недостатками, показал некоторое практическое владение изучаемыми элементами языка ассемблера, оценка составляет 7 баллов, если работа выполнена в срок, и 6 баллов, если график работы нарушен
- Лабораторная работа считается выполненной **на базовом** уровне, если студент выполнил задание полностью, но показал неполное практическое владение изучаемыми элементами языка ассемблера, отчет оформил без недостатков, оценка составляет 8 баллов, если работа выполнена в срок, и 6 баллов, если график работы нарушен
- Лабораторная работа считается выполненной **на продвинутом** уровне, если студент выполнил задание полностью, отчет оформил без недостатков, показал свободное практическое владение изучаемыми элементами языка ассемблера, оценка составляет 9 баллов, если работа выполнена в срок, и 7 баллов, если график работы нарушен
- Лабораторная работа считается **не защищенной**, если студент ответил на вопросы слишком неполно, не показал владение теоретическими навыками, оценка составляет 3 балла
- Лабораторная работа считается защищенной **на пороговом** уровне, если студент лишь частично ответил на вопросы, показал некоторое владение теоретическими аспектами изучаемых элементов языка ассемблера, оценка составляет 7 баллов, если работа защищена в срок, и 5 баллов, если график защиты работы нарушен

- Лабораторная работа считается защищенной **на базовом** уровне, если студент, отвечая на вопросы, не дал развернутых ответов, показал владение теоретическими аспектами изучаемых элементов языка ассемблера, оценка составляет 8 баллов, если работа защищена в срок, и 6 баллов, если график защиты работы нарушен
- Лабораторная работа считается защищенной **на продвинутом** уровне, если студент дал развернутые ответы на вопросы, показал свободное владение теоретическими аспектами изучаемых элементов языка ассемблера, оценка составляет 9 баллов, если работа защищена в срок, и 7 баллов, если график защиты работы нарушен

3. Шкала оценки

В общей оценке по дисциплине баллы за РГЗ учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Перечни вариантов

ЛАБОРАТОРНАЯ РАБОТА №1

1. Сложение чисел, ввод в восьмеричной системе счисления (не менее 3-х знаков каждое число), вывод в десятичной системе счисления.
2. Сложение чисел, ввод в шестнадцатеричной системе счисления (не менее 3-х знаков каждое число), вывод в десятичной системе счисления.
3. Сложение чисел, ввод в десятичной системе счисления (не менее 4-х знаков каждое число), вывод в шестнадцатеричной системе счисления.
4. Вычитание чисел, ввод в восьмеричной системе счисления (не менее 3-х знаков каждое число), вывод в шестнадцатеричной системе счисления.
5. Вычитание чисел, ввод в шестнадцатеричной системе счисления (не менее 3-х знаков каждое число), вывод в десятичной системе счисления.
6. Вычитание чисел, ввод в десятичной системе счисления (не менее 3-х знаков каждое число), вывод в шестнадцатеричной системе счисления.
7. Умножение чисел, ввод в восьмеричной системе счисления (не менее 3-х знаков каждое число), вывод в десятичной системе счисления.
8. Умножение чисел, ввод в шестнадцатеричной системе счисления (не менее 3-х знаков каждое число), вывод в десятичной системе счисления.
9. Умножение чисел, ввод в десятичной системе счисления (не менее 4-х знаков каждое число), вывод в шестнадцатеричной системе счисления.
10. Сложение чисел, ввод в двоичной системе счисления (не менее 16-ти знаков каждое число), вывод в десятичной системе счисления.
11. Умножение чисел, ввод в двоичной системе счисления (не менее 16-ти знаков каждое число), вывод в десятичной системе счисления.
12. Вычитание чисел, ввод в двоичной системе счисления (не менее 16-ти знаков каждое число), вывод в шестнадцатеричной системе счисления.

ЛАБОРАТОРНАЯ РАБОТА № 2

1. Написать программу формирования сжатой строки символов. Сжатие заключается в удалении точек из исходной строки при просмотре ее слева направо.
2. Написать программу формирования сжатой строки символов. Сжатие заключается в удалении точек из исходной строки при просмотре ее справа налево.
3. Написать программу выделения из исходной строки подстроки символов заданной длины с указанного номера позиции.
4. Написать программу, определяющую номер позиции, с которой начинается первое слева вхождение заданной конфигурации символов в исходную строку.
5. Написать программу формирования строки из исходной путем заданного числа повторений исходной строки.

6. Написать программу, выполняющую следующую функцию. Заданы две строки. Проверить вхождение каждого символа первой строки во вторую строку. Если какой-либо (первый слева) символ первой строки не представлен во второй строке, то фиксируется номер позиции этого символа в первой строке.

7. Написать программу, которая по исходной строке формирует инвертированную строку и проверяет, совпадают ли обе строки.

8. Написать программу, находящую максимальный и минимальный символы в исходной строке.

9. Написать программу, заменяющую в исходной строке все десятичные цифры на заданный символ.

10. Написать программу, удаляющую из исходной строки повторные вхождения заданного символа.

11. Написать программу, удаляющую точки в конце исходной строки.

12. Написать программу, удаляющую из исходной строки заданную конфигурацию символов.

ЛАБОРАТОРНАЯ РАБОТА № 3

1. Метод половинного деления
2. Метод хорд
3. Метод Ньютона
4. Метод секущих
5. Метод Стеффенсена
6. Метод Хейли
7. Метод Ньютона–половинного деления
8. Метод Ньютона с подвижным полюсом
9. Однопараметрический полюсный метод Ньютона
10. Метод простых итераций
11. Δ^2 -процесс Эйткена
12. Метод Вегстейна

Правила аттестации студентов по учебной дисциплине Низкоуровневое программирование

Текущая аттестация студента проводится по результатам выполнения им лабораторных работ и РГЗ, а также по результатам работы на лекциях.

Баллы за каждую из трех лабораторных работ и РГЗ выставляются по правилам, приведенным в таблице 1.

Таблица 1

Вид деятельности	Минимальное количество баллов	Максимальное количество баллов
Выполнение лабораторной работы, РГЗ (оценивается полнота выполнения задания, работоспособность разработанного программного обеспечения, полнота набора тестов, полнота отчета, его соответствие требованиям оформления, своевременность* выполнения лабораторной работы)	6	9
Защита (оценивается полнота и глубина теоретических знаний, умение применить знания на практике, своевременность* защиты)	5	9
Всего за лабораторную работу, РГЗ	11	18

* график выполнения и защиты лабораторных работ, РГЗ приведен ниже

Минимальное допустимое количество баллов за лабораторную работу, РГЗ получает студент, выполнивший задание не полно, оформивший отчет с недостатками и показавший на защите некоторое владение теоретическими и практическими навыками.

Минимальное допустимое количество баллов за лабораторную работу, РГЗ получает студент, полностью выполнивший задание, оформивший отчет без недостатков и показавший на защите свободное владение теоретическими и практическими навыками.

Студент, не выполнивший и не защитивший лабораторную работу, не допускается к выполнению следующей лабораторной работы.

График выполнения и защиты лабораторных работ, РГЗ:

- 1 – 7 неделя – 1-я лабораторная работа;
- 8 – 12 неделя – 2-я лабораторная работа;
- 13 – 18 неделя – 3-я лабораторная работа, РГЗ.

За работу на лекциях студент получает от 0 до 8 баллов (по 1 баллу за каждую лекцию, исключая вводную).

Оценка за контрольные недели выставляется в соответствии с таблицей 2.

Таблица 2

Номер недели	7 контрольная неделя			12 контрольная неделя		
Оценка за контрольную неделю	0	1	2	0	1	2
Текущий рейтинг студента	Менее 7	От 7 до 11	15 и более	Менее 22	От 22 до 38	30 и более
Количество зачтенных лабораторных заданий	0	1*	1	1	2*	2

* лабораторная работа выполнена, но не защищена

Зачет сдается в виде письменных ответов на два теоретических вопроса. К зачету допускаются студенты, выполнившие и защитившие все лабораторные работы, РГЗ и набравшие не менее 44 баллов. Максимальное количество баллов за теоретический зачет – 20 баллов, минимальное – 6. Оценивается правильность ответа, последовательность и полнота ответа, владение профессиональной терминологией. Минимальное количество баллов за теоретический зачет получает студент, частично ответивший на один из вопросов.

Итоговая оценка в соответствии с 15-уровневой шкалой оценок ECTS и в традиционной форме («зачтено») выставляется на основе набранных баллов (таблица 3).

Таблица 3

Характеристика работы студента	Диапазон баллов рейтинга	Оценка ECTS	Традиционная (4-уровневая) шкала оценки
«Отлично» – работа высокого качества, уровень выполнения отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	98-100	A+	Зачтено
	94-97	A	
	90-93	A-	
«Очень хорошо» – работа хорошая, уровень выполнения отвечает большинству требований, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному	87-89	B+	
	83-86	B	
	80-82	B-	
«Хорошо» – уровень выполнения работы отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки	77-79	C+	
	73-76	C	
	70-72	C-	

«Удовлетворительно» – уровень выполнения работы отвечает большинству основным требованиям, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнены, некоторые виды заданий выполнены с ошибками	67-69	D+	
	63-66	D	
	60-62	D-	
«Посредственно» – работа слабая, уровень выполнения не отвечает большинству требований, теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному	50-59	E	
«Неудовлетворительно» (с возможностью пересдачи) – теоретическое содержание курса освоено частично, необходимые практические навыки работы не сформированы, большинство предусмотренных программой обучения учебных заданий не выполнено, либо качество их выполнения оценено числом баллов, близким к минимальному; при дополнительной самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий	25-49	FX	Не зачтено
«Неудовлетворительно» (без возможности пересдачи) – теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, все выполненные учебные задания содержат грубые ошибки; дополнительная самостоятельная работа над материалом курса не приведет к какому-либо значительному повышению качества выполнения учебных заданий	0-24	F	