«

»

“ ”

“ ”

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Машинные языки и программирование

: 09.03.01

, :

: 3, : 5

		5
1	()	3
2		108
3	, .	61
4	, .	18
5	, .	0
6	, .	36
7	, .	18
8	, .	2
9	, .	5
10	, .	47
11	(, ,)	
12		

(): 09.03.01

5 12.01.2016 ., : 09.02.2016 .

: 1, ,

(): 09.03.01

, 7 20.06.2017

, 6 21.06.2017

:

,

:

,

:

. . . .

1.

1.1

Компетенция ФГОС: ОПК.1 способность устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем; в части следующих результатов обучения:	
1.	-
Компетенция ФГОС: ОПК.4 способность участвовать в настройке и паладке программно-аппаратных комплексов; в части следующих результатов обучения:	
1.	
Компетенция ФГОС: ОПК.5 способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности; в части следующих результатов обучения:	
8.	
Компетенция ФГОС: ПК.3 способность обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности; в части следующих результатов обучения:	
4.	
6.	

2.

2.1

(	)
---	---

.1. 1	
1.уметь устанавливать, тестировать, испытывать и использовать программно-аппаратные средства вычислительных и информационных систем	; ;
.3. 4	
2.знать современные компиляторы, отладчики и оптимизаторы программного кода	; ;
.3. 6	
3.знать основы методологии создания программного обеспечения	; ;
.4. 1	
4.знать основы построения и архитектуры вычислительной техники	; ;
.5. 8	
5.владеть персональным компьютером как средством управления информацией	; ;

3.

3.1

	,	.		
: 5				
:				

1.	IBM PC	0	2	1, 2, 3, 4, 5	
2.	IBM PC.	0	2	1, 2, 3, 4, 5	
: IBM PC					
3.	(.EXE .COM).	0	2	1, 2, 3, 4, 5	
4.		0	2	1, 2, 3, 4, 5	
5.	C/C++.	0	2	1, 2, 3, 4, 5	
:					
6.	Include	0	2	1, 2, 3, 4, 5	
:					
7.		0	2	1, 2, 3, 4, 5	
: Borland C++ Turbo Accembler					

8.	Assembler Turbo Borland C++.	0	2	1, 2, 3, 4, 5	
9.	++.	0	2	1, 2, 3, 4, 5	

3.2

: 5					
:					
1.		4	6	1, 2, 3, 4, 5	
2.		2	4	1, 2, 3, 4, 5	
3.	MS DOS	4	8	1, 2, 3, 4, 5	
4.	BIOS	2	4	1, 2, 3, 4, 5	
5.		2	8	1, 2, 3, 4, 5	
6.	++	4	6	1, 2, 3, 4, 5	

4.

: 5				
1		1, 2, 3, 4, 5	8	2

4. ЭБС "Znanium.com" : <http://znanium.com/>

5. :

8.

8.1

1. Афанасьев В. А. Программирование. Раздел "Assembler IBM PC" [Электронный ресурс] : электронный учебно-методический комплекс / В. А. Афанасьев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000156300. - Загл. с экрана.

2. Юров В. И. Assembler : [учебное пособие по направлению " Информатика и вычислительная техника"] / В. И. Юров. - СПб. [и др.], 2010. - 636 с. : ил., табл.

8.2

1 C++Builder 2007 Professional R2

9.

-

1	(-) , ,	

1	(Internet)	

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

Кафедра автоматизированных систем управления

“УТВЕРЖДАЮ”
ДИРЕКТОР ИСТР
д.соц.н., профессор Л.А. Осьмук
“ ____ ” _____ _____ г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

УЧЕБНОЙ ДИСЦИПЛИНЫ

Машинные языки и программирование

Образовательная программа: 09.03.01 Информатика и вычислительная техника, профиль:
Автоматизированные системы обработки информации и управления в социальной сфере

1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине Машинные языки и 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.1 способность установить программное и аппаратное обеспечение для информационных и автоматизированных систем	у1. уметь установить, тестировать, испытывать и использовать программно-аппаратные средства вычислительных и информационных систем	<p>Ассемблирование и отладка готовых программ. Изучение системы команд, их кодирование, способы адресации операндов. Введение в язык ассемблера. Архитектурное своеобразие процессоров ПК.</p> <p>Характеристика типов данных в ПК IBM PC Представление чисел в формате с плавающей точкой. Команды передачи данных. Способы адресации операндов. Арифметические команды и машинная арифметика. Команды логических операций и сдвига. Операции над битами и байтами. Команды передачи управления и организации циклов. Команды обработки строк. Программирование типовых управляющих структур C/C++. Концепция модульного программирования.</p> <p>Процедуры, передача параметров между программой и процедурой. Многомодульные программы и их ассемблирование. Макросы. Блоки повторения. Библиотеки макросов. Директива Include Настройка программной среды.</p> <p>Принципы взаимодействия между Turbo Assembler и Borland C++. Практические приёмы построения двуязычных программ. Процедура вызова из C++ ассемблерных функций. Передача параметров ассемблерным модулям. Соглашения о сохранении регистров. Возврат значений из ассемблерного модуля в программу C++. Вызов функций C++ из программы на ассемблере.</p> <p>Демонстрационная программа с использованием взаимных вызовов. Преобразование форматов числовых данных в</p>	Отчет по лабораторной работе, РГЗ.	Зачет, вопросы 1 – 28.

		<p>операциях ввода-вывода Программирование алгоритмов сложных арифметических операций Программная модель IBM PC. Работа со стеком. Режимы адресации и форматы команд. Разработка программы на языке C++ с ассемблерным модулем. Синтаксис ассемблера. Основные этапы разработки и отладки программы на ассемблере. Объектный код программы. Структуры данных и их инициализация в программе. Директивы резервирования и инициализации данных. Макроопределения или директивы присваивания. Выражения и операторы. Система прерываний. Программные и аппаратные прерывания и процедуры их обработки. Прикладные обработчики прерываний Системные функции MS DOS ввода - вывода символьной информации. Обработка строковых переменных Стандартные директивы управления сегментами. Упрощенные директивы описания сегментов. Образы программ на диске и в памяти (формат .EXE и .COM). Использование сегментов данных дальнего типа. Функции BIOS для работы с экраном и клавиатурой</p>		
<p>ОПК.4 способность участвовать в настройке и наладке программно-аппаратных комплексов</p>	<p>з1. знать основы построения и архитектуры вычислительной техники</p>	<p>Ассемблирование и отладка готовых программ. Изучение системы команд, их кодирование, способы адресации операндов. Введение в язык ассемблера. Архитектурное своеобразие процессоров ПК. Характеристика типов данных в ПК IBM PC Представление чисел в формате с плавающей точкой. Команды передачи данных. Способы адресации операндов. Арифметические команды и машинная арифметика. Команды логических операций и сдвига. Операции над битами и байтами. Команды передачи управления и организации циклов. Команды обработки строк. Программирование типовых управляющих структур C/C++. Концепция модульного программирования. Процедуры, передача параметров между</p>	<p>Отчет по лабораторной работе, РГЗ.</p>	<p>Зачет, вопросы 1 – 28.</p>

		<p>программой и процедурой. Многомодульные программы и их ассемблирование. Макросы. Блоки повторения. Библиотеки макросов. Директива Include Настройка программной среды. Принципы взаимодействия между Turbo Assembler и Borland C++. Практические приёмы построения двуязычных программ. Процедура вызова из C++ ассемблерных функций. Передача параметров ассемблерным модулям. Соглашения о сохранении регистров. Возврат значений из ассемблерного модуля в программу C++. Вызов функций C++ из программы на ассемблере. Демонстрационная программа с использованием взаимных вызовов. Преобразование форматов числовых данных в операциях ввода-вывода Программирование алгоритмов сложных арифметических операций Программная модель IBM PC. Работа со стеком. Режимы адресации и форматы команд. Разработка программы на языке C++ с ассемблерным модулем. Синтаксис ассемблера. Основные этапы разработки и отладки программы на ассемблере. Объектный код программы. Структуры данных и их инициализация в программе. Директивы резервирования и инициализации данных. Макроопределения или директивы присваивания. Выражения и операторы. Система прерываний. Программные и аппаратные прерывания и процедуры их обработки. Прикладные обработчики прерываний Системные функции MS DOS ввода - вывода символьной информации. Обработка строковых переменных Стандартные директивы управления сегментами. Упрощенные директивы описания сегментов. Образы программ на диске и в памяти (формат .EXE и .COM). Использование сегментов данных дальнего типа. Функции BIOS для работы с экраном и клавиатурой</p>		
--	--	--	--	--

<p>ОПК.5 способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности</p>	<p>у8. владеть персональным компьютером как средством управления информацией</p>	<p>Ассемблирование и отладка готовых программ. Изучение системы команд, их кодирование, способы адресации операндов. Введение в язык ассемблера. Архитектурное своеобразие процессоров ПК. Характеристика типов данных в ПК IBM PC Представление чисел в формате с плавающей точкой. Команды передачи данных. Способы адресации операндов. Арифметические команды и машинная арифметика. Команды логических операций и сдвига. Операции над битами и байтами. Команды передачи управления и организации циклов. Команды обработки строк. Программирование типовых управляющих структур C/C++. Концепция модульного программирования. Процедуры, передача параметров между программой и процедурой. Многомодульные программы и их ассемблирование. Макросы. Блоки повторения. Библиотеки макросов. Директива Include Настройка программной среды. Принципы взаимодействия между Turbo Assembler и Borland C++. Практические приёмы построения двуязычных программ. Процедура вызова из C++ ассемблерных функций. Передача параметров ассемблерным модулям. Соглашения о сохранении регистров. Возврат значений из ассемблерного модуля в программу C++. Вызов функций C++ из программы на ассемблере. Демонстрационная программа с использованием взаимных вызовов. Преобразование форматов числовых данных в операциях ввода-вывода Программирование алгоритмов сложных арифметических операций Программная модель IBM PC. Работа со стеком. Режимы адресации и форматы команд. Разработка программы на языке C++ с ассемблерным модулем. Синтаксис ассемблера. Основные этапы разработки и отладки программы на ассемблере. Объектный код программы.</p>	<p>Отчет по лабораторной работе, РГЗ.</p>	<p>Зачет, вопросы 1 – 28.</p>
---	--	---	---	-------------------------------

		<p>Структуры данных и их инициализация в программе. Директивы резервирования и инициализации данных. Макроопределения или директивы присваивания. Выражения и операторы. Система прерываний. Программные и аппаратные прерывания и процедуры их обработки. Прикладные обработчики прерываний Системные функции MS DOS ввода - вывода символьной информации. Обработка строковых переменных Стандартные директивы управления сегментами. Упрощенные директивы описания сегментов. Образы программ на диске и в памяти (формат .EXE и .COM). Использование сегментов данных дальнего типа. Функции BIOS для работы с экраном и клавиатурой</p>		
<p>ПК.3/НИ готовность обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности</p>	<p>з2. знать современные компиляторы, отладчики и оптимизаторы программного кода</p>	<p>Ассемблирование и отладка готовых программ. Изучение системы команд, их кодирование, способы адресации операндов. Введение в язык ассемблера. Архитектурное своеобразие процессоров ПК. Характеристика типов данных в ПК IBM PC Представление чисел в формате с плавающей точкой. Команды передачи данных. Способы адресации операндов. Арифметические команды и машинная арифметика. Команды логических операций и сдвига. Операции над битами и байтами. Команды передачи управления и организации циклов. Команды обработки строк. Программирование типовых управляющих структур C/C++. Концепция модульного программирования. Процедуры, передача параметров между программой и процедурой. Многомодульные программы и их ассемблирование. Макросы. Блоки повторения. Библиотеки макросов. Директива Include Настройка программной среды. Принципы взаимодействия между Turbo Assembler и Borland C++. Практические приёмы построения двуязычных программ. Процедура вызова из C++ ассемблерных функций.</p>	<p>Отчет по лабораторной работе, РГЗ.</p>	<p>Зачет, вопросы 1 – 28.</p>

		<p>Передача параметров ассемблерным модулям. Соглашения о сохранении регистров. Возврат значений из ассемблерного модуля в программу С++. Вызов функций С++ из программы на ассемблере.</p> <p>Демонстрационная программа с использованием взаимных вызовов. Преобразование форматов числовых данных в операциях ввода-вывода</p> <p>Программирование алгоритмов сложных арифметических операций</p> <p>Программная модель IBM PC. Работа со стеком. Режимы адресации и форматы команд.</p> <p>Разработка программы на языке С++ с ассемблерным модулем. Синтаксис ассемблера. Основные этапы разработки и отладки программы на ассемблере.</p> <p>Объектный код программы. Структуры данных и их инициализация в программе.</p> <p>Директивы резервирования и инициализации данных.</p> <p>Макроопределения или директивы присваивания.</p> <p>Выражения и операторы.</p> <p>Система прерываний.</p> <p>Программные и аппаратные прерывания и процедуры их обработки. Прикладные обработчики прерываний</p> <p>Системные функции MS DOS ввода - вывода символьной информации. Обработка строковых переменных</p> <p>Стандартные директивы управления сегментами.</p> <p>Упрощенные директивы описания сегментов. Образы программ на диске и в памяти (формат .EXE и .COM).</p> <p>Использование сегментов данных дальнего типа.</p> <p>Функции BIOS для работы с экраном и клавиатурой</p>		
ПК.3/НИ	34. знать основы методологии создания программного обеспечения	<p>Ассемблирование и отладка готовых программ. Изучение системы команд, их кодирование, способы адресации операндов.</p> <p>Введение в язык ассемблера.</p> <p>Архитектурное своеобразие процессоров ПК.</p> <p>Характеристика типов данных в ПК IBM PC Представление чисел в формате с плавающей точкой. Команды передачи данных. Способы адресации операндов. Арифметические команды и машинная арифметика. Команды</p>	Отчет по лабораторной работе, РГЗ.	Зачет, вопросы 1 – 28.

		<p>логических операций и сдвига. Операции над битами и байтами. Команды передачи управления и организации циклов. Команды обработки строк. Программирование типовых управляющих структур С/С++. Концепция модульного программирования. Процедуры, передача параметров между программой и процедурой. Многомодульные программы и их ассемблирование. Макросы. Блоки повторения. Библиотеки макросов. Директива Include Настройка программной среды. Принципы взаимодействия между Turbo Assembler и Borland C++. Практические приёмы построения двуязычных программ. Процедура вызова из С++ ассемблерных функций. Передача параметров ассемблерным модулям. Соглашения о сохранении регистров. Возврат значений из ассемблерного модуля в программу С++. Вызов функций С++ из программы на ассемблере. Демонстрационная программа с использованием взаимных вызовов. Преобразование форматов числовых данных в операциях ввода-вывода</p> <p>Программирование алгоритмов сложных арифметических операций</p> <p>Программная модель IBM PC. Работа со стеком. Режимы адресации и форматы команд. Разработка программы на языке С++ с ассемблерным модулем. Синтаксис ассемблера. Основные этапы разработки и отладки программы на ассемблере. Объектный код программы. Структуры данных и их инициализация в программе. Директивы резервирования и инициализации данных. Макроопределения или директивы присваивания. Выражения и операторы. Система прерываний. Программные и аппаратные прерывания и процедуры их обработки. Прикладные обработчики прерываний</p> <p>Системные функции MS DOS ввода - вывода символьной информации. Обработка строковых переменных</p>		
--	--	--	--	--

		Стандартные директивы управления сегментами. Упрощенные директивы описания сегментов. Образы программ на диске и в памяти (формат .EXE и .COM). Использование сегментов данных дальнего типа. Функции BIOS для работы с экраном и клавиатурой		
--	--	--	--	--

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.1, ОПК.4, ОПК.5, ПК.3/НИ.

Зачет проводится в устной форме, по билетам.

Кроме того, сформированность компетенций 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.

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

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОПК.1, ОПК.4, ОПК.5, ПК.3/НИ, 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 «Машинные языки и программирование», 5 семестр

1. Методика оценки

Зачет проводится в устной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов с 1 по 14, второй вопрос из диапазона вопросов с 15 по 28 (список вопросов приведен ниже). В ходе зачета преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

Форма билета для зачета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет ИСТР

Билет № _____

к зачету по дисциплине «Машинные языки и программирование»

1. Программная модель 32-разрядных процессоров. Общие и специальные функции регистров.
2. Ассемблерная команда Loop организации цикла и цикл типа For в C++. Дополнительные команды организации циклов в ассемблере. Пример применения.

Утверждаю: зав. кафедрой _____ должность, ФИО
(подпись) _____ (дата)

2. Критерии оценки

- Ответ на билет для зачета считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, допускает принципиальные ошибки, оценка составляет *до 5 баллов*.
- Ответ на билет для зачета засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, допускает непринципиальные ошибки, оценка составляет *6 – 10 баллов*.
- Ответ на билет для зачета билет засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, дает характеристику процессов, проводит анализ причин, условий, может представить качественные характеристики процессов, оценка

составляет 11 – 15 баллов.

• Ответ на билет для зачета билет засчитывается на **продвинутом** уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, способен представить количественные характеристики определенных процессов, приводит конкретные примеры из практики, не допускает ошибок и способен обосновать выбор метода решения задач, оценка составляет 16 – 20 баллов.

3. Шкала оценки

Зачет считается сданным, если сумма баллов по всем заданиям билета оставляет не менее 5 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 «Машинные языки и программирование»

1. Представление числовых и символьных данных в ПК. Базовый одинарный формат для вещественных чисел.

2. Сегментированная модель памяти реального режима. Формирование исполнительного (физического) адреса.

3. Программная модель 32-разрядных процессоров. Общие и специальные функции регистров.

4. Функциональное назначение сегментных регистров в ассемблерных программах. Возможности переназначения. Организация стека.

5. Флаги условий и управления процессором. Характеристика команд с точки зрения их влияния на флаги.

6. Форматы команд базового процессора I8086 и основные принципы их двоичного кодирования. Виды адресации.

7. Структуры данных программы и их инициализация (размещение) в памяти.

8. Использование директив макроопределений EQU и =.

9. Операторы языка ассемблера и их использование в адресных и константных выражениях.

10. Стандартные директивы управления сегментами. Шаблоны программ .EXE и .COM.

11. Набор упрощенных директив управления сегментами. Шаблоны программ .EXE и .COM.

12. Организация программы с использованием сегментов дальнего типа.

13. Основные типы команд передачи данных. Примеры записи команд общего назначения для различных способов адресации операндов.

14. Команды загрузки эффективного адреса, табличной трансляции и преобразования форматов (CBW, CWD и CWDE).

15. Арифметические команды. Рассмотреть примеры выполнения команды сложения (вычитания) для конкретных операндов с целью определения состояния флагов результата. Сформулируйте понимание и реализацию принципа арифметики многократной точности.

16. Характеристика команд для выполнения логических операций и команд сдвига. Примеры типового использования команд.

17. Команды безусловной и условной передачи управления. Примеры ассемблерной записи команды JMP для различных типов переходов.

18. Перечень команд условных переходов с указанием условий перехода (состояний флагов).

19. Программирование на ассемблере управляющих структур языка C++:

20. Условный оператор if_else и его разновидности;

21. Логический переключатель switch-case.
22. Ассемблерная команда Loop организации цикла и цикл типа For в C++.
Дополнительные команды организации циклов в ассемблере. Пример применения.
23. Организация процедуры и ее интерфейс с основной программой. Способы передачи параметров между основной программой и процедурой.
24. Макросы. Определения. Примеры. Локальные метки в макросе.
25. Концепция модульного программирования. Организация программных модулей.
Директивы PUBLIC и EXTRN.
26. Система прерываний в компьютере. Аппаратные и программные средства, реализующие механизм прерываний. Процедура обработки прерываний в реальном режиме.
27. Резидентные программы и их организация.
28. Нерентабельность MS DOS и пути ее преодоления в обработчиках аппаратных прерываний.

Паспорт расчетно-графического задания

по дисциплине «Машинные языки и программирование», 5 семестр

1. Методика оценки

В рамках расчетно-графического задания по дисциплине студенты должны разработать программный продукт, протестировать его и оформить отчет по работе.

Требования к оформлению и содержанию отчета по РГЗ.

1. Отчет должен иметь титульный лист и состоять из 12 - 15 листов печатного текста шрифтом TNR 14 pt.

2. Содержание работы включает следующие обязательные разделы:

- Задание.
- Структуру программы и обоснование выбора ее отдельных модулей (составных обработчиков прерываний). При разработке структуры должны учитываться следующие факторы:
 - вычислительные процедуры, используемые в программе, должны соответствовать заданному алгоритму реализации;
 - резидентная программа должна обязательно включать прикладной обработчик клавиатурного прерывания `Int new_09h`, быть защищенной от повторной установки, обладать способностью самовыгрузки из памяти;
 - прикладной аппаратный обработчик прерываний должны строиться с учетом свойства нереентерабельности DOS;
 - программа должна иметь хорошо продуманный экранный интерфейс.
- Справочное описание используемых в программе системных обработчиков, сервисных функций DOS и BIOS.
- Функциональное описание программы и результаты ее демонстрации.
- Листинги программ.

Этапы выполнения работы соответствуют разделам пояснительной записки к РГЗ.

При защите оцениваются: корректность решения поставленной задачи, соответствие программной реализации вычислительных процедур разработанным алгоритмам, корректность функционирования программы, знание теоретического материала, использованного при выполнении работы.

2. Критерии оценки

- работа считается **не выполненной**, если студент не выполнил задание полностью или выполнил с очень существенными ошибками, оформил пояснительную записку, но имеются многочисленные замечания, не может объяснить используемые алгоритмы и технологии, оценка составляет 0 - 5 баллов.
- работа считается выполненной **на пороговом уровне**, если студент выполнил задание, но есть небольшие недочеты, оформил пояснительную записку, но имеются несущественные замечания, грамотно объясняет используемые алгоритмы и технологии, оценка составляет 6 - 10 баллов.

- работа считается выполненной **на базовом уровне**, если студент полностью выполнил задание, оформил пояснительную записку согласно требованиям, грамотно и полно объясняет используемые алгоритмы и технологии, оценка составляет 11 - 15 баллов.
- работа считается выполненной **на продвинутом уровне**, если студент полностью выполнил задание, применил оригинальные алгоритмы и программные подходы, оформил пояснительную записку согласно требованиям, грамотно и полно объясняет используемые алгоритмы и технологии, оценка составляет 16 - 20 баллов.

3. Шкала оценки

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

РГЗ считается сданным, если сумма баллов составляет не менее 5 баллов (из 20 возможных).

Для приведения соответствия баллов, традиционной оценки и буквенной оценки ECTS по дисциплине руководствуемся Положением о балльно-рейтинговой системе оценки достижений студентов НГТУ.

4. Примерный перечень тем РГЗ

1. Составьте программу извлечения квадратного корня положительного числа (≤ 65535) руководствуясь алгоритмом: квадратный корень из целого числа равен количеству последовательных нечетных чисел, которое можно из него вычесть.
2. Напишите программу для выдачи на экран сообщения вида "Вы получили за экзаменационную работу N баллов", где N - число от 1 до 127, двоичный эквивалент которого определен макроопределением (например, N = 99).
3. Массив знаковых чисел с именем Data_list включает N слов. Напишите программу, которая помещает наибольшее число в регистр AX.
4. Определить номер m (индекс) числа A в массиве из N слов. Значение числа A определить директивой равенства '='. Номер числа вернуть в регистре DL (CF=1). Если число не найдено, установить CF=0.
5. Напишите программу, которая подсчитывает число положительных, нулевых и отрицательных чисел в буфере Arrau, включающем 128 слов. Соответствующие значения поместить в переменные: Num_positive, Num_zerro, Num_negative.
6. Выполните реверсирование ASCII-строки длиной N байт. Для реверсирования используйте временный буфер в сегменте кода.
7. Выполнить преобразование ASCIIZ-строки (признаком конца строки является нулевой ASCII-символ), исключив из неё русские прописные буквы от А до Я (80h...9Fh).
8. Составьте программу вывода текста некоторой ASCII строки с некоторым атрибутом в точку "строка-столбец", используя непосредственное программирование видеобуфера. Длина строки не превышает 40 символов. Выход из программы - по нажатию клавиши.
9. Перевести положительное 6 разрядное восьмеричное ASCII-число из буфера Octo_buf (6 байт) в 16-разрядный двоичный код в AX (младший разряд числа располагается на месте старшего байта буфера). Установить флаг CF=1, если произошло переполнение.