

Документ подписан простой электронной подписью Информация о владельце: ФИО: Таскаев Сергей Валерьевич Должность: Ректор Дата подписания: 04.06.2025 12:39:43 Уникальный программный ключ: 04c19ed8bfb98f3b6cb77a486b9a8788b832237	МИНОВЕР НАУКИ РОССИИ Федеральное государственное бюджетное образовательное учреждение высшего образования «Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)	Рабочая программа дисциплины "Языки Ассемблера" по направлению подготовки (специальности) 10.05.01 "Компьютерная безопасность" направленности (профилю) специализация N 1 "Анализ безопасности компьютерных систем" ФГБОУ ВО «ЧелГУ»	стр. 1
---	--	--	--------

**Рабочая программа дисциплины (модуля)*
Языки Ассемблера**

Направление подготовки (специальность)

10.05.01 Компьютерная безопасность

Направленность (профиль)

специализация N 1 "Анализ безопасности компьютерных систем"

Присваиваемая квалификация (степень)

специалист по защите информации

Форма обучения

очная

Год(ы) набора 2025

*Рабочая программа дисциплины (модуля) адаптирована для инклюзивного обучения инвалидов и лиц с ограниченными возможностями здоровья

Челябинск 2025 г.



Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре ОПОП
3. Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля)
4. Объем дисциплины (модуля)
5. Структура и содержание дисциплины (модуля)
6. Фонд оценочных средств
 - 6.1. Перечень видов оценочных средств
 - 6.2. Типовые контрольные задания и иные материалы для текущей аттестации
 - 6.3. Типовые контрольные вопросы и задания для промежуточной аттестации
 - 6.4. Критерии оценивания
7. Учебно-методическое и информационное обеспечение дисциплины (модуля)
 - 7.1. Рекомендуемая литература
 - 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"
 - 7.3. Перечень информационных технологий
8. Материально-техническое обеспечение дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Специальные условия освоения дисциплины обучающимися с инвалидностью и ограниченными возможностями здоровья



1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Изучение программной и системной архитектуры современных процессоров и умение создавать код любой сложности с использованием языков и систем программирования низкого уровня.

Результаты обучения по дисциплине направлены на достижение индикаторов:

ОПК-7.1 Знает общие принципы построения, области и особенности применения языков программирования высокого и низкого уровня; язык программирования высокого и низкого уровня (объектно-ориентированное программирование); знает язык ассемблера персонального компьютера.

ОПК-7.2 Умеет работать с интегрированной средой разработки программного обеспечения; разрабатывать и реализовывать на языке высокого и низкого уровня алгоритмы решения типовых профессиональных задач;

ОПК-7.3 Владеет навыками разработки, документирования, тестирования и отладки программ.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Цикл (раздел) ОПОП: Б1.О.13

2.1 Требования к предварительной подготовке обучающегося:

Обучающиеся должны владеть знаниями и навыками в области информатики и программирования, получаемыми в рамках следующих дисциплин:

Языки программирования

Аппаратные средства вычислительной техники

Информатика

2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:

Данная дисциплина является предшествующей для следующих дисциплин:

Системное программирование

Защита программ и данных

Исследование вредоносного программного обеспечения

Анализ уязвимостей программного обеспечения

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

ОПК-7: Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ;

Знать:

- специфику создания низкоуровневого кода под современные процессоры.
- синтаксис языков ассемблера для современных процессоров.

Уметь:

- проектировать программное обеспечение с учётом низкоуровневой специфики архитектуры современных процессоров.
- создавать код любой сложности под современные процессоры.

Владеть:

- навыки сопряжения низкоуровневого кода с другими программными системами.
- навыками использования инструментальных средств создания кода под современные процессоры.

В результате освоения дисциплины обучающийся должен

3.1 Знать:

- 3.1.1 – системную и программную архитектуру современных процессоров (x86, x64, ARM, MIPS, AVR);
- 3.1.2 – инструментальные средства создания низкоуровневого кода.

3.2 Уметь:

- 3.2.1 – проектировать и создавать низкоуровневый программный код.



3.3 Владеть:

3.3.1 – навыками использования компиляторов и компоновщиков для создания низкоуровневого кода.

4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость	7 ЗЕТ
Часов по учебному плану : 252 в том числе : аудиторные занятия : 136 самостоятельная работа : 71,1 часов на контроль : 27 контактная работа: 153,9 ИКР: 17,9	Виды контроля в семестрах: экзамены 5 зачеты 4

5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Литература
	Раздел 1. Понятие об архитектуре вычислительной машины. Архитектура фон Неймана.			
1.1	Архитектура фон Неймана. Понятие системы команд центрального процессора. Состав и назначение регистров центрального процессора Intel 80x86. Слово состояния процессора. Схема формирования исполнительного адреса памяти. Директивы сегментной организации памяти в ассемблере masm86. /Лек/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
1.2	Понятие об архитектуре вычислительной машины. Архитектура фон Неймана. /Ср/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1
	Раздел 2. Типы данных используемых в IBM PC			
2.1	Числовые и символьные типы данных. Символьный тип. Основная и дополнительная таблицы кодов ASCII. Форматы целых чисел: целое слово, короткое целое, длинное целое. Представление отрицательных чисел. Форматы вещественных чисел: короткое вещественное, длинное вещественное, временное вещественное. Двоично-десятичный кодированный формат- упакованный и неупакованный. /Лек/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
2.2	Практическое ознакомление с типами данных используемых в IBM PC /Лаб/	4	10	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
2.3	Типы данных используемых в IBM PC /Ср/	4	3	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
	Раздел 3. Режимы адресации в системе команд IBM PC			
3.1	Перечень режимов адресации данных в системе команд IBM PC. Схемы формирования исполнительного адреса памяти. Соглашения об использовании сегментных регистров при адресации памяти в программах на MASM86. Режимы адресации переходов в системе команд IBM PC Схемы формирования исполнительного адреса перехода. /Лек/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
3.2	Режимы адресации в системе команд IBM PC. /Ср/	4	3	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
	Раздел 4. Машинные форматы команд IBM PC			
4.1	Понятие формата машинной команды. Форматы команд IBM PC. Общий двухоперандный формат. Формат двухоперандной команды с непосредственным операндом. Общий однооперандный формат. Укороченные форматы /Лек/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2



4.2	Практическое ознакомление с машинными форматами команд IBM PC /Лаб/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
4.3	Машинные форматы команд IBM PC /Ср/	4	3,1	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
Раздел 5. Система команд IBM PC				
5.1	Команды загрузки пересылки и соответствующие им цепочечные команды. Арифметические команды. Установка флагов после арифметических операций. Логические команды. Команды сдвигов. Команды ветвления условные и безусловные. Команды циклов. Команды ввода-вывода. Команды работы с подпрограммами. Команды программных прерываний. Команды особых регистровых операций и преобразования длины. Команды преобразования формата. /Лек/	4	14	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
5.2	Практическое ознакомление с системами команд IBM PC /Лаб/	4	14	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
5.3	Машинные форматы команд IBM PC /Ср/	4	4	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
Раздел 6. Основные сведения об ассемблере MASM86				
6.1	Основные элементы языка ассемблера - операторы, команды макрокоманды, метки. Понятие о каждом из элементов языка макроассемблера. /Лек/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
6.2	Основные сведения об ассемблере MASM86 /Ср/	4	4	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
Раздел 7. Данные в текстах на Ассемблере MASM86				
7.1	Данные в текстах на MASM86: константы. Данные в текстах на MASM86: ячейки памяти, записи. Операторы mask и width. Данные в текстах на MASM86: структуры. /Лек/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
7.2	Данные в текстах на Ассемблере MASM86 /Ср/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
Раздел 8. Операторы в текстах на Ассемблере MASM86				
8.1	Операторы в текстах на Ассемблере MASM86: операторы действий. Атрибутные операторы, возвращающие и присваивающие значения. /Лек/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
8.2	Практическое ознакомление с операторами в текстах на Ассемблере MASM86. /Лаб/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
8.3	Операторы в текстах на Ассемблере MASM8 /Ср/	4	4	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
Раздел 9. Процедуры в текстах на Ассемблере MASM86				
9.1	Использование процедур в программах на ассемблере. Способы передачи параметров. /Лек/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
9.2	Практическое ознакомление с процедурами в текстах на Ассемблере MASM86. /Лаб/	4	2	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
9.3	Процедуры в текстах на Ассемблере MASM86 /Ср/	4	4	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2



	Раздел 10. Средства макрогенерации и условной генерации MASM86			
10.1	Операторы, обеспечивающие макрогенерацию. Понятие макроопределения, макровызова, макроподстановки, макрорасширения. Отличие макрокоманды от процедуры. Сравнение макрокоманды и процедуры, Встроенные макро-определения MASM86. Условные операторы и оператор альтернативы. /Лек/	4	4	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
10.2	Практическое ознакомление с средствами макрогенерации и условной генерации MASM86 /Лаб/	4	4	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
10.3	Средства макрогенерации и условной генерации MASM86 /Ср/	4	4	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
	Раздел 11. Иная контактная работа			
11.1	Зачёт /Зачёт/	4	0	Л1.1 Л1.2Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
11.2	Иная контактная работа: индивидуальные консультации, текущий контроль. /ИКР/	4	6,9	
	Раздел 12. Процессоры архитектуры IA-32			
12.1	Архитектура процессоров семейства IA-32 /Лек/	5	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3 Л2.4Л3.1 Э1 Э2
12.2	Архитектура процессоров семейства IA-32 /Лаб/	5	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3 Л2.4Л3.1 Э1 Э2
12.3	Архитектура процессоров семейства IA-32 /Ср/	5	8	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3 Л2.4Л3.1 Э1 Э2
12.4	Создание кода для процессоров семейства IA-32 /Лек/	5	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
12.5	Создание кода для процессоров семейства IA-32 /Лаб/	5	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
12.6	Создание кода для процессоров семейства IA-32 /Ср/	5	5	Л1.1 Л1.2 Л1.3Л2.1 Л2.3 Л2.4Л3.1 Э1 Э2
	Раздел 13. Процессоры архитектуры Intel 64			
13.1	Архитектура процессоров семейства Intel 64 /Лек/	5	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3 Л2.4 Э1 Э2
13.2	Архитектура процессоров семейства Intel 64 /Лаб/	5	4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3 Л2.4 Э1 Э2
13.3	Архитектура процессоров семейства Intel 64 /Ср/	5	5	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3 Л2.4 Э1 Э2



13.4	Создание кода для процессоров семейства Intel 64 /Лек/	5	6	Л1.1 Л1.2 Л1.3Л2.1 Л2.3 Л2.4 Э1 Э2
13.5	Создание кода для процессоров семейства Intel 64 /Лаб/	5	6	Л1.1 Л1.2 Л1.3Л2.1 Л2.3 Л2.4 Э1 Э2
13.6	Создание кода для процессоров семейства Intel 64 /Ср/	5	5	Л1.1 Л1.2 Л1.3Л2.1 Л2.3 Л2.4 Э1 Э2
Раздел 14. RISC процессоры				
14.1	Архитектура RISC-процессоров ARM, MIPS, AVR. /Лек/	5	8	Л1.1Л2.3 Л2.4 Э1 Э2
14.2	Архитектура RISC-процессоров ARM, MIPS, AVR. /Лаб/	5	8	Л1.1Л2.3 Л2.4 Э1 Э2
14.3	Архитектура RISC-процессоров ARM, MIPS, AVR. /Ср/	5	8	Л1.1Л2.3 Л2.4 Э1 Э2
14.4	Создание кода /Лек/	5	8	Л1.1Л2.3 Л2.4 Э1 Э2
14.5	Создание кода /Лаб/	5	8	Л1.1Л2.3 Л2.4 Э1 Э2
14.6	Создание кода /Ср/	5	7	Л1.1Л2.3 Л2.4 Э1 Э2
Раздел 15. Иная контактная работа				
15.1	Экзамен /Экзамен/	5	27	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3 Л2.4Л3.1 Э1 Э2
15.2	Иная контактная работа: индивидуальные консультации, текущий контроль. /ИКР/	5	11	

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

6.1. Перечень видов оценочных средств

Зачетные задания.
Домашние задания.
Аудиторные задания.
Вопросы на зачёте.
Задания на зачёте.
Вопросы на экзамене.
Задания на экзамене.

6.2. Типовые контрольные задания и иные материалы для текущей аттестации

Темы для домашних, аудиторных и зачетных заданий:

1. Программный интерфейс DOS.
2. Программный интерфейс BIOS.
3. Ассемблер `masm`.
4. Система команд процессоров архитектуры x86.
5. Система команд процессоров архитектуры IA-32.
6. Система команд процессоров архитектуры Intel 64.
7. Арифметический сопроцессор.
8. SIMD-расширения процессоров Intel.
9. WinAPI.
10. ABI операционных систем Windows NT и Linux.
11. Система команд процессоров архитектуры ARM.
12. Система команд процессоров архитектуры MIPS.



13. Система команд процессоров архитектуры AVR.
14. Работа с динамическими структурами данных на языке ассемблера.

Примеры домашних, аудиторных и зачетных заданий:

1. Копирование файлов через интерфейс DOS.
2. Алгоритм рисования линии через видеобуфер.
3. Реализовать эмулятор для собственной простой архитектуры процессора.
4. Реализовать алгоритмы обработки данных (сортировка, поиск, контейнеры и др.) на языке ассемблера.
5. Реализовать на языке ассемблера стандартные функции работы со строками.
6. Реализовать арифметику больших чисел.
7. Реализовать замену символа в строке.
8. Реализовать замену строк.
9. Реализовать сортировку массива строк.
10. Реализовать копирование файла.
11. Реализовать замену строк в файле.
12. Реализовать разбиение строки на подстроки по подстроке.
13. Инвертирование массива чисел.
14. Сложение элементов массивов.
15. Поиск элементов массива.
16. Реализовать стандартные строковые функции с помощью строковых инструкций.
17. Реализовать динамические структуры данных и программы для их тестирования.
18. Реализовать на языке ассемблера программы с графическим интерфейсом.
19. Реализовать вычисление интеграла через интегральные суммы.
20. Реализовать функцию вывода на экран в десятичном виде вещественного числа в 4- и 8-байтовом формате.
21. Реализовать функцию возведения в степень с помощью инструкций f2xm1, fyl2x, fscale.
22. Нарисовать график функции x в произвольной целой степени.
23. Реализовать программу отображения графиков функций с использованием арифметического сопроцессора.
24. Реализовать программу для RISC-процессора разбора арифметических выражений.
25. На языке ассемблера для процессоров ARM32 написать функцию, обрабатывающую массивы указателей.
26. На языке ассемблера для процессоров ARM32 вывести на экран файл, имя которого передано в качестве аргумента программы.
27. На языке ассемблера для процессоров ARM32 реализовать копирование файлов.

6.3. Типовые контрольные вопросы и задания для промежуточной аттестации

Темы заданий на экзамене:

1. Система команд 16-ти разрядных процессоров архитектуры x86.
2. Система команд 32-х разрядных процессоров архитектуры x86.
3. Программный интерфейс DOS.
4. Программный интерфейс BIOS.
5. Ассемблер `masm`.

Примеры заданий на экзамене:

1. Копирование файлов через интерфейс DOS.
2. Алгоритм рисования линии через видеобуфер.
3. Реализовать алгоритмы обработки данных (сортировка, поиск, контейнеры и др.) на языке ассемблера.
4. Реализовать на языке ассемблера стандартные функции работы со строками.
5. Реализовать арифметику больших чисел.
6. Реализовать замену символа в строке.
7. Реализовать замену строк.
8. Реализовать сортировку массива строк.
9. Реализовать копирование файла.
10. Реализовать замену строк в файле.
11. Реализовать разбиение строки на подстроки по подстроке.
12. Инвертирование массива чисел.
13. Сложение элементов массивов.
14. Поиск элементов массива.
15. Реализовать стандартные строковые функции с помощью строковых инструкций.
16. Реализовать динамические структуры данных и программы для их тестирования.



Примеры вопросов на экзамене:

1. Программное окружение 16-ти разрядных процессоров архитектуры x86.
2. Система команд 16-ти разрядных процессоров архитектуры x86.
3. Программный интерфейс DOS.
4. Программный интерфейс BIOS.
5. Синтаксис ассемблера `masm`.
6. Макросредства ассемблера `masm`.

Темы заданий на зачёте:

1. Программный интерфейс DOS.
2. Программный интерфейс BIOS.
3. Ассемблер `masm`.
4. Система команд процессоров архитектуры x86.
5. Система команд процессоров архитектуры IA-32.
6. Система команд процессоров архитектуры Intel 64.
7. Арифметический сопроцессор.
8. SIMD-расширения процессоров Intel.
9. WinAPI.
10. ABI операционных систем Windows NT и Linux.
11. Система команд процессоров архитектуры ARM.
12. Система команд процессоров архитектуры MIPS.
13. Система команд процессоров архитектуры AVR.
14. Работа с динамическими структурами данных на языке ассемблера.

Примеры заданий на зачёте:

1. Реализовать алгоритмы обработки данных (сортировка, поиск, контейнеры и др.) на языке ассемблера.
2. Реализовать на языке ассемблера стандартные функции работы со строками.
3. Реализовать арифметику больших чисел.
4. Реализовать замену символа в строке.
5. Реализовать замену строк.
6. Реализовать сортировку массива строк.
7. Реализовать копирование файла.
8. Реализовать замену строк в файле.
9. Реализовать разбиение строки на подстроки по подстроке.
10. Инвертирование массива чисел.
11. Сложение элементов массивов.
12. Поиск элементов массива.
13. Реализовать стандартные строковые функции с помощью строковых инструкций.
14. Реализовать динамические структуры данных и программы для их тестирования.
15. Реализовать на языке ассемблера программы с графическим интерфейсом.
16. Реализовать вычисление интеграла через интегральные суммы.
17. Реализовать функцию вывода на экран в десятичном виде вещественного числа в 4- и 8-байтовом формате.
18. Реализовать функцию возведения в степень с помощью инструкций `f2xm1`, `fyl2x`, `fscale`.
19. Нарисовать график функции x в произвольной целой степени.
20. Реализовать программу отображения графиков функций с использованием арифметического сопроцессора.
21. Реализовать программу для RISC-процессора разбора арифметических выражений.
22. На языке ассемблера для процессоров ARM32 написать функцию, обрабатывающую массивы указателей.
23. На языке ассемблера для процессоров ARM32 вывести на экран файл, имя которого передано в качестве аргумента программы.
24. На языке ассемблера для процессоров ARM32 реализовать копирование файлов.

Примеры вопросов на зачёте:

1. Программный интерфейс процессоров архитектуры IA-32.
2. Система команд процессоров архитектуры IA-32.
3. Программный интерфейс процессоров архитектуры Intel 64.
4. Система команд процессоров архитектуры Intel 64.
5. Программный интерфейс арифметического сопроцессора
6. Расширения MMX, SSE, AVX, AES процессоров Intel.
7. Реализация функций на языке ассемблера. Соглашения о вызовах.



8. Программный интерфейс процессоров архитектуры ARM.
9. Система команд процессоров архитектуры ARM.
10. Программный интерфейс процессоров архитектуры MIPS.
11. Система команд процессоров архитектуры MIPS.
12. Программный интерфейс процессоров архитектуры AVR.
13. Система команд процессоров архитектуры AVR.

6.4. Критерии оценивания

За своевременное и самостоятельное выполнение учебных работ в течение семестра студент получает рейтинговые баллы. Сумма за выполнение основных заданий в полном объёме - 100. Сверх этой суммы могут начисляться баллы за выполнение дополнительных заданий.

Пропуск по неубажительной причине одной пары влечет вычет 1 балла из итоговой суммы за семестр.

При нехватке баллов преподавателем может быть предоставлено дополнительное задание или возможность доделать задание, в котором была оценена не вся функциональность.

Основные баллы выставляются за выполнение объемных зачётных заданий, которые выполняются дома и сдаются в течение семестра. Сумма в 100 баллов делится между зачётными заданиями (не обязательно равномерно). При выдаче зачётного задания определено сколько баллов выставляется за реализацию определённой функциональности. Для зачётных заданий может быть определена функциональность повышенной сложности, за выполнение которой выставляются дополнительные баллы. Также дополнительные баллы могут быть выставлены по усмотрению преподавателя за особо примечательную реализацию.

При сдаче зачётного задания производится опрос по техническим деталям реализации и по теории, используемой при выполнении заданий. Неудовлетворительный ответ будет означать несамостоятельность выполнения задания, что влечёт выставление 0 баллов за соответствующую функциональность. Если для одного задания это повторяется более 2 раз, то за всё задание выставляется 0 баллов без возможности повторной сдачи.

Выполнение заданий предполагает некоторую программную реализацию, к которой будут предъявляться обычные требования по качеству кода. Код должен быть удобочитаемым, хорошо структурированным, написанным в едином стиле. Иначе возможна сбавка до 5 баллов. За программные ошибки, приводящие к работоспособности кода не для всех возможных случаев, возможна сбавка до 10 баллов (в зависимости от критичности ошибки). За программные ошибки, приводящие к аварийному некорректному завершению программы, возможна сбавка до 10 баллов (в зависимости от критичности ошибки). По пройденному материалу выдаются небольшие задания для выполнения дома и/или во время семинарских занятий. За эти задания выставляются небольшие дополнительные баллы. Сдавать их можно либо в день выдачи либо на следующем занятии. Домашние и аудиторные задания – это небольшие задания, за которые обычно выставляется 1-2 балла. Они оцениваются атомарно: либо задание выполнено (выставляется указанное при выдаче задания количество баллов), либо не выполнено (0 баллов).

Итоговая оценка за дисциплину выставляется по результатам выполнения заданий текущего контроля. При необходимости во время зачёта и экзамена может быть предоставлена возможность получить дополнительные баллы (не более 20), выполнив дополнительные задания и ответив (в устной форме) на вопросы.

Дополнительные задания, выдаваемые на зачёте и экзамене, являются относительно объемными, за них выставляется до 10-15 баллов. Поэтому к ним применимы описанные выше критерии оценивания зачётных заданий с соответствующей корректировкой баллов: сбавка за некорректную работу до 5 баллов, за аварийное завершение – до 5 баллов.

Ответ на зачёте и экзамене оценивается по трём параметрам.

1. Построение ответа (структура ответа, грамотность речи, последовательность и т.д.).

Студент самостоятельно правильно выстраивает структуру ответа, изложение последовательное, речь грамотная без оговорок. 2 балла

Изложение студента непоследовательное и обрывочное, взаимосвязи частей ответа не всегда прослеживаются. Раскрытие сути ответа невозможно без уточняющих вопросов. 1 балл.

Студент испытывает существенные трудности при самостоятельном построении ответа, способен только давать краткие ответы на конкретные вопросы. 0 баллов.

2. Фактическая полнота ответа.

Студент правильно ответил на теоретический вопрос билета. Показал отличные знания в рамках усвоенного учебного материала. Ответил на все дополнительные вопросы. 4 балла.

Студент ответил на теоретический вопрос билета с небольшими неточностями. Показал хорошие знания в рамках усвоенного учебного материала. Ответил на большинство дополнительных вопросов. 3 балла.

Студент ответил на теоретический вопрос билета с существенными неточностями. Показал удовлетворительные знания в рамках усвоенного учебного материала. При ответах на дополнительные вопросы



было допущено много неточностей. 2 балла.

При ответе на теоретический вопрос билета студент продемонстрировал недостаточный уровень знаний. При ответах на дополнительные вопросы было допущено множество неправильных ответов. 1 балл.

Студент не ответил на вопрос. 0 баллов.

3. Собственный анализ излагаемого материала его оценка в контексте взаимодействия с другими областями, умение применять на практике.

Студент ясно осознаёт место излагаемого материала в общей структуре профессионального знания, знает стандартные примеры использования и предлагает свои, даёт собственные компетентные оценки. 4 балла.

Студент осознаёт взаимосвязи и знает стандартные примеры использования. Допускает неточности при самостоятельном анализе. 3 балла.

Студент в общих чертах осознаёт взаимосвязи и знает стандартные примеры использования. При проведении самостоятельного анализа нуждается в уточняющих вопросах, при этом допускает существенные неточности. 2 балла.

Студент знает основные стандартные примеры использования. Не осознаёт взаимосвязей с другими областями. 1 балл.

Студент не осознаёт взаимосвязей и практическое приложение излагаемого материала. 0 баллов.

Перевод рейтинговых баллов в оценки за зачет:

61 и более баллов – зачтено;

60 и менее баллов – не зачтено.

Перевод рейтинговых баллов в оценки за экзамен:

91 и более баллов – отлично;

76 - 90 баллов – хорошо;

61 - 75 баллов – удовлетворительно;

60 и менее баллов – неудовлетворительно.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1. Рекомендуемая литература

7.1.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л1.1	Рудаков П. И., Финогенов К. Г.	Язык ассемблера: уроки программирования: практическое пособие (https://biblioclub.ru/index.php?page=book&id=89393)	Москва : Диалог-МИФИ, 2001	ЭБС
Л1.2	Пильщиков В. Н.	Программирование на языке ассемблера IBM PC: учебное пособие (https://biblioclub.ru/index.php?page=book&id=447687)	Москва : Диалог-МИФИ, 2014	ЭБС
Л1.3	Аблязов Р. З.	Программирование на ассемблере на платформе x86-64 (http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=1273)	Москва : ДМК Пресс, 2011	ЭБС

7.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л2.1	Довгий П. С., Поляков В. И.	Прикладная архитектура базовой модели процессора Intel (http://e.lanbook.com/books/element.php?pl1_id=43560)	Санкт-Петербург : НИУ ИТМО, 2012	ЭБС
Л2.2	Федюшкин П. П.	Организация и функционирование виртуальной памяти ЭВМ: практическое пособие (https://biblioclub.ru/index.php?page=book&id=141461)	Москва : Лаборатория книги, 2011	ЭБС
Л2.3	Громов Ю. Ю., Иванова О. Г., Серегин М. Ю., Ивановский М. А., Дидрих В. Е.	Архитектура ЭВМ и систем: учебное пособие (https://biblioclub.ru/index.php?page=book&id=277352)	Тамбов : Тамбовский государственный технический университет (ТГТУ), 2012	ЭБС



	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л2.4	Лисицин Д.В.	Программирование на языке ассемблера: учебное пособие (https://znanium.com/catalog/document?id=396959)	Новосибирск : Новосибирский государственный технический университет (НГТУ), 2018	ЭБС

7.1.3. Методические разработки

	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л3.1	Скороход С. В., Селянкин В. В., Дроздов С. Н., Калачев Д. П., Хусаинов Н. Ш.	Основы программирования микропроцессоров Intel для встраиваемых систем: учебное пособие (https://biblioclub.ru/index.php?page=book&id=493316)	Таганрог : Южный федеральный университет, 2016	ЭБС

7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Лань [Электронный ресурс] : электронно-библиотечная система (ЭБС) / издательство Лань. – URL: http://e.lanbook.com/ .
Э2	Университетская библиотека онлайн [Электронный ресурс] : электронно-библиотечная система (ЭБС) / ООО Директмедиа Паблишинг. – URL: http://biblioclub.ru/ .

7.3 Перечень информационных технологий

7.3.1 Программное обеспечение

Visual Studio

VirtualBox

Adobe Reader

Notepad++

7.3.2 Профессиональные базы данных и информационно-справочные системы

1. Электронный каталог научной библиотеки ЧелГУ [Электронный ресурс] : база данных / Челяб. гос. ун-т. – Челябинск, 1992 – .
2. MSDN Library [Электронный ресурс]. URL: <https://msdn.microsoft.com/>.

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для реализации дисциплины используются учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы.

Учебные аудитории укомплектованы специализированной мебелью и техническими средствами обучения: проектором, экраном, магнитно-маркерной доской, маркером, персональными компьютерами; с возможностью демонстрации электронных презентаций при уровне освещения, достаточном для работы с конспектом.

Для проведения занятий лекционного типа имеется демонстрационное оборудование: проектор, экран.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с подключением к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для изучения дисциплины предусмотрены проведение лекционных и лабораторных занятий, а также самостоятельная работа студентов.

Лекционные занятия обеспечивают теоретическое изучение дисциплины. Основными методами обучения являются информационно-объяснительный и проблемный. На лекциях излагается основное содержание тем программы, проводится анализ основных понятий и методов, разбираются примеры программного кода, демонстрируются приемы создания низкоуровневого программного обеспечения, обсуждаются возможные приложения изложенных методов.

Лабораторные занятия служат для закрепления теоретических основ, излагаемых в лекциях, и получения практического опыта создания низкоуровневого кода. Основные методы: репродуктивный и частично-поисковый. Для проведения текущего промежуточного контроля на каждом лабораторном занятии выдаются небольшие



практические задания, рассчитанные на выполнение в течение занятия, и домашние задания.
Для самостоятельной работы студентам следует использовать методические материалы, имеющиеся в Научной библиотеке ЧелГУ, а также выложенные на сайте математического факультета ЧелГУ и на сайте кафедры компьютерной безопасности и прикладной алгебры. Для студентов проводятся индивидуальные консультации, каждому студенту при необходимости могут быть выданы индивидуальные задания для самостоятельной работы, позволяющие углубленно изучить отдельные темы дисциплины.

Стоит особо подчеркнуть, что курс носит практический характер. Основным результатом освоения курса является умение применять изученные методы на практике. Поэтому для успешного освоения курса обязательно полное выполнение всех практических заданий.

В случае применения при обучении дисциплины электронного обучения, дистанционных образовательных технологий общение обучающихся и преподавателя осуществляется в режиме реального времени (онлайн-лекции (вебинары), чаты, видео-конференции и др.) или отложенного времени (система дистанционного обучения Moodle, форумы, электронная почта и др.).

Большую часть времени обучающиеся самостоятельно работают с учебно-методическими материалами. Студенты имеют возможность консультироваться с преподавателем по всем вопросам, возникающим в ходе самостоятельной работы посредством электронной почты, мессенджеров, социальных сетей и т.п.

Доступ обучающегося к учебным ресурсам в режиме отложенного времени, самостоятельной работы осуществляется через сеть Интернет в удобном для него месте, времени и темпе.

При обучении лиц с ограниченными возможностями здоровья электронное обучение, дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах.

Реализация дисциплины с применением электронного обучения, дистанционных образовательных технологий (далее – ЭО, ДОТ) осуществляется на основании «Положения о реализации основных и дополнительных образовательных программ с применением электронного обучения и дистанционных образовательных технологий в федеральном государственном бюджетном образовательном учреждении высшего образования «Челябинский государственный университет», «Положения о порядке зачета обучающимися по основным профессиональным образовательным программам высшего образования в ФГБОУ ВО «ЧелГУ» результатов освоения в организациях, осуществляющих образовательную деятельность, учебных предметов, курсов, дисциплин (модулей), практик, дополнительных образовательных программ» посредством электронной информационно-образовательной среды ФГБОУ ВО «ЧелГУ». В исключительных случаях (форс-мажор и т.п.) при реализации образовательной деятельности с применением ЭО, ДОТ могут применять компоненты, не входящие в перечень электронной информационно-образовательной среды.

10. СПЕЦИАЛЬНЫЕ УСЛОВИЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ ОБУЧАЮЩИМИСЯ С ИНВАЛИДНОСТЬЮ И ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Освоение дисциплины инвалидами и лицами с ограниченными возможностями здоровья осуществляется с использованием специальных технических средств и информационных технологий, предоставляемых Ресурсным учебно-методическим центром по обучению инвалидов и лиц с ограниченными возможностями здоровья ЧелГУ по запросу обучающегося (мобильные специальные технические средства для лиц с нарушениями зрения и с нарушением слуха, ассистивные информационные технологии).

При необходимости для обучающихся с нарушениями зрения на рабочих местах для проведения практических или лабораторных занятий устанавливается специальное программное обеспечение (программа речевой навигации, речевые синтезаторы, экранные лупы).

В учебные аудитории обеспечивается беспрепятственный доступ для обучающихся с инвалидностью и с ограниченными возможностями здоровья. В каждой аудитории, где обучаются инвалиды и лица с ограниченными возможностями здоровья, предусматривается соответствующее количество мест для обучающихся с учетом нарушений их здоровья.

Для освоения дисциплины инвалидам и лицам с ограниченными возможностями здоровья предоставляется доступ к печатным источникам, имеющимся в научной библиотеке ЧелГУ, с помощью специальных технических средств; доступ с помощью специальных технических и программных средств к электронным источникам, представленным в форме электронного документа в фонде научной библиотеки ЧелГУ или электронно-библиотечных системах.

Учебно-методические материалы для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья предоставляются в формах, адаптированных к ограничениям их здоровья и особенностям восприятия информации.

Для инвалидов и лиц с ограниченными возможностями здоровья освоение дисциплины может быть частично или полностью осуществлено с использованием дистанционных образовательных технологий.

При проведении промежуточной аттестации по дисциплине обучающимся с инвалидностью и с ограниченными



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Рабочая программа дисциплины "Язык Ассемблера" по направлению подготовки (специальности) 10.05.01
"Компьютерная безопасность" направленности (профилю) специализация N 1 "Анализ безопасности
компьютерных систем" ФГБОУ ВО «ЧелГУ»

стр. 14

возможностями здоровья обеспечивается по их заявлению предоставление в доступной форме в зависимости от их индивидуальных особенностей инструкции о порядке проведения промежуточной аттестации, оценочных средств и возможности ответов на задания (письменно на бумаге, набор ответов на компьютере, письменно шрифтом Брайля, с использованием услуг ассистента, устно).

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование предоставленных ЧелГУ или собственных технических средств, необходимых им в связи с их индивидуальными особенностями. При необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на задания, процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

