

МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ))



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
высшего образования - программы бакалавриата
по направлению подготовки
09.03.01 Информатика и вычислительная техника,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Низкоуровневые языки

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность (профиль): Вычислительные системы и сети

Форма обучения: Очная

Рабочая программа дисциплины (модуля) в виде
электронного документа выгружена из единой
корпоративной информационной системы управления
университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)
ID подписи: 4196
Подписал: заведующий кафедрой Желенков Борис
Владимирович
Дата: 15.03.2024

1. Общие сведения о дисциплине (модуле).

Целями освоения дисциплины являются:

- изучение архитектуры процессоров семейства x86 на уровне программной модели, регистровых структур процессора и его системы команд;
- приобретение навыков и умений использования инструментальных средств под-готовки и отладки низкоуровневых программ;
- приобретение опыта реализации типовых алгоритмов командами процессора.

Задачами дисциплины (модуля) являются:

- получение студентами устойчивых представлений о принципах функционирования процессора, его взаимодействия с памятью, понятии системы команд процессора, формата команд;
- формирование навыков разработки машинно-ориентированных программ на сим-волическом языке ассемблера (транслятора)

2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

ОПК-8 - Способен разрабатывать алгоритмы и программы, пригодные для практического применения;

ОПК-9 - Способен осваивать методики использования программных средств для решения практических задач;

ПК-2 - Способность разрабатывать компоненты системных программных продуктов.

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

Знать:

- основы архитектуры, систему и форматы команд процессоров архитектуры x86;
- конструкции символического языка ассемблера;
- структуры ассемблерных программ, принципы трансляции и дизассемблирования.

Уметь:

- проектировать содержательные и детальные алгоритмы низкоуровневых программ;

- использовать команды процессора для реализации линейной, разветвленной и циклической логики решения задач;
- использовать программные сервисы операционной системы в ассемблерных программах.

Владеть:

- технологией и инструментальными средствами трансляции, компоновки и отладки ассемблерной программы;
- навыками разработки ассемблерных программ;
- навыками протоколирования и отладки низкоуровневых программ, локализацией и поиском ошибок.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 5 з.е. (180 академических часа(ов)).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов	
	Всего	Сем. №5
Контактная работа при проведении учебных занятий (всего):	96	96
В том числе:		
Занятия лекционного типа	48	48
Занятия семинарского типа	48	48

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 84 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме

контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

4. Содержание дисциплины (модуля).

4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	<p>1. Введение в курс Рассматриваемые вопросы: - понятие исходной и исполняемой программы, объектного и исполняемого кода в системах программирования; - сравнение ассемблеров и языков высокого уровня, применение языков низкого уровня.</p> <p>2. Кодирование целых чисел и арифметика в hex---кодах Рассматриваемые вопросы: - представление целых чисел знаковых и беззнаковых - представление двоично-кодированных чисел в hex-коде, переводы - арифметика в hex---кодах.</p> <p>3. Архитектура процессоров семейства x86 Рассматриваемые вопросы: - общая структура вычислительной системы, взаимодействие процессора с памятью, - командный цикл процессора, регистровые структуры процессоров x386.</p> <p>4. Архитектура процессоров семейства x86 Рассматриваемые вопросы: - сегментная адресация памяти процессором, внутрисегментные адреса; - сегментация программы, сегментные структуры программ, «разрядность» программы.</p> <p>5. Характеристика системы команд процессора Рассматриваемые вопросы: - структура многобайтной команды; - характеристика основных группы команд; - операнды в командах процессора, способы задания внутрисегментных адресов операндов.</p> <p>6. Основы символического языка ассемблера Рассматриваемые вопросы: - семантика ассемблерной программы, - директивы транслятора для описания программных сегментов.</p> <p>7. Основы символического языка ассемблера Рассматриваемые вопросы: - структура исходного текста односегментной и многосегментной программ; - директивы транслятора для размещения данных в памяти, резервирование памяти.</p> <p>8. Основы символического языка ассемблера Рассматриваемые вопросы:</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<p>- использование символических адресов данных и команд; - синтаксис записи числовых и символьных данных; - атрибуты длины операндов в командах процессора.</p> <p>9. Технология разработки и отладки ассемблерной программы Рассматриваемые вопросы: - этапы подготовки ассемблерной программы: трансляция, компоновка и отладка; - инструментальные средства Borland TurboAssembler.</p> <p>10. Технология разработки и отладки ассемблерной программы Рассматриваемые вопросы: - содержательный процесс разработки; - документирование разработки: содержательный и детальный алгоритмы, исходный текст и протокол трансляции, данные для отладки и протоколирование отладки.</p> <p>11. Пересылка данных и арифметические команды Рассматриваемые вопросы: - команды пересылки данных; - команды преобразования форматов данных; - команды целочисленной арифметики и примеры их использования.</p> <p>12. Логические команды Рассматриваемые вопросы: - команды логических операций и их выполнение процессором; - команды сдвигов, типы сдвигов; - команды битовых операций и их применение.</p> <p>13. Команды передачи управления Рассматриваемые вопросы: - безусловная передача управления: прямая и косвенная, внутрисегментные и межсегментные переходы; - условные внутрисегментные переходы по состоянию флагов и по соотношению величин - применение команд передачи управления для реализации разветвленных алгоритмов.</p> <p>14. Циклы. Работа с массивом данных. Рассматриваемые вопросы: - способы организации цикла командами процессора: вычитающий счетчик, суммирующий счетчик, команда цикла - особенности «массива», как способа размещения данных в памяти.</p> <p>15. Циклы. Работа с массивом данных. Рассматриваемые вопросы: - способы реализации внутрисегментной адресации элементов массива - косвенный и прямой с косвенным смещением; - строковые команды для работы с массивом, сравнительная оценка их эффективности.</p> <p>16. Стековые команды и процедурный вызов Рассматриваемые вопросы: - понятие стекового доступа, механизм выполнения процессором стековых команд; - процедурная передача управления, механизм выполнения процессором команд вызова процедуры и возврата из нее.</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<p>17. Стековые команды и процедурный вызов Рассматриваемые вопросы: - описание процедуры в кодовом сегменте, способы передачи входных и выходных параметров процедуры; - разработка программы с использованием процедуры с параметрами.</p> <p>18. Команды обращения к портам контроллеров Рассматриваемые вопросы: - понятие о контроллерах внешних устройств и их «портах»; - команды обращения к портам; - строковые команды пересылки между портами и памятью; - примеры программного обращения к портам.</p> <p>19. Использование системного сервиса в ассемблерных программах Рассматриваемые вопросы: - понятие и типы прерываний, механизм выполнения процессором прерывания; - вызов системного сервиса через программные прерывания; - характеристика системного сервиса операционной системы и BIOS, функции системного сервиса.</p> <p>20. Экранный системный сервис. Рассматриваемые вопросы: - алгоритмы преобразования числовых кодов в коды символов - 2-х, 10-х и 16-х кодов для последующего отображения; - системный сервис операционной системы для вывода символьной информации на экран, примеры использования.</p> <p>21. Системный сервис для ввода с клавиатуры. Рассматриваемые вопросы: - форматы кода нажатия клавиши контроллера клавиатуры; - алгоритмы преобразования символьной информации в числовые коды; - системный сервис для ввода с клавиатуры, примеры использования.</p> <p>22. Системный файловый сервис. Рассматриваемые вопросы: - характеристика системного файлового сервиса; - файловые функции системного сервиса для создание файла, открытие файла, чтение и/или изменение атрибутов файла, чтение байтов из файла/устройства, запись в файл/устройство, позиционирование смещения в файле, закрытие файла; - примеры использования файлового сервиса.</p> <p>23. Форматы команд процессора с операндами Рассматриваемые вопросы: - обозначения полей в описании форматов команд процессора x86 и их назначение; - структура байта кода операции, постбайт режима адресации; - механизм преобразования символической команды в машинный код (трансляция); - примеры формирования машинного кода по символическому виду команды.</p> <p>24. Форматы команд прямых переходов и вызовов, специфические форматы Рассматриваемые вопросы: - относительность прямых внутрисегментных переходов, короткие переходы; - формат прямых межсегментных переходов и вызовов; - специфические форматы команд.</p>

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	<p>1. Виртуализация операционных систем Студенты знакомятся со средствами программной виртуализации для самостоятельной домашней работы с 16-разрядными ассемблерными программами</p> <p>2. Инструментальные средства создания ассемблерной программы. Студент знакомится и приобретает навыки использования среды файлового менеджера для работы с консольными утилитами,</p> <p>3. Инструментальные средства создания ассемблерной программы. Студент знакомится и осваивает инструментальные средства TASM для трансляции и компоновки ассемблерной программы, знакомство с протоколом трансляции</p> <p>4. Отладка ассемблерной программы Студент осваивает технологию отладки низкоуровневой исполняемой программы в отладчике Td: работа в окне кода и окне регистров</p> <p>5. Отладка ассемблерной программы Студент осваивает возможности работы в окне данных отладчика Td</p> <p>6. Демонстрация навыков использования инструментальных средств. Отчет на компьютере</p> <p>7. Пересылка и преобразование формата данных Студент знакомится с командами процессора группы пересылки данных. Разрабатывает алгоритм для процессора по своему индивидуальному заданию.</p> <p>8. Пересылка и преобразование формата данных Студент создает исходный текст линейной программы по разработанному алгоритму, транслирует ее, компонует и получает исполняемую программу.</p> <p>9. Пересылка и преобразование формата данных Студент отлаживает свою первую законченную ассемблерную программу, готовит отчет и защищает выполненную работу</p> <p>10. Арифметические операции Студент знакомится с основными арифметическими командами процессора. Разрабатывает алгоритм решения второго индивидуального задания</p> <p>11. Реализация арифметических операций Студент создает исходную программу с использованием арифметических команд процессора и доводит ее до уровня исполняемого кода</p> <p>12. Отладка программы Подготовка данных для отладки, отладка исполняемой программы с использованием протокола трансляции, подготовка отчета. Защита выполненной работы</p> <p>13. Логические операции и сдвиги</p>

№ п/п	Наименование лабораторных работ / краткое содержание
	<p>Студент знакомятся с командами логических операций и командами сдвигов из системы команд процессора. Выполняются упражнения на практику их использования</p> <p>14. Разветвленные алгоритмы Студент знакомится с командами внутрисегментной передачи управления. Разрабатывает алгоритм задания с использованием логических разветвлений</p> <p>15. Реализация разветвлений. Студент приобретает навыки реализации разветвлений в исходном тексте программы, отладки программы на множественных вариантах исполнения</p> <p>16. Работа с массивами данных Студент знакомится с особенностями организации программной работы с массивами данных на уровне команд процессора. Осваивает способы внутрисегментной адресации данных в памяти</p> <p>17. Работа с массивами данных Выполнение индивидуального задания, предусматривающего разработку алгоритма и программы работы с массивами данных разных форматов</p> <p>18. Работа с массивами данных Подготовка данных для отладки, отладка исполняемого кода, протоколирование и подготовка отчета о выполненном задании</p> <p>19. Процедуры в ассемблерных программах Студент знакомится с механизмами выполнения процессором стековых команд, команд вызова процедур и возврата из процедуры, директивами транслятора для описания процедур</p> <p>20. Реализация процедур В ходе выполнения индивидуального задания студент приобретает навыки использования процедур в разрабатываемой программе, определять параметры процедур, навыки структурирования программной логики</p> <p>21. Реализация процедур Отладка разработанной программы с использованием процедур, ее протоколирование, подготовка отчета и защита работы</p> <p>22. Форматы команд с операндами В результате выполнения индивидуального задания студент закрепляет понимание принципов построения машинного кода процессора и процесса преобразования символической команды в машинный код, выполняемого транслятором</p> <p>23. Специфические форматы команд В результате выполнения задания ручной трансляции студент получает представление о специфике машинных кодах команд прямых переходов</p> <p>24. Дизассемблирование машинного кода Студенты получают навык получения символической команды на основе машинного кода команды</p>

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Анализ и проработка лекционного материала.
2	Изучение рекомендуемой учебной литературы
3	Освоение инструментария трансляции, компоновки и отладки ассемблерной программы
4	Выполнение индивидуальных заданий, предусмотренных лабораторными работами
5	Подготовка отчетов о выполнении индивидуальных заданий
6	Выполнение курсовой работы.
7	Подготовка к промежуточной аттестации.
8	Подготовка к текущему контролю.

4.4. Примерный перечень тем курсовых работ

Индивидуальное задание на курсовую работу предполагает программную разработку на тему «Использование системного сервиса в ассемблерных программах».

Примерный перечень тем курсовых работ

-Реализация ввода с клавиатуры символьных данных в двоичном виде и преобразование их в числовые коды.

-Реализация ввода символьных данных с клавиатуры в шестнадцатиричном виде и получение их числовых эквивалентов.

-Реализация ввода с клавиатуры символьных данных в десятичном знаковом виде и преобразование их в числовые коды.

-Преобразование числовых кодов в символьные и реализация вывода на экран в двоичном виде.

-Преобразование числовых кодов в символьные и реализация вывода на экран в шестнадцатиричном виде.

-Преобразование числовых кодов в символьные и реализация вывода на экран в десятичном виде.

-Чтение потока байтов из файла с размещением в памяти и их последующая обработка.

-Запись потока байтов в файл с заданной позиции, установка атрибутов файла.

-Обработка строк в текстовых файлах

5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
1	Ларина Т.Б. Технология подготовки и отладки ассемблерных программ. Методические указания. М: МИИТ, 2014. - 38 с.	каф.ВССиИБ, ауд.1332. - 30 экз.
2	Ларина Т.Б. Низкоуровневые языки. Учебное пособие. М.:РУТ (МИИТ), 2018. - 147 с.	http://library.miit.ru/bookscatalog/metod/DC-899.pdf (доступа: 01.03.2024). - Текст : непосредственный
3	Ларина Т.Б. Использование системного сервиса в ассемблерных программах. Учебное пособие. М: МИИТ, 2009. -132 с.	каф.ВССиИБ, ауд.1332. - 30 экз.
4	Ларина Т.Б. Виртуализация операционных систем. Учебное пособие. - М.: РУТ (МИИТ), 2020. - 65 с.	http://library.miit.ru/bookscatalog/metod/DC-1368.pdf (доступа: 01.03.2024). - Текст : непосредственный каф.ВССиИБ, ауд.1332. - 30 эк

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

Официальный сайт РУТ (МИИТ) <http://miit.ru>

Научно-техническая библиотека РУТ (МИИТ): <http://library.miit.ru>

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

Операционная система Microsoft Windows, Microsoft Office.

Программные средства виртуализации операционных систем (при использовании хостовых 64-разрядных систем).

Интегрированные программные средства Borland разработки и отладки ассемблерных программ для реального режима процессоров.

При проведении занятий с применением дистанционных образовательных технологий могут применяться средства коммуникаций: ЭИОС РУТ(МИИТ), Microsoft Teams, электронная почта, Zoom, WhatsApp.

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).

Лекционная аудитория, оснащенная компьютером и проектором.

Персональные компьютеры в учебной лаборатории с необходимым программным обеспечением.

В случае проведения дистанционных занятий необходимо наличие средств для организации удаленных коммуникаций.

9. Форма промежуточной аттестации:

Курсовая работа в 5 семестре.

Экзамен в 5 семестре.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

доцент, доцент кафедры
«Вычислительные системы, сети и
информационная безопасность»

Т.Б. Ларина

Согласовано:

Заведующий кафедрой ВССиИБ

Б.В. Желенков

Председатель учебно-методической
комиссии

Н.А. Андриянова