

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



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
базового высшего образования
по направлению подготовки
02.03.02 Фундаментальная информатика и
информационные технологии,
утвержденной первым проректором ФГАОУ ВО
РУТ (МИИТ) Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Программирование для процессоров архитектуры x86

Направление подготовки: 02.03.02 Фундаментальная информатика и
информационные технологии

Направленность (профиль): Квантовые 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 ФГАОУ ВО РУТ
(МИИТ)

ID подписи: 4196

Подписал: заведующий кафедрой Желенков Борис
Владимирович

Дата: 14.09.2026

1. Общие сведения о дисциплине (модуле).

Целями освоения дисциплины являются:

- освоение дополнительных возможностей символического языка транслятора (ассемблера)

- приобретение опыта использования системы команд процессора.

Задачами дисциплины (модуля) являются:

- формирование дополнительных навыков разработки низкоуровневых программ на символическом языке ассемблера

- получение навыков анализа и ручного дизассемблирования машинного кода программ

2. Планируемые результаты обучения по дисциплине (модулю).

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

ОПК-4 - Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям;

ОПК-7 - Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности.

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

Знать:

- расширенные конструкции символического языка ассемблера;
- дополнительные структуры ассемблерных программ, принципы дизассемблирования.

Уметь:

- проектировать содержательные и детальные алгоритмы низкоуровневых программ;
- реализовывать содержательные и детальные алгоритмы низкоуровневых программ.

Владеть:

- расширенными навыками разработки ассемблерных программ;

- навыками анализа и дизассемблирования машинного кода

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 2 з.е. (72 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	Семестр №5
Контактная работа при проведении учебных занятий (всего):	32	32
В том числе:		
Занятия лекционного типа	16	16
Занятия семинарского типа	16	16

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 40 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	Нестандартные многосегментные структуры ассемблерных программ Рассматриваемые вопросы:

№ п/п	Тематика лекционных занятий / краткое содержание
	- структура программы с несколькими кодовыми сегментами - структура программы с использованием более шести сегментов данных
2	Макрокоманды в ассемблере Рассматриваемые вопросы: - псевдокоманды транслятора – макрокоманды, их назначение - описание макрокоманды и ее параметры - использование макрокоманды
3	Символические средства в языке ассемблера Рассматриваемые вопросы: - сопоставление символического имени непосредственному операнду - использование регистра IP в командах переходов - арифметические конструкции ассемблера для вычислений на этапе трансляции
4	BSCD-коды и операции с ними Рассматриваемые вопросы: - двоично-десятичное кодирование данных - команды процессора для выполнения арифметики с bcd-кодами данных
5	Использование системного сервиса BIOS Рассматриваемые вопросы: - структура полей в машинных кодах команд процессора и их назначение; - поля кода операции и постбайт режима адресации - поля непосредственного операнда и числовых адресных смещений
6	Ассемблирование символических команд в машинные коды Рассматриваемые вопросы: - механизм преобразования символической команды в машинный код (трансляция); - примеры формирования машинного кода по символическому виду команды
7	Дизассемблирование машинных кодов Рассматриваемые вопросы: - принцип и логика дизассемблирования; - примеры дизассемблирования машинного кода

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Многосегментные структуры программ На занятии разбираются варианты реализации ассемблерной программы с несколькими кодовыми и несколькими сегментами данных
2	Макрокоманды ассемблера На занятии решается задача реализации некоторого функционала в виде процедуры и в виде макрокоманды. Обсуждаются преимущества и недостатки каждого решения

№ п/п	Тематика практических занятий/краткое содержание
3	Символические средства в языке ассемблера На занятии решается задача, в которой предлагается использовать вычислительные операции непосредственно в символической записи команды, символические обозначения констант и реализовать переходы относительно текущего значения внутрисегментного адреса
4	BCD-коды и операции с ними На занятии выполняется задача на арифметические действия над данными, представленными в двоично-десятичном коде.
5	Использование системного сервиса BIOS На занятии разрабатываются фрагменты программы с использованием функций консольного (экран, клавиатура) сервиса BIOS. Приводятся примеры использования информационных сервисов BIOS
6	Форматы машинных команд процессора с операндами На занятии на примерах разбирается структура машинного кода команды без операндов, с одним операндом и с двумя операндами
7	Ассемблирование символических команд в машинные коды Выполняется практическая работа по ручному ассемблированию предложенных команд процессора в машинный код
8	Дизассемблирование машинных кодов Дизассемблирование машинных кодов. Выполняется практическая работа по ручному дизассемблированию предложенных в машинном виде команд в символический вид

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Анализ и проработка лекционного материала.
2	Изучение рекомендуемой учебной литературы
3	Анализ и проработка результатов выполненных на практических занятиях работ
4	Подготовка к промежуточной аттестации.

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

№ п/п	Библиографическое описание	Место доступа
1	Ларина Т.Б. Низкоуровневые языки. Учебное пособие. М.: РУТ (МИИТ), 2018. -147 с.	http://library.mii.ru/bookscatalog/metod/DC-899.pdf (дата доступа: 20.02.2026). - Текст : непосредственный.

2	Ларина Т.Б. Программирование на ассемблере в информационных системах железнодорожного транспорта: Учеб.пособие для вузов жел.-дор.транспорта. –М.:МИИТ,2005.- 175 с.	http://library.miit.ru/miitpublishing/04-35159.pdf (дата обращения: 20.02.2026). - Текст : непосредственный.
3	Ларина Т.Б. Виртуализация операционных систем. Учебное пособие. - М.: РУТ (МИИТ), 2020.	http://library.miit.ru/bookscatalog/upos/DC-1368.pdf (дата обращения: 20.02.2026). - Текст : непосредственный.
4	Зубков С.В. Assembler. Для DOS, Windows и Unix: практическое руководство. Москва: ДМК-Пресс, 2017, - 639 с.	https://znanium.ru/catalog/product/2046016 (дата обращения: 19.03.2026). – Режим доступа: по подписке.

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).

необходимого для освоения дисциплины (модуля).

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.

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

Учебная аудитория для проведения учебных занятий (занятий лекционного типа, лабораторных работ, курсового проектирования (выполнения курсовых работ):

- компьютер преподавателя, рабочие станции студентов, мультимедийное оборудование, доска.

Аудитория подключена к сети «Интернет».

9. Форма промежуточной аттестации:

Зачет 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 и
квантовые коммуникации»

Т.Б. Ларина

Согласовано:

Заведующий кафедрой ВСиКК

Б.В. Желенков

Председатель учебно-методической
комиссии

Н.А. Андриянова