

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«САХАЛИНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

**Аннотация рабочей программы дисциплины
Б1.В.ДВ.5.2 Программирование на низкоуровневом языке Assembler**

Направление подготовки
01.03.02 Прикладная математика и информатика

Профиль подготовки
Системное программирование и компьютерные технологии

1. Цели освоения дисциплины

Целью дисциплины Программирование на низкоуровневом языке Assembler, является формирование у студентов теоретических знаний о принципах программирования микропроцессорных систем, способности самостоятельно выполнять программы на языке Assembler.

2. Место дисциплины в структуре образовательной программы

Дисциплина Программирование на низкоуровневом языке Assembler относится к разделу дисциплин по выбору (Б1.В.ДВ.5.2.). Для изучения дисциплины необходимы знания, умения и навыки, сформированные в результате изучения дисциплин: Языки и методы программирования; Операционные системы; Объектно-ориентированное программирование. Данная дисциплина необходима для успешного прохождения учебной и преддипломной практики.

3. Требования к результатам освоения содержания дисциплины

Дисциплина нацелена на формирование общепрофессиональных компетенций ОПК-2, ОПК-4 и профессиональных компетенций ПК-3, ПК-4 выпускника.

общепрофессиональные компетенции (ОПК):

(ОПК-2)	– способностью приобретать новые научные и профессиональные знания, используя современные образовательные и информационные технологии;
(ОПК-4)	– способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.

профессиональные компетенции (ПК):

(ПК-3)	– способностью критически переосмысливать накопленный опыт, изменять при необходимости вид и характер своей профессиональной деятельности
(ПК-4)	– способностью работать в составе научно-исследовательского и производственного коллектива и решать задачи профессиональной деятельности;

В результате освоения дисциплины студент должен:

Знать:

- систему команд центрального процессора;
- иерархию памяти;

- принципы взаимодействия с внешними устройствами.

Уметь:

- программировать на языке Assembler;
- осуществлять вставки на языке Assembler в программы, написанные на языке С.

Владеть:

- навыками разработки программ на языке Assembler;
- навыками использования отладчиков.

4. Структура и содержание дисциплины *Программирование на низкоуровневом языке Assembler*

Для очной формы обучения общая трудоемкость дисциплины составляет 4 зачетные единицы, 144 часа, в том числе лекции – 16 часов, лабораторные занятия – 30 часов, самостоятельная работа студента – 62 часа. Вид промежуточной аттестации – экзамен (.

№ п/п	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)						Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
		всего	лк	лб	срс	контроль	зет	
1	7	144	16	30	62	36	4	экзамен
итого		144	16	30	62	36	4	

№ п/п	Раздел Дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)	
			всего	лк	лб	срс	экз	по неделям семестра	по семестрам
1.	Основы архитектуры семейства 80x86	7	14	2	2	10	36	Лабораторная работа	Экзамен
2.	Команды ЦП: пересылки, арифметические, перехода. Кодирование команд		16	2	4	10		Лабораторная работа	
3.	Стек, подпрограммы, программные прерывания.		16	2	4	10		Лабораторная работа	
4.	Средства языков Ассемблера		18	2	6	10		Лабораторная работа	
5.	Команды ЦП: битовые, строковые		20	4	6	10		Лабораторная работа	
6.	Модульное программирование		24	4	8	12		Лабораторная работа	
Итого			144	16	30	62	36		

Содержание разделов дисциплины

Раздел 1. Основы архитектуры семейства 80x86

Общие сведения о семействе процессоров 80x86. Общая шина. Память: ОЗУ и ПЗУ. Кэш-память. Адресное пространство. Сегмент и смещение, параграф. Нормализованный адрес. Распределение памяти персональной ЭВМ: обычная память, видеобуфер, ROM BIOS, дополнительная (expanded) и расширенная (extended) память, область верхней памяти (UMA – upper memory area). Команды отладчика debug для просмотра ячеек памяти. Сегментные

регистры. Структура центрального процессора: исполнительный блок и шинный интерфейс. Конвейеризация. Очередь команд. Регистры центрального процессора: HL-регистры, PC-регистры, счетчик команд, регистр флагов. Назначение битов регистра флагов. Работа с отладчиком debug: запуск отладчика, пример сеанса работы с отладчиком, команды отладчика, сгруппированные по функциональному признаку.

Раздел 2. Команды ЦП: пересылки, арифметические, перехода. Кодирование команд

Методы адресации: регистровый и непосредственный. Структура описания команд. Команды пересылки. Арифметические команды: сложение, вычитание. Длинные целые. Операнды различной длины. Арифметические команды: умножение, деление. Десятичная арифметика. Условные переходы. Команды безусловных переходов. Индексная адресация. Команды организации циклов. Косвенная адресация. Команды косвенного перехода. Кодирование команд: однооперандные команды, непосредственный операнд, двухоперандные команды. Префикс замены сегмента. Команды загрузки исполнительного адреса.

Раздел 3. Стек, подпрограммы, программные прерывания.

Стек. Команды работы со стеком. Использование стека для организации вложенных циклов. Стековый кадр. Подпрограммы. Передача параметров: через стек и через регистры. Возвращение результата через флаг CF. Команды изменения флага CF. Вложенные подпрограммы. Использование Turbo Debugger для отладки программ. Прерывания: их классификация. Команды запрета и разрешения внешних прерываний. Программные прерывания. Векторы прерываний. Команды прерываний. Пример программы обработки прерывания: определение размера доступной памяти. Однобайтная команда INT и ее применение при трассировке программ. Структура программного обеспечения IBM PC. Таблица векторов прерываний. Прерывания BIOS (примеры). Прерывания DOS (примеры). Трассировка подпрограмм и прерываний в отладчиках. Прерывание INTO. Операции с сегментными регистрами. Организация прерывания посредством дальнего косвенного вызова.

Раздел 4. Средства языков Ассемблера

Язык Ассемблера. Определение данных. Пример программы на языке Ассемблера. Этапы создания программы: ассемблирование, компоновка, выполнение. Листинг. Сообщения TASM. Различия в диагностике ошибок TASM и MASM. Карта памяти. Подпрограммы. Макросы. Вложенные макросы. Условное ассемблирование. Локальные метки. Блоки повторений. Включаемые файлы. Использование кода завершения в командных файлах. Директивы определения идентификаторов. Командный файл для ассемблирования, компоновки и отладки программ, содержащихся в одном файле.

Раздел 5. Команды ЦП: битовые, строковые

Битовые команды: булевские, линейные и циклические сдвиги. Битовые операции на этапе ассемблирования. Локальный счетчик адреса. Строковые команды: пересылка строк. Флаг направления. Префикс повторения. Команды сравнения. Команды загрузки адресов. Пример применения битовых и строковых команд: заполнение экрана в текстовом режиме. Команда табличного преобразования.

Раздел 6. Модульное программирование

Директивы связи. Сегментные директивы. Директива ASSUME. Параметры директивы SEGMENT. Сегментные группы. Взаимоотношение упрощенных и стандартных сегментных директив. Модели памяти. Исполняемые файлы в COM-формате. Отладка COM-программ. Встроенный ассемблерный код (inline). Стыковка модулей, написанных на языках Си и Ассемблера. Передача параметров в Си. Примеры программ для разных моделей памяти. Директивы, упрощающие стыковку модулей.