

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

**Аннотация рабочей программы дисциплины
Б1.В.ДВ.5.1 Программирование на языке ассемблера NASM**

Направление подготовки
01.03.02 Прикладная математика и информатика

Профиль подготовки
Системное программирование и компьютерные технологии

1. Цели освоения дисциплины

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

2. Место дисциплины в структуре образовательной программы

Дисциплина Программирование на языке ассемблера NASM относится к разделу дисциплин по выбору (Б1.В.ДВ.5.1.). 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-4, ПК-5 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-4)	способностью работать в составе научно-исследовательского и производственного коллектива и решать задачи профессиональной деятельности;
(ПК-5)	способностью осуществлять целенаправленный поиск информации о новейших научных и технологических достижениях в информационно-телекоммуникационной сети "Интернет" (далее – сеть "Интернет") и в других источниках.

В результате освоения дисциплины студент должен:

Знать:

- систему команд центрального процессора;
- иерархию памяти;
- принципы взаимодействия с внешними устройствами.

Уметь:

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

Владеть:

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

4. Структура и содержание дисциплины *Программирование на языке ассемблера NASM*

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 системы команд i386	7	14	2	2	10	36	Лабораторная работа	Экзамен
2.	Стек, подпрограммы, рекурсия		16	2	4	10		Лабораторная работа	
3.	Основные особенности ассемблера NASM		16	2	4	10		Лабораторная работа	
4.	Макросредства и макропроцессор		18	2	6	10		Лабораторная работа	
5.	Взаимодействие с операционной системой		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 системы команд i386***

Система регистров. Память пользовательской задачи. Директивы для отведения памяти. Команда mov. Косвенная адресация, исполнительный адрес. Размеры операндов и их допустимые комбинации. Целочисленное сложение и вычитание. Целочисленное умножение и деление. Условные и безусловные переходы. Побитовые операции, строковые операции.

Раздел 2. Стек, подпрограммы, рекурсия

Понятие стека и его предназначение. Организация стека в процессоре. Дополнительные команды работы со стеком. Подпрограммы: общие принципы. Вызов подпрограмм и возврат из

них. Организация стековых фреймов. Основные конвенции вызовов подпрограмм. Локальные метки.

Раздел 3. Основные особенности ассемблера NASM

Ключи и опции командной строки. Основы синтаксиса. Псевдокоманды. Константы. Вычисление выражений во время ассемблирования. Критические выражения. Выражения в составе исполнительного адреса

Раздел 4. Макросредства и макропроцессор

Простейшие примеры макросов. Однострочные макросы; макропеременные. Условная компиляция. Макроповторения. Многострочные макросы и локальные метки. Макросы с переменным числом параметров. Макродирективы для работы со строками.

Раздел 5. Взаимодействие с операционной системой

Мультизадачность и её основные виды. Аппаратная поддержка мультизадачности. Прерывания и исключения. Системные вызовы и «программные прерывания». Конвенция системных вызовов ОС Linux. Конвенция системных вызовов ОС FreeBSD. Примеры системных вызовов. Доступ к параметрам командной строки.

Раздел 6. Модульное программирование

Директивы связи. Сегментные директивы. Директива ASSUME. Параметры директивы SEGMENT. Сегментные группы. Взаимоотношение упрощенных и стандартных сегментных директив. Модели памяти.