

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

«Приднестровский государственный университет
имени Т.Г. Шевченко»

Рыбницкий филиал

Кафедра информатики и программной инженерии



Директор Рыбницкого филиала ПГУ им. Т.Г. Шевченко,
профессор И.А. Павлинов
« 15 » 10 2020 г.

РАБОЧАЯ ПРОГРАММА

на 2020/2021 учебный год

УЧЕБНОЙ ДИСЦИПЛИНЫ

«МАШИННО-ЗАВИСИМЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ»

2.09.03.04 «Программная инженерия»

Профиль подготовки:

«Разработка программно-информационных систем»

квалификация (степень) выпускника:

Бакалавр

Форма обучения:

очная

Год набора 2018

Рыбница 2020г.

Рабочая программа дисциплины «*Машино-зависимые языки программирования*»/составитель
А.Б. Глазов. – Рыбница: Рыбницкий филиал ПГУ им. Т.Г. Шевченко, 2020. – 13 с.

**РАБОЧАЯ ПРОГРАММА ПРЕДНАЗНАЧЕНА ДЛЯ ПРЕПОДАВАНИЯ ДИСЦИПЛИНЫ
ВАРИАТИВНОЙ ЧАСТИ ДИСЦИПЛИН (МОДУЛЕЙ) СТУДЕНТАМ ОЧНОЙ ФОРМЫ
ОБУЧЕНИЯ ПО НАПРАВЛЕНИЮ ПОДГОТОВКИ 2.09.03.04 – «ПРОГРАММНАЯ
ИНЖЕНЕРИЯ», ПРОФИЛЯ «РАЗРАБОТКА ПРОГРАММНО-ИНФОРМАЦИОННЫХ
СИСТЕМ».**

Рабочая программа составлена с учетом Федерального Государственного образовательного стандарта высшего образования по направлению подготовки 2.09.03.04 – «Программная инженерия», утвержденного приказом Министерства образования и науки Российской Федерации от 12 марта 2015 года №229.

Составитель _____ А.Б. Глазов, ст. преподаватель

Сай

1. Цели и задачи освоения дисциплины

Целью освоения дисциплины является получение знаний студентами по постановке, подготовке задач к решению и решению задач на машинно-ориентированном языке ассемблер.

В результате изучения дисциплины студенты должны:

- знать возможности машинно-ориентированного языка ассемблер, форматы представления данных, команд и директив, их использование, логику работы и технологию отладки и решения задач;
- уметь решать задачи с использованием языка ассемблера, создавать и использовать программные модули, организовывать ввод и вывод данных, использовать макросредства и подпрограммы ассемблера.

2. Место дисциплины в структуре ООП ВО

Дисциплина «Машинно-зависимые языки программирования» относится к вариативной части блока дисциплин (модулей) (Б1.В.09), читается студентам очной формы обучения по направлению 2.09.03.04 – «Программная инженерия» профиля подготовки «Разработка программно-информационных систем».

Для изучения данной дисциплины необходимы знания школьного курса математики, а также знания по дисциплинам ООП бакалавриата: «Типы и структуры данных», «Программирование», а также навыки, полученные в период прохождения учебной (компьютерной) практики.

Дисциплина является предшествующей для изучения дисциплин «Защита информации», «Производственная практика», «Преддипломная практика».

3. Требования к результатам освоения дисциплины

Изучение дисциплины направлено на формирование следующих компетенций:

Код компетенции	Формулировка компетенции
ПК-1	готовностью применять основные методы и инструменты разработки программного обеспечения
ПК-21	владением навыками чтения, понимания и выделения главной идеи прочитанного исходного кода, документации

В результате изучения данной дисциплины студент должен:

3.1. Знать:

- архитектуру микропроцессоров семейства x86 в реальном и защищенном режимах;
- организацию оперативной памяти и структуру адресации;
- элементы, операнды, директивы и типы данных языка ассемблера;
- системы исчисления, представления целочисленных чисел и чисел с плавающей точкой;
- архитектуру, типы данных и команды расширений процессора и сопроцессора;
- структуру и организацию программы на ассемблере;
- программные средства и библиотеки языка ассемблера для управления оборудованием и внешними устройствами.

3.2. Уметь:

- составить программу на языке ассемблера по заданному алгоритму решения задачи;
- использовать ассемблерные вставки в программах на языках высокого уровня;
- оптимизировать программу с использованием языка ассемблера для увеличения быстродействия и уменьшения размера машинного кода;
- использовать макросредства языка ассемблера;

3.3. Владеть:

- навыками составления программы на языке ассемблера, трансляции программы в машинный код, компоновки программы;
- отладки программы в машинной коде с помощью инструментальных средств;
- анализа и дизассемблирования машинного кода;
- создания приложения на языке ассемблера для ОС Windows, подключаемых библиотек и драйверов устройств.

4. Структура и содержание дисциплины

4.1. Распределение трудоемкости в з.е./часах по видам аудиторной и самостоятельной работы студентов по семестрам

Студентов по семестрам							
Семестр	Количество часов						Форма итогового контроля
	Трудоемкость, з.е./часы	В том числе				Самост. работы	
		Аудиторных					
		Всего	Лекций	Лаб. раб.	Практич. зан.		
5	3/108	54	18	36	-	54	Зачет с оценкой
Итого:	3/108	54	18	36	-	54	Зачет с оценкой

4.2. Распределение видов учебной работы и их трудоемкости по разделам дисциплины

№ раздела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеауд. работа (СР)
			Л	ПЗ	ЛР	
1	Основы программирования на языке ассемблера	36	6	-	12	20
2	Команды и операции языка ассемблера	36	6	-	12	16
3	Создание приложений на языке ассемблера	36	6	-	12	18
Итого:		108	18	-	36	54

4.3. Тематический план по видам учебной деятельности

Лекции

№ п/п	Номер раздела дисциплины	Объем часов	Тема лекции	Учебно-наглядные пособия
1	1	2	Базовая архитектура компьютера с точки зрения программиста. Система с ОШ. Взаимодействие ЦП с контроллерами внешних устройств и памятью. Типовые временные промежутки по временам доступа Базовая архитектура процессоров Intel x86. История развития процессоров Intel, Кэш 1 и 2го уровня. Конвейер команд т данных. Регистры CPU и FPU. РОНЫ, сегментные регистры, Счетчик команд, регистр признаков. Работа со стеком	Мультимедийные презентации
2		2	Основы создания приложений на языке ассемблера. Ассемблирование исходного текста. Компоновка программ. Системы счисления, применяемые в Ассемблере. Особенности разработки программ в MASM. Синтаксис языка ассемблера. Представление данных в компьютере. Первичные элементы языка ассемблера. Операнды. Операнды-выражения. Директивы сегментации. Простые типы данных ассемблера. Способы адресации данных.	Мультимедийные презентации
3		2	Структура программы на языке ассемблера. Организация сегментов. Директивы управления сегментами и моделями памяти макроассемблера MASM. Структура программ на ассемблере MASM.	Мультимедийные презентации
6		2	Команды пересылки. Порядок расположения в памяти многобайтных переменных. Размещение переменных в памяти. Команды MOV, XCHG, LEA, PUSH, POP Организация табличного преобразования XLAT.	Мультимедийные презентации

			Преобразование типов CBW, CWD/ Двоичная арифметика ADD, ADC, INC, SUB, SBB, DEC, MUL, DIV, NEG. Команды, учитывающие знак операнда: IMUL, IDIV. Команды десятичной коррекции AAA, DAA, AAS, DAS. Вспомогательные команды BSWAP, CMP. Влияние на биты флагов.	
7	2	2	Команды логики и сдвига CPU. Команды AND, OR, XOR, NOT, TEST, их применение для проверки, установки и сброса отдельных битов. Команды сдвига: Линейный, арифметический и циклический сдвиги. Влияние на биты флагов. Организация вычислительных циклов. Безусловные переходы JMP, CALL, RET, INT. Метка \$. Прямой, короткий и косвенный переходы. Условные переходы: по результату сравнения, по состоянию флагов, по содержимому CX/ Команды управления циклами: Условные переходы и ветвления. Команды безусловного перехода. Организация циклов. LOOP, LOOPNZ	Мультимедийные презентации
8			Способы генерации исполняемых модулей в MASM32. Разделение процесса создания на компиляцию и редактирование связей модуле. Параметры компиляции. Параметры редактор связей. BAT файлы для автоматической установки параметров компиляции	Мультимедийные презентации
11		2	Создание консольных приложений в MASM32. Организация процесса исполнения программы. Разбиение программы на процедуры. Многомодульное программирование. Глобальные и локальные параметры процедуры Организация ввода-вывода информации в консольных приложениях в MASM32. Функции ввода-вывода для консольных приложений. Макросы в MASM32 и их применение для организации ввода-вывода в консольной программе Создание оконных приложений в MASM32. Основные составляющие оконного приложения: создание класса Окна, обработчик очереди сообщений, процедуры реакций на конкретные сообщения. CallBack вызовы оконных функций: механизм и назначение.	Мультимедийные презентации
12	3	2	Основные сведения по оконным сообщениям. Назначение параметров структуры сообщения. Типовые значения кода сообщения и способы их обработки. Организация ввода-вывода информации в оконных приложениях в MASM32. Функции ввода-вывода для оконных приложений. Макросы в MASM32 и их применение для организации ввода-вывода в оконной программе. Применение ресурсов для обмена информацией. Диалог – упрощенный способ создания оконного приложения. Отличие Диалогового приложения от оконного. Упрощенная структура диалогового приложения. Модальный и немодальный диалоги.	Мультимедийные презентации
13		2	Способ разделения исполняемого кода. Динамическая библиотека. Структура Динамической библиотеки. Процесс написания и компиляции библиотеки. Предоставления отдельных функций для внешнего использования. Применение Сторонних динамических библиотек в программе на Ассемблере. Два основных способа использования процедур внешней библиотеки: статическое и динамическое подключение. Работа с файловой системой. Создание пустого файла. Запись в файл текстовых данных. Чтение из файла текстовых данных.	Мультимедийные презентации
Итого:		18		

Практические занятия

Не предусмотрены учебным планом

Лабораторные работы

№ п/п	Номер раздела дисциплины	Объем часов	Тема лабораторного занятия	Наименование лаборатории	Учебно- наглядные пособия
1	1	4	Пакет MASM32. Знакомство, установка. Структура и назначение подкаталогов. Основные исполняемые файлы. Процесс создания исполняемого файла из исходного, Компилятор ML и редактор связей LINK	Компьютерный класс	Мультимедийные средства
2		4	Средства отладки OllyDBG. Общие сведения, установка пакета. Структура и назначение отдельных файлов отладчика. Настройка путей перед использованием. Назначение отдельных окон среды отладки. Создание исполняемых модулей с учетом возможности отладки. Процесс отладки простого модуля	Компьютерный класс	
3		4	Базовые средства построения и отладки консольных приложений. Стандартные функции ввода-вывода на консоль. Форматирование числовых данных перед выводом	Компьютерный класс	
4		2	Составление программ разветвляющихся и циклических вычислительных процессов	Компьютерный класс	
5	2	4	Программирование задач при работе с байтами и битами.	Компьютерный класс	
6		4	Программирование задач с использованием макрокоманд .	Компьютерный класс	
7		4	Основы построения оконных приложений. Основная функция, Цикл обработки сообщений, Оконная функция	Компьютерный класс	
8	3	4	Ресурсы оконных приложений: Главное меню, кнопки, Иконки. Подключение их в приложение с обработчиками	Компьютерный класс	
9		2	Диалоговое окно как ресурс и как основа приложения. Работа с файлами и памятью	Компьютерный класс	
10		4	Создание и применение библиотек динамической компоновки. Подключение их на этапе компиляции и в RunTime	Компьютерный класс	
Итого:		36			

Самостоятельная работа студента

Раздел дисциплины	№ п/п	Тема и вид СРС	Трудоемкость (в часах)
Раздел 1	1	Выполнение четырех лабораторных работ, оформление отчетов	8
	2	Изучение параметров ml, link	6
	3	OlyDebug – назначение основных окон и пунктов меню	8
Раздел 2	4	Выполнение четырех лабораторных работ, оформление отчетов	8
	5	Ресурсы типа курсора, рисунков текстовых полей	6
	6	Изучить графический редактор ресурсов	6
Раздел 3	7	Выполнение двух лабораторных работ, оформление отчетов	6
	8	Построить DLL для бинарной обработки данных	6
Итого:			54

5. Примерная тематика курсовых проектов (работ)

Не предусмотрено учебным планом

6. Образовательные технологии

Семестр	Вид занятия (Л, ПР, ЛР)	Используемые интерактивные образовательные технологии	Количество часов
1	Л	Разбор конкретных ситуаций, экспресс-опросы, интерактивные лекции с элементами мозгового штурма, дискуссии, проблемные лекции	7
	ЛР	Метод проектов, работа в малых группах	8
Итого:			15

7. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

I вариант

- Какая из следующих директив используется для выделения байта памяти?
a) DB b) DW c) DD d) DX
- К какому режиму адресации относится следующая запись:
MOV AX, [BX+2]
a) не явный b) непосредственный
c) косвенно-регистровый d) индексный
- Какой из следующих регистров является указателем стека?
a) BP b) SP c) CX d) DI
- Какое действие выполняет следующая последовательность команд:
MOV CX, 500
REP MOVS STR1, STR2
a) загрузку строки STR1 в строку STR2, повторяя цикл 500 раз
b) загрузку строки STR2 в строку STR1, повторяя цикл 500 раз
c) сравнение строк STR1 и STR2 500 раз
d) сохранение строк STR1, STR2 повторяя цикл 250 раз
- В какой регистр засылается число количества повторений при работе с командой LOOP?
a) SI b) DX c) BP d) CX

II вариант

- Какая из следующих команд выполняет арифметический сдвиг влево?
a) SHL b) SAL c) SHR d) SAR
- К какому режиму адресации относится следующая запись:
MOV AX, BX
a) прямая адресация b) регистровая адресация
c) косвенно-регистровая d) индексная адресация
- Какой парой регистров определяется команда, которая должна быть выполнена следующей после выполнения процедуры:
a) BX, DS b) CS, IP c) CS, BP d) DS, IP
- Для выделения 9-го бита в слове находящемся в регистре AX и замены его на противоположный необходимо записать последовательность команд :
a) MOV DX, AX
AND DX, 200H
NOT DX
AND DX, 200H
c) MOV DX, AX
TEST DX, 200H
NOT DX
OR AX, DX
b) MOV DX, AX
AND DX, 0FFFDH
NOT AX
d) AND AX, 38H
ROL AX, 1

5. Следующая команда ADD является...

- а) одноадресной в) безадресной
с) двухадресной d) трехадресной

III Вариант

1. Какой из следующих регистров не может непосредственно участвовать в операциях?
a) CX b) CS c) AX d) DI
2. Команды работы со строками обеспечивают обработку последовательности строк с длиной....
a) 64 kb b) 128 kb c) 32 kb d) 256 kb
3. Какая директива определяет макроопределение имеющее только один формальный аргумент и расширяющееся без дополнительного макровывода?
a) REPT b) IRPC c) IRP d) PURGE
4. Какое действие выполняет следующее посредственность команд :

```
C: MOV DX, N-1
```

MOV CX, DX

MOV BX, 0

B: MOV AX, MUS [BX]

CMP AX, MUS [BX+2]

JLA

XCHG AX, MUS[BX+2]

```
MOV     MUS[BX], AX
```

A: ADD BX,2

LOOP B

DEC DX

MOV CX, DX

LOOP C

- a) запись массива из одного места в другую
- b) сортировка массива
- c) нахождение $\max$ Эл-та в массиве
- d) нахождение $\min$ в массиве.

IV Вариант

1. Какая из этих следующих команд выполняет загрузку символа-байта строки в регистр AX?
a) LODSB b) STOSB c) SCASB d) LODSW
2. С каким регистром работает команда LOOP?
a) CX b) DX c) BP d) SI
3. Содержимое какого регистра можно использовать в качестве номера порта?
a) DI b) BX c) DX d) CX
4. Какое действие выполняет следующая последовательность команд:

```
MOV CX, 100
```

REPNE CMPS DEST, SOUR

- а) сравнивают строки до тех пор, пока не найдется элемент в строке DEST
совпадающих с соотношением элементом в SOUR или пока не будет просмотрено 100 пар
- б) сравнивает 100 пар или пока не найдется элемент в DEST не совпадающий с элементом
SOUR
- в) просмотрит все 100 пар безусловно
- г) повторит цикл сравнения 10 раз
5. Установка и сброс какого бита регистра флагов обеспечивает пошаговое
выполнения программы?
- а) IF б) TF в) ZF г) CF

V Вариант

1. Какой парой регистров характеризуется строка-источник?
a) DS, SI b) ES, DI c) DS, BX d) ES, CX
2. Какая из следующих команд выполняет арифметический сдвиг вправо?
a) SHL b) SAL c) SHR d) SAR
3. К какому режиму адресации относится следующая запись: MOV AX, 2[SI]
a) прямая b) регистровая
c) по базису d) индексная
4. Какое действие выполняет следующая запись
MOV DX, ((C GE 10) AND 3) OR ((C LT 10) AND 5)
загружает значение C в регистр DX если оно > или равно 10
a) загружает значение 3 в DX, если $C \geq 10$ и 5 в противном случае
b) загружает значение 3 в регистр DX если $C \leq 10$ и 5 в противном случае
c) загружает значение регистра DX в C если $DX \geq 10$
5. Установка и сброс какого бита регистра флагов обеспечивает направление обработки строки?
a) DF b) CF c) ZF d) IF

VI Вариант

1. В каком регистре сохраняется остаток от деления, если операндами являются байты:
a) AX b) BL c) BH d) AH
2. Какая команда помещает содержимое регистра-аккумулятора в строку?
a) LODS b) STOS c) SCAS d) CMPS
3. При выполнении команды вызова процедуры CALL(near) содержимое какого регистра помещается в стек?
a) IP b) BP c) BH d) DS
4. Какое действие выполняет следующая последовательность команд:
mov si, 0
mov ax, dx
A1: div 10
Mov buf[si], ah
inc si
cmp al, 0
JNE A1
a) выводит на экран монитора последовательность символов.
b) формирует массив чисел.
c) выполняет деление чисел в массиве.
d) формирует многозначное число в виде остатков для вывода его на экран.
5. Какая из следующих команд равносильна команде деления на два?
a) SAL b) SAR c) SHR d) SHL

VII Вариант

1. Какой из следующих команд можно выполнить умножение на два?
a) SAL b) SAR c) SHL d) SHR
2. Какой префикс повторений обеспечивает работу цикла пока выполняется условие?
a) REP b) REPE c) REPNE d) REPNZ
3. Укажите, какой флаг в регистре флагов указывает знак результата при действии со знаками?
a) ZF b) TF c) DF d) SF
4. Макрорасширением называется.....
a) при трансляции программы макроопределение, в котором формирование аргументы замены на фактические

- b) макроопределение формальными аргументами
- c) процедура в программе
- d) макрос без аргументов

5. Что выполняет следующая последовательность команд?

```
CLD
LEA SI, SOUR
LEA DI, ES: DEST
MOV CX, 100
REP MOVSB DEST, SOUR
```

- a) копирует 100 байтов из строки SOUR находящийся в сегменте данных в DEST
- b) сопоставляет строку-источник находящийся в дополнительном сегменте со строкой – приемником
- c) копирует 100 байтов из SOUR в DEST, где обе строки находятся в сегменте данных
- d) копирует 100 байтов из SOUR в DEST, где обе строки находятся в дополнительном сегменте

VIII Вариант

1. Действие сложение по модулю 2 выполняет команда.....

- a) OR
- b) AND
- c) XOR
- d) NOT

2. Какая директива обеспечивает чётность счетчика адресов, добавляя к нему 1, если значение чётное и оставляет без изменения в противном случае.

- a) INCLUDE
- b) GROUP
- c) EVEN
- d) ORG

3. К какой группе команд относятся JMP?

- a) безусловной
- b) условной
- c) логической
- d) команде сдвигов

4. С каким регистром работает команда сканирования SCASB?

- a) BL
- b) DL
- c) AL
- d) DS

5. Что выполняет следующая последовательность команд?

```
MOV CX, DI
P3: DEC DI
MOV DX, buf[DI]
MOV AH, 02H
INT 21h
LOOP P3
```

посимвольный вывод массива buf

- a) посимвольный ввод массива buf
- b) вывод строки
- c) формирование массива

IX Вариант

1. К какому режиму адресации относятся следующая запись: MOV AX, 5

- a) прямая
- b) непосредственная
- c) регистровая
- d) индексная

2. Какая из следующих команд меняет знак операнда на противоположный?

- a) NEG
- b) XLAT
- c) AAA
- d) XCHG

3. Какая из следующих команд передает управление на метку, если выполняется условие?

- a) LOOP
- b) LOOPNE
- c) LOOPE
- d) LOOPNZ

4. Какой функцией можно выполнить вывод строки на экран?

- a) 09h
- b) 01h
- c) 02h
- d) 10h

5. Что выполнит следующая последовательность команд?

```
MOV AH, DH
and ah, 00000001b
cmp ah, 0
```

```

je A1
mov dl, '1'
jmp A2
A1: mov dl, '0'
mov ah, 2h
int 21h

```

- a) вывод 0 или 1 путём логического умножения
 b) ввод 0 или 1 путём логического умножения
 c) вывод 0 или 1 путём логического сложения
 d) ввод 0 или 1 путём логического сложения

X Вариант

- Какой парой регистров характеризуется строка приемник?

a) ES, DI b) DS, SI c) ES, SI d) DS, DI
- К какой группе операций относятся SEG, OFFSET

a) операции присваивается атрибутов b) логические
c) операции отношения d) операции возвращения значения
- MOV AH, 4ch
Int 21h выполняет действие.....

a) завершение программы b) вывод строки
c) вывод символа d) установка позиции курсора
- Следующая команда RET является.....

a) двухадресной b) безадресной
c) одноадресной d) трехадресной
- Какое действие выполняет следующая запись
MOV DX, ((C GE 10) AND 3) OR ((C LT 10) AND 5)

d) загружает значение C в регистр DX если оно > или равно 10
e) загружает значение 3 в DX, если $C \geq 10$ и 5 в противном случае
f) загружает значение 3 в регистр DX если $C \leq 10$ и 5 в противном случае
g) загружает значение регистра DX в C если $DX \geq 10$

Ответы:

I	1.a)	2.c)	3.b)	4b)	5d)
II	1b)	2b)	3b)	4a)	5c)
III	1b)	2a)	3c)	4b)	5b)
IV	1a)	2d)	3c)	4a)	5b)
V	1a)	2d)	3d)	4b)	5a)
VI	1d)	2c)	3a)	4d)	5b)
VII	1a)	2b)	3d)	4a)	5a)
VIII	1c)	2c)	3a)	4c)	5a)
IX	1a)	2a)	3c)	4a)	5a)
X	1a)	2d)	3a)	4b)	5

Перечень заданий для самостоятельной работы и проведения текущего контроля:

- Аппаратные и программные средства современной ЭВМ?
- Представление данных и команд в машинном слове ЭВМ
- Директивы определения данных и распределения памяти
- Составление программ на Ассемблере для выполнения действий с повышенной точностью.
- Программирование преобразований чисел из ASCII кодов в двоичную систему и обратно.
- Двоично-десятичные преобразования.
- Программирование операций ввода и вывода данных
- Выполнение индивидуального задания

Вопросы к зачету:

1. Режимы адресации (привести примеры).
2. Какие регистры входят в состав процессора?
3. Формат и представление данных в машине.
4. Представление чисел в двоично-десятичной системе счисления.
5. Представление данных в ASCII кодах.
6. Директивы выделения памяти.
7. Логические команды, их особенность.
8. Логические, арифметические, циклические сдвиги.
9. Привести примеры одноадресной, двухадресной и безадресной команд.
10. Какими командами можно организовать в программе циклы и разветвления?
11. Каким образом работают команды XOR, OR, AND, NOT?
12. Какому арифметическому действию соответствует арифметический сдвиг влево?
13. Какому арифметическому действию соответствует арифметический сдвиг вправо?
14. Перевод чисел из одной системы счисления в другую.
15. Представление данных в упакованном и в неупакованном форматах.
16. Что такое маска и для чего она используется?
17. Команды работы со строками, их особенность.
18. Какие регистры используются в качестве указателей адресов, символов.
19. Какой флаг устанавливает направление обработки строк?
20. Каким образом модифицируются значения регистров SI, DI при обработке строк?
21. Каким образом работают команды :SCAS, LODS, STOS?
22. В чем особенность команд работы со строками?
23. Для чего используется дополнительный сегмент?
24. Сколько байт обычно выделяется для стека?
25. Макрокоманды (привести примеры).
26. Макроопределение. Формат макроопределения.
27. Чем отличается макроопределение от процедуры?
28. В каких случаях целесообразней использовать макроопределение?
29. Что такое макрорасширение?
30. Влияет ли макроопределение на объем программы, когда и каким образом?
31. Каким образом производится вызов макрокоманды?
32. Формальные и фактические аргументы.
33. Может ли количество формальных аргументов превышать количество фактических?
34. Может ли количество фактических аргументов превышать количество формальных?
35. Директива EXITM.
36. Логические операции и их применение в выражениях.
37. Условное ассемблирование.
38. Технология отладки программ.
39. Процедуры и их использование при решениях задач.
40. Библиотека объектных модулей.
41. Ввод и вывод данных.

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Основная литература

1. Бройдо В.Л., Ильина О.П. «Архитектура ЭВМ и систем», учеб. для вузов., 2-е изд., М., СПб. и др., Питер, 2009.
2. Вахлаева К.П. «Интернет-пособие для проведения лабораторных занятий «часть 1», <http://course.sgu.ru/course/view.php?id=36>
3. Вахлаева К.П. «Интернет-пособие для проведения лабораторных занятий «часть 2», <http://course.sgu.ru/course/view.php?id=20>
4. Калашников О.А. «Ассемблер? Это просто! Учимся программировать», СПб., БХВ-Петербург, 2007 365, +1 эл. опт. диск (CD-ROM) (в медиазале) + в электронном виде на сайте <http://www.kalashnikoff.ru/Assembler>
5. Марек Р. Ассемблер на примерах Базовый курс, СПб., Наука и техника, 2005, 240с.

6. Федорова А.Г. Электронный учебник «Основы программирования на языке Ассемблер для процессора INTEL» на сервере <http://course.sgu.ru/course/view.php?id=268>

8.2. Дополнительная литература

1. SASecurity gr., BarMentalisk . «Ассемблер под Windows для Ч.» <http://sa-sec.org/?cat=11&paged=3>
2. Tomasz Grysztar. «Flat Assembler Programmer's Manual», Официальный сайт FASM, <http://flatassembler.net/docs.php?article=manual>
3. Зубков С.В. «Assembler для DOS, Windows и UNIX», 3-е издание, М. ДМК Пресс, СПб. Питер, 2005
4. Пильщиков В.Н «Программирование на языке ассемблера IBM PC» : учеб. пособие, М., Диалог-Мифи, 2008.
5. Цикл статей «Уроки по Win32 API», <http://www.wasm.ru/series.php?sid=1>
6. Цилькер Б.Я., Орлов С.А. «Организация ЭВМ и систем», учебник, М., СПб. и др., Питер, 2007.
7. Юров В.И. «Assembler», 2-е изд., М., СПб. и др., Питер, 2008.

8.3. Программное обеспечение и Интернет-ресурсы

Пакет разработчика MASM32. Отладочный комплекс OlyDebug

9. Материально-техническое обеспечение дисциплины

Учебная аудитория (наличие доски обязательно), оснащенная оргтехникой (для проведения отдельных лекций с использованием электронных презентаций необходимо наличие проектора и компьютерной техники). Для проведения лабораторных работ требуется компьютерный класс, в котором в процессе выполнения работ студенты устанавливают и осваивают пакеты Masm32, OlyDebug.

10. Методические рекомендации по организации изучения дисциплины

Изучаемый теоретический материал в рамках дисциплины «Машинно-зависимые языки программирования» носит практическую значимость и используется при подготовке бакалавров по направлению 2.09.03.04 – «Программная инженерия».

При преподавании курса необходимо ориентироваться на современные образовательные технологии. Аудиторная и самостоятельная работы должны быть направлены на углубление и расширение полученных знаний, на закрепление приобретенных навыков и применение формируемых компетенций. Кроме того, рекомендуется использовать дифференцированное обучение и активные методы проверки знаний при проведении проверочных работ, тестирования. Это достигается, например, путем организации индивидуальной самостоятельной работы студентов.

Для полноценного прохождения дисциплины полезны циклы заданий по всем разделам дисциплины и банк контрольных задач, чтобы самостоятельная работа студентов была спланирована сразу на весь семестр. Это представляется важным и как средство текущего контроля, организующего учебную работу в аудитории и особенно при самостоятельном изучении и освоении разделов дисциплин.

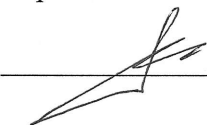
При проведении промежуточной аттестации, независимо от формы ее проведения (устной или письменной), важно учесть все виды работ, оценить уровень знаний студентов по всем разделам учебной дисциплины.

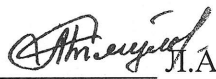
Примерный перечень вопросов к итоговой форме контроля должен доводиться до студентов в начале изучения дисциплины. При необходимости он может быть уточнен не позднее, чем за месяц до начала зачетной сессии.

Рабочая учебная программа по дисциплине «Машинно-зависимые языки программирования» составлена в соответствии с требованиями Федерального Государственного образовательного стандарта ВО по направлению 2.09.03.04 – «Программная инженерия» и учебного плана по профилю подготовки «Разработка программно-информационных систем».

11. Технологическая карта дисциплины

Модульно-рейтинговая система не введена.

Составитель  А.Б. Глазов, ст. преподаватель

Зав. кафедрой информатики и программной инженерии  И.А. Тягульская, доцент