

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Организация ЭВМ и систем»
Тема: «Трансляция, отладка и выполнение программ на языке
Ассемблера»

Студентка гр. 1303

Куклина Ю.Н.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Написать программу, которая считает высоту дерева, путем их обхода.

Задание.

Часть 1

1. Просмотреть программу `hello1.asm`, которая формирует и выводит на экран приветствие пользователя с помощью функции ОС MSDOS, вызываемой через прерывание с номером `21H` (команда `Int 21h`). Выполняемые функцией действия и задаваемые ей параметры - следующие: - обеспечивается вывод на экран строки символов, заканчивающейся знаком `"$"`; - требуется задание в регистре `ah` номера функции, равного `09h`, а в регистре `dx` - смещения адреса выводимой строки; - используется регистр `ax` и не сохраняется его содержимое.
2. Разобраться в структуре и реализации каждого сегмента программы. Строку-приветствие преобразовать в соответствии со своими личными данными.
3. Загрузить файл `hello1.asm` из каталога Задания в каталог Masm.
4. Протранслировать программу с помощью строки `> masm hello1.asm` с созданием объектного файла и файла диагностических сообщений (файла листинга). Объяснить и исправить синтаксические ошибки, если они будут обнаружены транслятором. Повторить трансляцию программы до получения объектного модуля.
5. Скомпоновать загрузочный модуль с помощью строки `> link hello1.obj` с созданием карты памяти и исполняемого файла `hello1.exe`.
6. Выполнить программу в автоматическом режиме путем набора строки `> hello1.exe` убедиться в корректности ее работы и зафиксировать результат выполнения в протоколе.
7. Запустить выполнение программы под управлением отладчика с помощью команды `> afd hello1.exe`

Записать начальное содержимое сегментных регистров CS, DS, ES и SS. Выполнить программу в пошаговом режиме с фиксацией используемых регистров и ячеек памяти до и после выполнения каждой команды. Результаты прогона программы под управлением отладчика должны быть представлены в виде, показанном на примере одной команды в табл.1, и подписаны преподавателем. Табл.1

Часть 2

Выполнить пункты 1 - 7 части 1 настоящего задания применительно к программе hello2.asm, приведенной в каталоге Задания, которая выводит на экран приветствие пользователя с помощью процедуры WriteMsg, а также использует полное определение сегментов. Сравнить результаты прогона под управлением отладчика программ hello1 и hello2 и объяснить различия в размещении сегментов.

Исходные тексты программ hello1.asm и hello2.asm.

1. hello1.asm

```

DOSSEG                                ; Задание сегментов под ДОС
.MODEL SMALL                          ; Модель памяти-SMALL (Малая)
.STACK 100h                           ; Отвести под Стек 256 байт
.DATA                                 ; Начало сегмента данных
Greeting LABEL BYTE                  ; Текст приветствия
DB 'Вас приветствует ст.гр.1303 - Куклина Ю.Н.',13,10,'$'
.CODE                                ; Начало сегмента кода
mov ax, @data                        ; Загрузка в DS адреса начала
mov ds, ax                           ; сегмента данных
mov dx, OFFSET Greeting              ; Загрузка в dx смещения
                                     ; адреса текста приветствия

DisplayGreeting:
mov ah, 9                            ; # функции ДОС печати строки
int 21h                              ; вывод на экран приветствия
mov ah, 4ch                          ; # функции ДОС завершения
                                     ; завершения программы и выход
int 21h
в ДОС
END

```

2. hello2.asm

```

1. EOFLine EQU '$'          ; Определение символьной константы
                               ; "Конец строки"

; стек программы

ASSUME CS:CODE, SS:AStack

AStack    SEGMENT STACK
          DW 12 DUP('!')    ; Отводится 12 слов памяти
AStack    ENDS

; Данные программы

DATA      SEGMENT

; Директивы описания данных

HELLO     DB 'Hello Worlds!', 0AH, 0DH, EOFLine
GREETING  DB 'Student from 1303 - Kuklina Yulia$.'
DATA      ENDS

; Код программы

CODE      SEGMENT
; Процедура печати строки
WriteMsg  PROC NEAR
          mov     AH, 9
          int     21h ; Вызов функции DOS по прерыванию
          ret
WriteMsg  ENDP

; Головная процедура
Main      PROC FAR
          push    DS          ; \ Сохранение адреса начала PSP в стеке
          sub     AX, AX      ; > для последующего восстановления по
          push    AX          ; / команде ret, завершающей процедуру.
          mov     AX, DATA    ; Загрузка сегментного
          mov     DS, AX       ; регистра данных.
          mov     DX, OFFSET HELLO ; Вывод на экран первой
          call    WriteMsg     ; строки приветствия.
          mov     DX, OFFSET GREETING ; Вывод на экран второй
          call    WriteMsg     ; строки приветствия.
          ret                ; Выход в DOS по команде,
                               ; находящейся в 1-ом слове PSP.

Main      ENDP
CODE      ENDS
          END Main

```

Тексты файлов диагностических сообщений hello1.lst и hello2.lst;

1. hello1.lst

Microsoft (R) Macro Assembler Version 5.10
19:43:32

9/15/22

Page 1-

1

```
; HELLO1.ASM - упрощенная версия учебной программы лаб.раб. N1 ;
по дисциплине "Архитектура компьютера";
*****
; Назначение: Программа формирует и выводит на экран приветствие
; пользователя с помощью функции ДОС "Вывод строки"
; (номер 09 прерывание 21h), которая:
; - обеспечивает вывод на экран строки символов,;
заканчивающейся знаком "$";
; - требует задания в регистре ah номера функции=09h,;
а в регистре dx - смещения адреса выводимой; строки;
; - использует регистр ax и не сохраняет его;
содержимое.
; *****
```

```
DOSSEG
; Задание сегментов под ДОС.MODEL SMALL
; Модель памяти-SMALL(Малая)
.STACK 100h ; Отвести под Стек 256 байт
.DATA ; Начало сегмента данных
0000 Greeting LABEL BYTE ; Текст приветствия
0000 C2 E0 F1 20 EF F0 DB 'Вас приветствует ст.гр.1303 -
Куклина Ю.Н.
```

```
' ,13,10,'$'
E8 E2 E5 F2 F1 F2
E2 F3 E5 F2 20 F1
F2 2E E3 F0 2E 31
33 30 33 20 2D 20
CA F3 EA EB E8 ED
E0 20 DE 2E CD 2E
0D 0A 24
.CODE ; Начало сегмента кода
0000 B8 ---- R mov ax, @data ; Загрузка в DS адреса начала
0003 8E D8 mov ds, ax сегмента данных
0005 BA 0000 R mov dx, OFFSET Greeting ; Загрузка в dx
смещения
Microsoft (R) Macro Assembler Version 5.10 9/15/22
19:43:32
```

Page 1-

2

```
; адрес
а текста приветствия
0008 DisplayGreeting:
0008 B4 09 mov ah, 9 ; #
функции ДОС печати строки
000A CD 21 int 21h ;
вывод на экран приветствия
000C B4 4C mov ah, 4ch ; #
функции ДОС завершения программы
000E CD 21 int 21h ;
завершение программы и выход в ДОС
END
Microsoft (R) Macro Assembler Version 5.10 9/15/22
19:43:32
```

Symbols-1

Segments and Groups:

	N a m e	Length	Align	Combine	Class
DGROUP	GROUP			
_DATA	002B	WORD	PUBLIC	'DATA'
_STACK	0100	PARA	STACK	'STACK'
_TEXT	0010	WORD	PUBLIC	'CODE'

Symbols:

	N a m e	Type	Value	Attr
DISPLAYGREETING	L NEAR	0008	_TEXT
GREETING	L BYTE	0000	_DATA
@CODE	TEXT	_TEXT	
@CODESIZE	TEXT	0	
@CPU	TEXT	0101h	
@DATASIZE	TEXT	0	
@FILENAME	TEXT	hello1	
@VERSION	TEXT	510	

33 Source Lines
33 Total Lines
19 Symbols

47998 + 461309 Bytes symbol space free

0 Warning Errors
0 Severe Errors

2. hello2.lst

Microsoft (R) Macro Assembler Version 5.10
20:00:07
Page 1-1

9/15/22

```

;
HELLO2 - Учебная программа лаб.раб.#1 по дисциплине "Архитектура
компьютера";
Программа использует процедуру для печати строки;
;      ТЕКСТ ПРОГРАММЫ

= 0024                      EOFLine EQU '$'          ; Определение
символьной константы;      "Конец строки"

; Стек программы

ASSUME CS:CODE, SS:AStack

```

```

0000          AStack      SEGMENT  STACK
0000 000C[
DW 12 DUP('!')      ; Отводится 12 слов памяти
0021      ]

0018          AStack      ENDS

; Данные программы

0000          DATA       SEGMENT

; Директивы описания данн
ых

0000 48 65 6C 6C 6F 20  HELLO      DB 'Hello Worlds!', 0AH, 0DH, EOFLine
57 6F 72 6C 64 73
21 0A 0D 24
0010 53 74 75 64 65 6E  GREETING  DB 'Student from 4350 - $'
74 20 66 72 6F 6D
20 34 33 35 30 20
2D 20 24

0025          DATA       ENDS

; Код программы

0000          CODE        SEGMENT
; Процедура печати строки
0000 WriteMsg  PROC  NEAR
0000 B4 09                      mov     AH, 9
0002 CD 21                      int     21h ; Вызов функции
DOS по прерыванию
0004 C3                          ret
0005 WriteMsg  ENDP

```

; Головная процедура
Microsoft (R) Macro Assembler Version 5.10
20:00:07

9/15/22

Page 1-

2

```

0005          Main       PROC  FAR
0005 1E                      push    DS ; \ Сохранени
е адреса начала PSP в стеке
0006 2B C0                      sub     AX, AX ; > для послед
ующего восстановления по
0008 50                      push    AX ; / команде ret
, завершающей процедуру.
0009 B8 ---- R                      mov     AX, DATA ; Загру
зка сегментного
000C 8E D8                      mov     DS, AX ; реги
тра данных.
000E BA 0000 R                      mov     DX, OFFSET HELLO ; Вывод
на экран первой
0011 E8 0000 R                      call    WriteMsg ; строк

```

```

                                и приветствия.
0014  BA 0010 R                mov    DX, OFFSET GREETING ; Вывод
                                на экран второй
0017  E8 0000 R                call   WriteMsg           ; строк
                                и приветствия.
001A  CB                      ret                       ; Выход
                                в DOS по команде,
                                                ; находящейся
в 1-ом слове PSP.
001B                          Main      ENDP
001B                          CODE      ENDS
                                END Main
Microsoft (R) Macro Assembler Version 5.10          9/15/22
20:00:07

```

Symbols-1

Segments and Groups:

N a m e	Length	Align	Combine	Class
ASTACK	0018	PARA	STACK	
CODE	001B	PARA	NONE	
DATA	0025	PARA	NONE	

Symbols:

N a m e	Type	Value	Attr
EOFLINE	NUMBER	0024	
GREETING	L BYTE	0010	DATA
HELLO	L BYTE	0000	DATA
MAIN	F PROC	0005	CODE Length = 0016
WRITEMSG	N PROC	0000	CODE Length = 0005
@CPU	TEXT	0101h	
@FILENAME	TEXT	hello2	
@VERSION	TEXT	510	

```

52 Source  Lines
52 Total   Lines
13 Symbols

```

47990 + 459267 Bytes symbol space free

```

0 Warning Errors
0 Severe  Errors

```


Протоколы пошагового исполнения каждой из программ под управлением отладчика.

Таблица 1- Протокол пошагового исполнения программы hello1.asm

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0000	MOV AX, 1A07	B8071A	(AX) = 0000 (DS) = 19F5 (IP) = 0010 (DX) = 0000 (SP) = 0100	(AX) = 1A07 (DS) = 19F5 (IP) = 0013 (DX) = 0000 (SP) = 0100
0003	MOV DS,AX	8ED8	(AX) = 1A07 (DS) = 19F5 (IP) = 0013 (DX) = 0000 (SP) = 0100	(AX) = 1A07 (DS) = 1A07 (IP) = 0015 (DX) = 0000 (SP) = 0100
0005	MOV DX,0000	BA0000	(AX) = 1A07 (DS) = 1A07 (IP) = 0015 (DX) = 0000 (SP) = 0100	(AX) = 1A07 (DS) = 1A07 (IP) = 0018 (DX) = 0000 (SP) = 0100
0008	MOV AH,09	B409	(AX) = 1A07 (DS) = 1A07 (IP) = 0018 (DX) = 0000 (SP) = 0100	(AX) = 0907 (DS) = 1A07 (IP) = 001A (DX) = 0000 (SP) = 0100
000A	INT 21	CD21	(AX) = 0907 (DS) = 1A07 (IP) = 001A (DX) = 0000	(AX) = 0907 (DS) = 1A07 (IP) = 001C (DX) = 0000

			(SP) =0100	(SP) =0100
000C	MOV AH,4C	B44C	(AX) = 0907 (DS) = 1A07 (IP) = 001C (DX) = 0000 (SP) =0100	(AX) = 4C07 (DS) = 1A07 (IP) = 001E (DX) = 0000 (SP) =0100
000E	INT 21	CD21	(AX) = 4C07 (DS) = 1A07 (IP) = 001E (DX) = 0000 (SP) =0100	(AX) = 0000 (DS) = 19F5 (IP) = 0010 (DX) = 0000 (SP) =0100

Таблица 2- Протокол пошагового исполнения программы hello2.asm

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0005	PUSH DS	1E	(AX) = 0000 (DS) = 19F5 (IP) = 0005 (DX) = 0000 (SP) =0018	(AX) = 0000 (DS) = 19F5 (IP) = 0006 (DX) = 0000 (SP) =0016
0006	SUB AX,AX	2BC0	(AX) = 0000 (DS) = 19F5 (IP) = 0005 (DX) = 0000 (SP) =0016	(AX) = 0000 (DS) = 19F5 (IP) = 0008 (DX) = 0000 (SP) =0016
0008	PUSH AH	50	(AX) = 0000 (DS) = 19F5 (IP) = 0008 (DX) = 0000 (SP) =0016	(AX) = 0000 (DS) = 19F5 (IP) = 0009 (DX) = 0000 (SP) =0014

0009	MOV AH,1A07	B8071A	(AX) = 0000 (DS) = 19F5 (IP) = 0009 (DX) = 0000 (SP) =0014	(AX) = 1A07 (DS) = 19F5 (IP) = 000C (DX) = 0000 (SP) =0014
000C	MOV DS,AX	8ED8	(AX) = 1A07 (DS) = 19F5 (IP) = 000C (DX) = 0000 (SP) =0014	(AX) = 1A07 (DS) = 1A07 (IP) = 000E (DX) = 0000 (SP) =0014
000E	MOV DX,0000	BA0000	(AX) = 1A07 (DS) = 1A07 (IP) = 000E (DX) = 0000 (SP) =0014	(AX) = 1A07 (DS) = 1A07 (IP) = 0011 (DX) = 0000 (SP) =0014
0011	CALL 0000	E8ECFF	(AX) = 1A07 (DS) = 1A07 (IP) = 0011 (DX) = 0000 (SP) =0014	(AX) = 0907 (DS) = 1A07 (IP) = 0014 (DX) = 0000 (SP) =0014
0014	MOV DX,0010	BA1000	(AX) = 0907 (DS) = 1A07 (IP) = 0014 (DX) = 0000	(AX) = 0907 (DS) = 1A07 (IP) = 0017 (DX) = 0010
0017	CALL 0000	E8E6FF	(AX) = 0907 (DS) = 1A07 (IP) = 0017 (DX) = 0010	(AX) = 0907 (DS) = 1A07 (IP) = 001A (DX) = 0010
001A	RET Far	CB	(AX) = 0907 (DS) = 1A07 (IP) = 001A (DX) = 0010 (SP) =0014	(AX) = 0907 (DS) = 1A07 (IP) = 0000 (DX) = 0010 (SP) =0018

Протокол работы на компьютере программы hello2.asm

1. На «рис.1» выполнен монтаж директории, протранслированна программа с созданием объектного файла и файла диагностических сообщений.

```
Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>MOUNT C "C:\ass1"
Drive C is mounted as local directory C:\ass1\

Z:\>C:

C:\>masm hello2.asm
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.

Object filename [hello2.OBJ]:
Source listing [NUL.LST]: hello2
Cross-reference [NUL.CRF]:

47986 + 459271 Bytes symbol space free

0 Warning Errors
0 Severe Errors
```

Рисунок1

2. На «рис.2» выполнена компоновка загрузочного модуля с созданием карты памяти и исполняемого файл hello2.exe.

```
C:\>link hello2.obj

Microsoft (R) Overlay Linker Version 3.64
Copyright (C) Microsoft Corp 1983-1988. All rights reserved.

Run File [HELLO2.EXE]:
List File [NUL.MAP]:
Libraries [LIB]:
```

Рисунок2

3. На «рис.3» корректно выполнена программа в автоматическом режиме.

```
C:\>hello2.exe
Hello Worlds!
Student from 1303 - Kuklina Yulia
C:\>
```

Рисунок3

Протокол работы на компьютере программы hello1.asm

1. На «рис.4» выполнен монтаж директории, протранслированна программа с созданием объектного файла и файла диагностических сообщений.

```
Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>MOUNT C "C:\ass1"
Drive C is mounted as local directory C:\ass1\

Z:\>C:

C:\>masm hello1.asm
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.

Object filename [hello1.OBJ]:
Source listing [NUL.LST]: LST2
Cross-reference [NUL.CRF]:

47998 + 461309 Bytes symbol space free

0 Warning Errors
0 Severe Errors
```

Рисунок4

2. На «рис.5» выполнена компоновка загрузочного модуля с созданием карты памяти и исполняемого файл hello1.exe.

```
C:\>link hello1.obj

Microsoft (R) Overlay Linker Version 3.64
Copyright (C) Microsoft Corp 1983-1988. All rights reserved.

Run File [HELLO1.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
```

Рисунок5

3. На «рис.6» корректно выполнена программа в автоматическом режиме.

```
C:\>hello1.exe
Вас приветствует ст.гр.1303 – Куклина Ю.Н.
```

Рисунок6

Выводы.

Разобрались в структуре и реализации каждого сегмента программ hello1.asm и hello2.asm. Научились работать с отладчиком AFDPRO, запускать программы на языке ассемблера, работая в эмуляторе DOSBox.