

**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**

Студент гр. 0383

\_\_\_\_\_

Сабанов П.А.

Преподаватель

\_\_\_\_\_

Ефремов М.А.

Санкт-Петербург

2021

### **Цель работы.**

Изучить работу и представление в памяти программ `hello1.asm` и `hello2.asm`, написанных на языке ассемблера, научиться пользоваться компилятором `masm`, компоновщиком `link` и отладчиком `afd`.

### **Текст задания.**

#### 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

и 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. Обычные команды выполняются по F1 (Step), а вызовы обработчиков прерываний (Int) - по F2 (StepProc), чтобы не входить внутрь обработчика прерываний.

Продвижение по сегментам экранной формы отладчика выполняется с помощью клавиш F7 – F10 (up, down, left, right). Перезапуск программы в отладчике выполняется клавишей F3 (Retrieve). Выход из отладчика - по команде Quit.

Результаты прогона программы под управлением отладчика должны быть представлены в виде, показанном на примере одной команды в табл.1, и подписаны преподавателем.

Табл.1

Адрес Команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			до выполнения .	После выполнения
0003	Mov DS, AX	8E D8	(AX) = 2D87 (DS) = 2D75 (IP) = 0003	(AX) = 2D87 (DS) = 2D87 (IP) = 0005

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.

### **Основные теоретические положения.**

MASM (Microsoft Macro Assembler) — ассемблер для процессоров семейства x86. Первоначально был произведён компанией Microsoft для написания программ в операционной системе MS-DOS и был в течение некоторого времени самым популярным ассемблером, доступным для неё.

### **Ход работы.**

Обработка программы hello1.

#### 1. Ввод в консоль команды

> masm hello1

с последующим указанием имени файла листинга. Компиляция завершилась без ошибок с созданием файла hello1.obj.

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

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

    47994 + 461313 Bytes symbol space free

    0 Warning Errors
    0 Severe Errors
```

#### 2. Ввод в консоль команды

> link hello1

с последующим указанием имени файла карты памяти. Компиляция прошла без ошибок с созданием файла hello1.exe.

```
C:\>link hello1
Microsoft (R) Overlay Linker Version 3.64
Copyright (C) Microsoft Corp 1983-1988. All rights reserved.

Run File [HELLO1.EXE]:
List File [NUL.MAP]: hello1.map
Libraries [.LIB]:
```

### 3. Ввод в консоль команды

> hello1

Вывод программы hello1:

```
C:\>hello1
Peter Sabanov, a student of group 0383, welcomes you.
```

Программа отработала без ошибок.

### 4. Запуск программы hello1 под управлением отладчика afd:

> afdpro hello1

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			до выполнения	после выполнения
0010	mov ax, @data (mov ax, 1A07)	B8071A	(AX) = 0000 (IP) = 0010	(AX) = 1A07 (IP) = 0013
0013	mov ds, ax	8ED8	(DS) = 19F5 (IP) = 0013	(DS) = 1A07 (IP) = 0015
0015	mov dx, offset Greeting (mov dx, 0000)	BA0000	(DX) = 0000 (IP) = 0015	(DX) = 0000 (IP) = 0018
0018	mov ah, 09	B409	(AX) = 1A07 (IP) = 0018	(AH) = 0907 (IP) = 001A
001A	int 21	CD21	(IP) = 001A	(IP) = 001C
001C	mov ah, 4C	B44C	(AX) = 0907 (IP) = 001C	(AX) = 4C07 (IP) = 001E
001E	int 21	CD21		

Сообщение от отладчика:

> Program terminated OK

Обработка программы hello2.

1. Аналогичным образом для программы hello2 создаются файлы hello2.obj, hello2.lst, hello2.ext и hello2.map:

```
C:\>masm hello2
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]:

48002 + 461305 Bytes symbol space free

0 Warning Errors
0 Severe Errors
```

```
C:\>link hello2

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

Run File [HELLO2.EXE]:
List File [NUL.MAP]: hello2
Libraries [.LIB]:
```

## 2. Запуск программы hello2 путём ввода команды

> hello2

Вывод программы hello2:

```
C:\>hello2
Hello Worlds!
Student from 0383 - Sabanov Peter
```

Программа отработала без ошибок.

## 3. Запуск программы hello1 под управлением отладчика afd:

> afdpro hello2

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			до выполнения	после выполнения
0005	push ds	1E	(SP) = 0018 (IP) = 0005 [SP] = 0000	(SP) = 0016 (IP) = 0006 [SP] = 19F5
0006	sub ax, ax	2BC0	(AX) = 0000 (IP) = 0006 [SP] = 19F5 [SP+2] = 0000	(AX) = 0000 (IP) = 0008 [SP] = 0000 [SP+2] = 19F5
0008	push ax	50	(SP) = 0016 (IP) = 0008	(SP) = 0014 (IP) = 0009
0009	mov ax, DATA (mov ax, 1A07)	B8071A	(AX) = 0000 (IP) = 0009	(AX) = 1A07 (IP) = 000C
000C	mov ds, ax	8ED8	(DS) = 19F5 (IP) = 000C	(DS) = 1A07 (IP) = 000E
000E	mov dx, offset HELLO (mov dx, 0000)	BA0000	(DX) = 0000 (IP) = 000E	(DX) = 0000 (IP) = 0011
0011	call WriteMsg	E8ECFF	(SP) = 0014 (IP) = 0011 [SP] = 0000 [SP+2] = 19F5 [SP+4] = 0000	(SP) = 0012 (IP) = 0000 [SP] = 0014 [SP+2] = 0000 [SP+4] = 19F5
0000	mov ah, 09	B409	(AX) = 1A07 (IP) = 0000	(AX) = 0907 (IP) = 0002
0002	int 21	CD21	(IP) = 0002	(IP) = 0004

0004	ret	C3	(SP) = 0012 (IP) = 0004 [SP] = 0014 [SP+2] = 0000 [SP+4] = 19F5	(SP) = 0012 (IP) = 0014 [SP] = 0000 [SP+2] = 19F5 [SP+4] = 0000
0014	mov dx, offset GREETING (mov dx, 0010)	BA1000	(DX) = 0000 (IP) = 0014	(DX) = 0010 (IP) = 0017
0017	call WriteMsg	E8E6FF	(SP) = 0014 (IP) = 0017 [SP] = 0000 [SP+2] = 19F5 [SP+4] = 0000	(SP) = 0012 (IP) = 0000 [SP] = 001A [SP+2] = 0000 [SP+4] = 19F5
0000	mov ah, 09	B409	(AX) = 0907 (IP) = 0000	(AX) = 0907 (IP) = 0002
0002	int 21	CD21	(IP) = 0002	(IP) = 0004
0004	ret	C3	(SP) = 0012 (IP) = 0004 [SP] = 001A [SP+2] = 0000 [SP+4] = 19F5	(SP) = 0014 (IP) = 001A [SP] = 0000 [SP+2] = 19F5 [SP+4] = 0000
001A	ret (ret far)	CB	(SP) = 0014 (CS) = 1A0B (IP) = 001A [SP] = 0000 [SP+2] = 19F5	(SP) = 0018 (CS) = 19F5 (IP) = 0000 [SP] = 0000 [SP+2] = 0000
0000	int 20	CD20		

Сообщение от отладчика:

> Program terminated OK

Различия в прогонах программ hello1 и hello2 с помощью отладчика заключаются в том, что в программе hello1 не использовались вызовы других функций в то время как в hello2 использовался вызов функции *WriteMsg* с помощью команды *CALL*, из-за чего приходилось использовать команду Step отладчика (клавиша F1), чтобы войти в функцию, а не перескочить через неё.

Различия в размещении сегментов в программах hello1 и hello2 в том, что в программе hello1 сегменты размещаются без выравнивания, а в hello2 сегменты выравниваются по 16 байт. Это можно увидеть на следующих гексдампах:

*хексдамн hello1.exe:*

```

00000000: 4d5a 5800 0200 0100 2000 1100 ffff 0600 MZX.....
00000010: 0001 1c69 1000 0000 1e00 0000 0100 1100 ...i.....
00000020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000030: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000040: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000070: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000080: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000090: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000a0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000b0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000c0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000d0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000e0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000f0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000100: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000110: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000120: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000130: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000140: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000150: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000160: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000170: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000180: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000190: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001a0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001b0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001c0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001d0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001e0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001f0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000200: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000210: b802 008e d8ba 0000 b409 cd21 b44c cd21 .....!.L.!
00000220: 5065 7465 7220 5361 6261 6e6f 762c 2061 Peter Sabanov, a
00000230: 2073 7475 6465 6e74 206f 6620 6772 6f75 student of grou
00000240: 7020 3033 3833 2c20 7765 6c63 6f6d 6573 p 0383, welcomes
00000250: 2079 6f75 2e0d 0a24 you...$

```



*хексдамн hello2.exe:*

```

00000000: 4d5a 7b00 0200 0100 2000 0000 ffff 0000 MZ{.....
00000010: 1800 4984 0500 0600 1e00 0000 0100 0a00 ..I.....
00000020: 0600 0000 0000 0000 0000 0000 0000 0000 .....
00000030: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000040: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000070: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000080: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000090: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000a0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000b0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000c0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000d0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000e0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000000f0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000100: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000110: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000120: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000130: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000140: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000150: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000160: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000170: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000180: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000190: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001a0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001b0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001c0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001d0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001e0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000001f0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00000200: 2300 2300 2300 2300 2300 2300 2300 2300 #. #. #. #. #. #. #.
00000210: 2300 2300 2300 2300 0000 0000 0000 0000 #. #. #. #. ....
00000220: 4865 6c6c 6f20 576f 726c 6473 210a 0d24 Hello Worlds!..$
00000230: 5374 7564 656e 7420 6672 6f6d 2030 3338 Student from 038
00000240: 3320 2d20 5361 6261 6e6f 7620 5065 7465 3 - Sabanov Pete
00000250: 7224 0000 0000 0000 0000 0000 0000 0000 r$.....

```

```
00000260: b409 cd21 c31e 2bc0 50b8 0200 8ed8 ba00 ...!...+.P.....  
00000270: 00e8 ecff ba10 00e8 e6ff cb .....
```

### **Выводы.**

Была изучена работа программ hello1 и hello2, были получены навыки компиляции, компоновки и отладки программ на ассемблере masm, было проанализировано выполнение программ в отладчике afd. Также были проанализированы хексдампы программ и на основании них различия в размещении их сегментов.

## ПРИЛОЖЕНИЕ А

### Исходный код программы hello1

```
; HELL01.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
Greeting LABEL BYTE   ; Текст приветствия
DB 'Peter Sabanov, a student of group 0383, welcomes you.',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 программы
int 21h                ; завершение программы и выход в ДОС
END
```

## ПРИЛОЖЕНИЕ Б

### Исходный код программы hello2

```
; HELLO2 - Учебная программа N2 лаб.раб.#1 по дисциплине "Архитектура
компьютера"
;      Программа использует процедуру для печати строки
;
;      ТЕКСТ ПРОГРАММЫ

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 0383 - Sabanov Peter$'
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
```

; Главная 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
```

## ПРИЛОЖЕНИЕ В

### Листинг компиляции программы hello1

#Microsoft (R) Macro Assembler Version 5.10

9/12/21 18:07:00

Page 1-1

```
; HELLO1.ASM - упрощенная версия учебной прог  
аммы лаб.раб. N1  
;  
; по дисциплине "Архитектура комп  
ьютера"  
;  
; *****  
*****  
;  
; Назначение: Программа формирует и выводит на  
экран приветствие  
;  
; пользователя с помощью функции ДО  
С "Вывод строки"  
;  
; (номер 09 прерывание 21h), котора  
я:  
;  
; - обеспечивает вывод на экран ст  
роки символов,  
;  
; заканчивающейся знаком "$";  
;  
; - требует задания в регистре ah  
номера функции=09h,  
;  
; а в регистре dx - смещения а  
дреса 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  50 65 74 65 72 20      DB 'Peter Sabanov, a student of group 0383,
                               welcomes you.',13,10,'$'

                53 61 62 61 6E 6F
                76 2C 20 61 20 73
                74 75 64 65 6E 74
                20 6F 66 20 67 72
                6F 75 70 20 30 33
                38 33 2C 20 77 65
                6C 63 6F 6D 65 73
                20 79 6F 75 2E 0D

```

Выводы.

Была изучена работа программ hello1 и hello2, были получены навыки компиляции, компоновки и отладки программ на ассемблере masm, было проанализировано выполнение программ в отладчике afd. Также были проанализированы хексдампы программ и на основании них различия в размещении их сегментов.

```

0A 24

                .CODE                                ; Начал
                о сегмента кода

0000  B8 ---- R      mov ax, @data                    ; Загр
                узка в DS адреса начала

0003  8E D8          mov ds, ax                      ; сегм
                ента данных

#Microsoft (R) Macro Assembler Version 5.10        9/12/21 18:07:00
                                                    Page      1-2

0005  BA 0000 R      mov dx, OFFSET Greeting          ; Загр
                узка в dx смещения

                ; адрес
                а текста приветствия
0008      DisplayGreeting:
0008  B4 09          mov ah, 9                        ; # фу
                нкции ДОС печати строки
000A  CD 21          int 21h                          ; вывод
                на экран приветствия
000C  B4 4C          mov ah, 4ch                      ; # фу
                нкции ДОС завершения программы

```

```

000E  CD 21                                int 21h                                ; завер
                                         шение программы и выход в ДОС
                                         END

```

```
#Microsoft (R) Macro Assembler Version 5.10
```

```
9/12/21 18:07:00
```

```
Symbols-1
```

# Segments and Groups:

N a m e	Length	Align	Combine Class
DGROUP . . . . .	GROUP		
_DATA . . . . .	0038 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

```

```
47994 + 461313 Bytes symbol space free
```

```

0 Warning Errors
0 Severe  Errors

```



## ПРИЛОЖЕНИЕ Г

### Листинг компиляции программы hello2

#Microsoft (R) Macro Assembler Version 5.10

9/12/21 19:20:16

Page 1-1

```

; HELLO2 - Учебная программа N2 лаб.раб.#1 по
; дисциплине "Архитектура компьютера"
; Программа использует процедуру для п
; ечати строки
;
;      ТЕКСТ ПРОГРАММЫ

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

; Стек программы

ASSUME CS:CODE, SS:AStack

0000      AStack      SEGMENT      STACK
0000  000C[      DW 12 DUP('#')      ; Отводится 12 слов
; памяти
;      0023
;      ]

0018      AStack      ENDS

; Данные программы

0000      DATA      SEGMENT

; Директивы описания данных

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 0383 - Sabanov Peter
                                $'
                                74 20 66 72 6F 6D
                                20 30 33 38 33 20
                                2D 20 53 61 62 61
                                6E 6F 76 20 50 65
                                74 65 72 24
0032                                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 пре
                                рыванию
0004  C3                                ret
0005                                WriteMsg ENDP

                                ; Главная процедура
0005                                Main      PROC FAR
#Microsoft (R) Macro Assembler Version 5.10
9/12/21 19:20:16
Page      1-2

```

```

0005  1E                                push    DS          ;\ Сохранение адреса
                                начала PSP в стеке
0006  2B C0                                sub     AX,AX       ; > для последующего в
                                останова по
0008  50                                push    AX          ;/ команде ret, завер
                                шающей процедуру.
0009  B8 ---- R                                mov     AX,DATA      ; Загрузка
                                сегментного
000C  8E D8                                mov     DS,AX        ; регистра
                                данных.
000E  BA 0000 R                                mov     DX, OFFSET HELLO ; Вывод на
                                экран первой
0011  E8 0000 R                                call    WriteMsg     ; строки пр
                                иветствия.
0014  BA 0010 R                                mov     DX, OFFSET GREETING ; Вывод на

```

```

                                экран второй
0017  E8 0000 R                call WriteMsg          ; строки пр
                                иветствия.
001A  CB                      ret                    ; Выход в D
                                OS по команде,
                                                ; находящей
                                ся в 1-ом слове PSP.
001B                          Main      ENDP
001B                          CODE      ENDS
                                END Main
#Microsoft (R) Macro Assembler Version 5.10          9/12/21 19:20:16
                                                Symbols-1

```

#### Segments and Groups:

N a m e	Length	Align	Combine Class
ASTACK . . . . .	0018	PARA	STACK
CODE . . . . .	001B	PARA	NONE
DATA . . . . .	0032	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

48002 + 461305 Bytes symbol space free

0 Warning Errors

0 Severe Errors

## ПРИЛОЖЕНИЕ Д

### Карта памяти программы hello1

Start	Stop	Length	Name	Class
00000H	0001FH	00020H	_TEXT	CODE
00020H	00057H	00038H	_DATA	DATA
00060H	0015FH	00100H	STACK	STACK

Origin	Group
0002:0	DGROUP

## ПРИЛОЖЕНИЕ Е

### Карта памяти программы hello2

Start	Stop	Length	Name	Class
00000H	00017H	00018H	ASTACK	
00020H	00051H	00032H	DATA	
00060H	0007AH	0001BH	CODE	

Program entry point at 0006:0005