

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: Трансляции, отладка и выполнение программ на языке
Ассемблера.

Студент гр. 9382

Герасев Г.А,

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

Изучить основные принципы трансляции, отладки и выполнения программ на языке Ассемблера.

Ход работы.

Часть 1.

1. Загрузил файлы AFDPRO.EXE, HELLO1.ASM, HELLO2.ASM, LINK.EXE, MASM.EXE в каталог ~/Assembler/comp_arch/labs/tools.
2. Запустил программу DOSBox, смонтировал виртуальный диск C: в каталоге MASM при помощи mount C ~/Assembler/comp_arch/labs/tools.
3. Просмотрел программу в текстовом редакторе, изучил структуру и реализации каждого сегмента программы. Преобразовал строку-приветствие.
4. В DOS перешёл на виртуальный диск при помощи команды C:
5. Выполнил трансляцию программы с помощью ввода команды:
 > MASM HELLO1.ASM
В результате трансляции был создан объектный файл HELLO1.OBJ. Трансляция прошла без ошибок и предупреждений.
6. Слинковал загрузочный модуль HELLO1.exe с помощью строки:
 > LINK HELLO1.OBJ
В результате работы линковщика создавался загрузочный модуль HELLO1.exe.
7. Загрузил русскую кодовую таблицу символов путём набора строки:
 > keyb ru 866
8. Запустил программу в автоматическом режиме путём набора строки:
 > HELLO1.EXE

9. Вывод программы:

> Вас приветствует ст.гр. 9382 – Герасев Георгий.

10. Используя отладчик, выполнил запуск программы HELLO1.exe. В ходе выполнения программы записывались используемые регистры и ячейки памяти до и после выполнения команд. Отладчик был запущен при помощи команды:

> AFDPRO HELLO1.EXE.

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.

Содержимое сегментных регистров до старта программы: CS:1A05, DS:19F5, ES:19F5, SS:1A0A, HS:19F5, FS:19F5

Адрес Команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			до выполнения	После выполнения
0010	MOV AX, 1A07	B8071A	(AX) = 0000 (IP) = 0010	(AX) = 1A07 (IP) = 0013
0013	MOV DS, AX	8ED8	(AX) = 1A07 (DS) = 19F5 (IP) = 0013	(AX) = 1A07 (DS) = 1A07 (IP) = 0015
0015	MOV DX, 0000	BA0000	(DX) = 0000 (IP) = 0015	(DX) = 0000 (IP) = 0018
0018	MOV AH,09	B409	(AX) = 1A07 (IP) = 0018	(AX) = 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	(AX) = 4C07 (DS) = 1A07 (IP) = 001E	(AX) = 0000 (DS) = 19F5 (IP) = 0010

Часть 2.

1. Просмотрел программу HELLO2.ASM в текстовом редакторе, изучил структуру и реализации каждого сегмента программы. Преобразовал строку-приветствие.

2. Выполнил трансляцию программы HELLO2.ASM с помощью транслятора MASM и команды:

>MASM HELLO2.ASM

В результате трансляции был создан объектный файл HELLO2.OBJ. Трансляция прошла без ошибок, но с одним предупреждением.

3. Используя линковщик LINK, создал загрузочный модуль HELLO2.EXE. Используемая команда:

>LINK HELLO2.EXE

4. Запустил программу HELLO2.EXE и проверил корректность ее работы:

> Привет!

> Студент Герасев Георгий из группы 9382.

5. Запустил программу с помощью отладчика AFDPRO.EXE и пошагово исполнил с записью используемых регистров и ячеек команд.

Содержимое сегментных регистров до старта программы: CS:1A0B, DS:19F5, ES:19F5, SS:1A05, HS:19F5, FS:19F5

Табл.2.

Адрес Команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0005	PUSH DS	1E	(DS) = 19F5 (SP) = 0018 (IP) = 0005 Stack +0 0000 +2 0000 +4 0000	(DS) = 19F5 (SP)=0016 (IP) = 0006 Stack +0 19F5 +2 0000 +4 0000
0006	SUB AX, AX	2BC0	(AX) = 0000 (IP) = 0006	(AX) = 0000 (IP) = 0008
0008	PUSH AX	50	(AX) = 0000 (SP)=0016 (IP) = 0008 Stack +0 19F5 +2 0000 +4 0000	(AX) = 0000 (SP) = 0014 (IP) = 0009 Stack +0 0000 +2 19F5 +4 0000
0009	MOV AX, 1A07	B8071A	(AX) = 0000 (IP) = 0009	(AX) = 1A07 (IP) = 000C
000C	MOV DS, AX	8ED8	(AX) = 1A07 (DS) = 19F5 (IP) = 000C	(AX) = 1A07 (DS) = 1A07 (IP) = 000E

000E	MOV DX, 0000	BA0000	(DX) = 0000 (IP) = 000E	(DX) = 0000 (IP) = 0011
0011	CALL 0000	E8ECFF	(SP) = 0014 (IP) = 0011 Stack +0 0000 +2 19F5 +4 0000	(SP) = 0012 (IP) = 0000 Stack +0 0014 +2 0000 +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 Stack +0 0014 +2 0000 +4 19F5	(SP) = 0014 (IP) = 0014 Stack +0 0000 +2 119C +4 0000
0014	MOV DX,000A	BA0A00	(DX) = 0000 (IP) = 0014	(DX) = 000A (IP) = 0017
0017	CALL 0000	E8E6FF	(SP) = 0014 (IP) = 0017 Stack +0 0000 +2 19F5 +4 0000	(SP) = 0012 (IP) = 0000 Stack +0 001A +2 0000 +4 119C
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 Stack +0 001A +2 0000 +4 19F5	(SP) = 0014 (IP) = 001A Stack +0 0000 +2 19F5 +4 0000
001A	RET Far	CB	(SP) = 0014 (IP) = 001A (CS) = 1A0B Stack +0 0000 +2 19F5 +4 0000	(SP) = 0018 (IP) = 0000 (CS) = 19F5 Stack +0 0000 +2 0000 +4 0000
0000	INT 20	CD20	(AX) = 0907 (DS) = 1A07 (IP) = 0000	(AX) = 0000 (DS) = 19F5 (IP) = 0005

Выводы.

В результате выполнения лабораторной работы был освоен этап трансляции, а также выполнение и пошаговое исполнение программы при помощи отладчика на языке Ассемблера.

Приложение А

Текст файла 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.9382 Герасев Георгий',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
```

Текст файла HELLO1.LST

Page 1-1

```
1
2 ; HELLO1.ASM -  ã`à®é¥ ï ç¥àášï ãç¥
i
®© `à®£à ¬¬ë « ï.à ï. N1
3 ; ® €šáæš`«š¥
"▯àâšâ¥ªâ
```

```

                                ãà  ª®¬îîâ¥à "
4                                ;
*****
                                *****
5                                ;  º § ¸¥š¥:  ºà®£à ¬¬  ä®à¬šàã¥â š
çëç
                                ®€šâ  íªà  ¬àšç¥âââçš¥
6                                ;                                ®«ì§®ç  â¥«î  á
¬®¬®éîî äã
                                ªæšš  ººº  "ºëç®€  áâà®ªš"
7                                ;                                (®¬¥à 09  ¬à¥àëç  š¥
21h)
                                ,  ª®â®à î:
8                                ;                                -  ®î¥à¬¥çšç  ¥â  çëç®€
                                íªà  áâà®ªš  áš¬ç®«®ç,
9                                ;                                § ª  çšç  íé¥©áî §-
ª®¬
                                "$";
10                               ;                                -  âà¥îã¥â §  €  šî  ç
à¥£š
                                áâà¥  ah  ®¬¥à  äãªæšš=09h,
11                               ;                                ç  à¥£šáâà¥  dx  -
á¬¥
                                é¥šî  €à¥á  çëç®€š¬®©
12                               ;                                áâà®ªš;
13                               ;                                -  šá¬®«ì§ã¥â  à¥£šáâà
ax
                                š  ¥  á®âà  î¥â  ¥£®
14                               ;                                á®€¥àšš¬®¥.
15                               ;
*****
                                *****
16
17                               DOSSEG
                                ;  º  €  š¥  á¥£¬¥â®ç  ¬®€  ººº
18                               .MODEL  SMALL
                                ;  º®€¥«ì  ¬  ¬îâš-
SMALL(º « î)
19                               .STACK  100h
                                ;  ºâç¥áâš  ¬®€  ºâ¥ª  256  î  ©â
20                               .DATA
                                ;  º  ç  «®  á¥£¬¥â  €  ëâ
21 0000                               Greeting LABEL BYTE
                                ;  º¥ªâ  ¬àšç¥âââçšî

```



```

22 0000 82 A0 E1 20 AF E0      DB '  á ¯àšç¥âââçã¥â
ââ.£à.7303 -  ç

                                ®ç  . . . ' , 13 , 10 , '$ '
23      A8 A2 A5 E2 E1 E2
24      A2 E3 A5 E2 20 E1
25      E2 2E A3 E0 2E 37
26      33 30 33 20 2D 20
27      88 A2 A0 AD AE A2
28      20 88 2E 88 2E 0D
29      0A 24
30      .CODE
                                ;  ç «® á¥£¬¥â  ª®€
31 0000 B8 ---- R      mov ax, @data
                                ;  £àã§ª ç DS €à¥á  ç «
32 0003 8E D8      mov ds, ax
                                ;  á¥£¬¥â € ăă
33 0005 BA 0000 R      mov dx, OFFSET Greeting

```


Symbols-1

Segments and Groups:

Combine Class	N a m e	Length	Align
DGROUP	GROUP		
_DATA	002C 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	HELL01	
@VERSION	TEXT	510	

```
33 Source  Lines
33 Total   Lines
19 Symbols
```

47460 + 461847 Bytes symbol space free

```
0 Warning Errors
0 Severe  Errors
```

Текст файла HELLO2.ASM

EOFLine EQU '\$'

AStack SEGMENT STACK

DW 12 DUP(?)

AStack ENDS

DATA SEGMENT

HELLO DB 'Привет!', 0AH, 0DH, EOFLine

GREETING DB 'Студент Герасев Георгий из группы 9382 - \$'

DATA ENDS

CODE SEGMENT

ASSUME CS:Code DS:DATA SS:AStack

WriteMsg PROC NEAR

mov AH, 9

int 21h

ret

WriteMsg ENDP

Main PROC FAR

push DS

sub AX, AX

push AX

mov AX, DATA

mov DS, AX

mov DX, OFFSET HELLO

```

        call WriteMsg
        mov  DX, OFFSET GREETING
        call WriteMsg
        ret

Main     ENDP
CODE     ENDS

END Main

```

Текст файла HELLO2.LST

Page 1-1

```

; HELLO2 -  çŸ; i ¯à®£à ¬¬ N2 « ;.à ;.#1 ¯®
€šáæš ¯«šŸ " àåšâŸªââà ª®¬ ¯îâŸà "
;      à®£à ¬¬ šá ¯®«ìšâŸâ ¯à®æŸ£âââ €«i ¯
Ÿç âš áâà®ªš
;
;

= 0024      EOFLine EQU '$'      ; ¯àŸ€Ÿ«ŸšŸ áš¬¢®«ì
®© ª®ââ âë
;      " ®Ÿæ áââ®ªš"

;  âŸª ¯à®£à ¬¬ë

0000      AStack  SEGMENT STACK
0000 000C[      DW 12 DUP(?) ;  â¢®€šââî 12 á«®¢ ¯
¬îâš
      ????
      ]

0018      AStack  ENDS

;  ëŸ ¯à®£à ¬¬ë

0000      DATA   SEGMENT

```

```

;  šàŸªš¢ë ® ¯šá šĩ € ěă

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

;  ®€ ¯à®£à ¯¬ě

0000                                CODE   SEGMENT
                                ASSUME CS:Code DS:DATA SS:AStack
HELLO2.ASM(28): warning A4001: Extra characters on line
;  à®æŸ€ãà ¯Ÿç âš áâà®ªš

0000                                WriteMsg PROC NEAR
0000 B4 09                                mov  AH,9
0002 CD 21                                int  21h ;  ě§®¢ äãªšš DOS ¯® ¯àŸ
                                àě¢ šĩ

0004 C3                                ret
0005                                WriteMsg ENDP

;  ®«®¢ ĩ ¯à®æŸ€ãà

0005                                Main   PROC FAR
0005 1E                                push DS    ;\  ®åà ŸšŸ €àŸá
                                ç «  PSP ¢ áâŸªŸ
0006 2B C0                                sub  AX,AX  ; > €«ĩ ¯®á«Ÿ€ãîéŸ£® ¢
                                ®ááâ ®¢«Ÿšĩ ¯®

```

```

0008 50                push AX      ;/ ^@¬ €¥ ret, § ¢¥à
                        è íé¥© ¯à®æ¥€ãää.
0009 B8 ---- R        mov AX,DATA      ; £ää§ª
                        á¥£¬¥â®£®
000C 8E D8            mov DS,AX        ; à¥£šáää
                        € äå.
000E BA 0000 R        mov DX, OFFSET HELLO ; ë¢®€
                        íªà ¯¥à¢®©
0011 E8 0000 R        call WriteMsg     ; áâà®ªš ¯à
                        š¢¥âáâ¢šì.
0014 BA 0010 R        mov DX, OFFSET GREETING ; ë¢®€
                        íªà ¢â®à®©
0017 E8 0000 R        call WriteMsg     ; áâà®ªš ¯à
                        š¢¥âáâ¢šì.
001A CB              ret                ; äå®€ ¢ D
                        OS ¯®ª®¬ €¥,
                                ; å®€íé¥©
                        áí ¢ 1-®¬ á«®¢¥ PSP.
001B                Main ENDP
001B                CODE ENDS
                        END Main

```

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	

51 Source Lines

51 Total Lines

13 Symbols

48002 + 461305 Bytes symbol space free

1 Warning Errors

0 Severe Errors