

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.

Студентка гр. 9382

Пя С.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

Изучить строение программ на языке Ассемблера, научиться следить за изменениями в ходе их выполнения, познакомиться с функционалом языка.

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. Непонятные фрагменты прояснить у преподавателя. 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. 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.

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.

Выполнение работы.

Часть 1.

1. Открыв в текстовом редакторе программу hello1.asm, изучила ее строение, обратив особое внимание на команды вывода.
2. Пошагово разобралась в структуре кода и преобразовала строку-приветствие в соответствии со своими личными данными.

3. Загрузила файл hello1.asm в каталог Masm.

4. Протранслировала программу с созданием объектного файла и файла диагностических сообщений. Ошибок, как и предупреждений, не было обнаружено.

5. Скомпилировала загрузочный модуль с созданием карты памяти и исполняемого файла hello1.exe.

6. Выполнила программу, предварительно введя строку для правильного отображения русских слов, программа сработала корректно, вывод программы:

> Вас приветствует ст.гр. 9382 – Пя С.

7. Запустила выполнение программы под управлением отладчика.

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	(DS) = 19F5 (IP) = 0013	(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. Открыла программу hello2.asm с помощью текстового редактора, акцентируя внимание на процедуру WriteMsg.

2. Изучила строение программы hello2.asm, параллельно преобразовав строку-приветствие в соответствие с личными данными.

3. Загрузила hello2.asm в каталог Masm.

4. Протранслировала программу с созданием объектного файла и файла с диагностическими сообщениями. Синтаксических ошибок не обнаружено.
5. Скомпилировала загрузочный модуль с созданием карты памяти и исполняемого файла hello2.exe.
6. Выполнила программу hello2.exe и проверила корректность ее работы:
 > Hello Worlds!
 > Student from 9382 – Pya Son Khwa
7. Запустила программу с помощью отладчика afdpro.exe и пошагово исполнила 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,0010	BA1000	(DX) = 0000 (IP) = 0014	(DX) = 0010 (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 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 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 Stack +0 0000 +2 19F5 +4 0000	(SP) = 0018 (IP) = 0000 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 ходе выполнения лабораторной работы было совершенно ознакомление с языком Ассемблера, был освоен этап трансляции, также была изучена структура данных программ и работа с программой DOSBox 0.74-3.

Приложение.

Текст файла HELLO1.ASM

__Microsoft (R) Macro Assembler Version 5.10

10/13/20 20:40:5

Page 1-1

```
; HELLO1.ASM - гїа®йГ п ўґабЁп гзґў® ĩа®Ја
;   -л « ў.а ў. N1
;   ;   ĩ® жЁбжЁї«ЁГ "баеЁвГёвга €®-ї
MOVГa "
;   ;   *****
;   ;   *****
;   ;   зГЁГ: ца®Ја - д®а-ёагГв Ё ўлў®жЁв
нёа ĩаЁўГвбвўЁГ
;   ;   ĩ®«м§®ў вГ«п б ĩ®-®ймо дгёжЁЁ „Һ
‘ „лў®ж бва®ёЁ"
;   ;   (®-ґа 09 ĩаґалў ЁГ 21h), €®в®а
п:
;   ;   - ®ўГбїґзЁў Гв ўлў®ж нёа бв
а®ёЁ бЁ-ў®«®ў,
;   ;   § € зЁў ойГ®бп § €®- "$";
;   ;   - ваГўГГв § ж Ёп ў аГЈЁбваГ аҺ
®-ґа дгёжЁЁ=09h,
;   ;   ў аГЈЁбваГ dx - б-ґйГЁп
жаГб ўлў®жЁ-®®
;   ;   бва®ёЁ;
;   ;   - Ёбї®«м§гГв аГЈЁбва ах Ё Г
б®еа пГв ГЈ®
;   ;   б®жаґа!Ё-®Г.
;   ;   *****
;   ;   *****

DOSSEG
;   ;   ‡ ж ЁГ бГЈ-Гв®ў ĩ®ж „Һ‘
.MODEL SMALL
;   ;   Њ®жГ«м ĩ -пвЁ-SMALL(Њ « п)
.STACK 100h
;   ;   ЊвўГбвЁ ĩ®ж ‘вГ€ 256 ў ®в
.DATA
;   ;   ;   з «® бГЈ-Гв ж ле
0000 Greeting LABEL BYTE
;   ;   ;   Гёбв ĩаЁўГвбвўЁп
0000 DB ' , 6 ĩаЁўГвбвўГГв бв.Ја.7303 - €ў ®ў €.€
;   ;   ;   .',13,10','$'

A8 A2 A5 E2 E1 E2
A2 E3 A5 E2 20 E1
E2 2E A3 E0 2E 37
33 30 33 20 2D 20
88 A2 A0 AD AE A2
20 88 2E 88 2E 0D
0A 24

.CODE
;   ;   ;   з «
® бГЈ-Гв €®ж
0000 B8 ---- R mov ax, @data ;   ;   ‡ Ја
г§€ ў DS жаГб з «
0003 8E D8 mov ds, ax ;   ;   бГЈ-
Гв ж ле
0005 BA 0000 R mov dx, OFFSET Greeting ;   ;   ‡ Ја
г§€ ў dx б-ґйГЁп
```

__Microsoft (R) Macro Assembler Version 5.10

10/13/20 20:40:5

```

                                ; ҝаГб
                                вГЄбв İаЁЎГвбвЎЁп
0008      DisplayGreeting:
0008  B4 09      mov  ah, 9                                ; # дГ
                                ЄжЁЁ „Һ‘ İГз вЁ бва®ЄЁ
000A  CD 21      int  21h                                ; ўлў®ҝ
                                нЄа İаЁЎГвбвЎЁп
000C  B4 4C      mov  ah, 4ch                            ; # дГ
                                ЄжЁЁ „Һ‘ § ўГаиГЁп İа®Ја ¬¬л
000E  CD 21      int  21h                                ; § ўГа
                                иГЁГ İа®Ја ¬¬л Ё ўле®ҝ ў „Һ‘
                                END
__Microsoft (R) Macro Assembler Version 5.10             10/13/20 20:40:5
                                                         Symbols-1

```

Segments and Groups:

N a m e	Length	Align	Combine	Class
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	HELLO1	
@VERSION	TEXT	510	

33 Source Lines
 33 Total Lines
 19 Symbols

47994 + 461313 Bytes symbol space free

0 Warning Errors
 0 Severe Errors

Текст файла HELLO1.ASM

```

; HELLO1.ASM - гİа®ЙГ п ўГабЁп гзГЎ®® İа®Ја ¬¬л « ў.а ў. N1
;                İ® ҝЁбжЁİ«ЁГ "ҝаеЁвГЄвҝа Є®¬İмовГҝа "
;
*****
; К § зГЁГ: Ца®Ја ¬¬ д®а¬ЁагГв Ё ўлў®ҝЁв нЄа İаЁЎГвбвЎЁГ
;                İ®«мS®ў вГ«п б İ®¬®Ймо дГЄжЁЁ „Һ‘ ",лў®ҝ бва®ЄЁ"

```


[illegible]

```

; HELLO2 - "зГЎ п İa®Ja ¬¬ N2 « Ÿ.a Ÿ.#1 İ®
æĖ6жĖİ«ĖĖ "baeĖвГЄвга €®¬İмовГa "
;      Ça®Ja ¬¬ Ė6İ®«мҫгГв İa®жГҫгаг æ«п İ
Гз вĖ бва®ЄĖ
;
;      '...л‘‘ ÇĥĥГĥГĥĥĥ»

= 0024      EOFLine EQU '$'      ; ĥİаГҫГ«ГĖГ бĖ¬Ÿ®«м
®© €®бв вл
;      "л®Гж бва®ЄĖ"

;      ‘вГЄ İa®Ja ¬¬л

0000      AStack      SEGMENT      STACK

```

```

0000 000C[                                DW 12 DUP(?)    ; Ңвў®ҔЁвбп 12 б«ў ї
                                -пвЁ
                                ????
                                ]

0018                                AStack    ENDS

                                ; „ лҒ їа®Ја -л

0000                                DATA      SEGMENT

                                ; „ЁаҒёвёўл ®їёб ёп Ҕ ле

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 9382 - SonKhwa$'
    74 20 66 72 6F 6D
    20 39 33 38 32 20
    2D 20 53 6F 6E 4B
    68 77 61 24
002C                                DATA      ENDS

                                ; Љ®Ҕ їа®Ја -л

0000                                CODE        SEGMENT
                                ASSUME CS:Code DS:DATA SS:AStack
HELLO2.ASM(28): warning A4001: Extra characters on line
                                ; Ҙа®җҒҔҒа їҒз вЁ бва®ёё
0000                                WriteMsg   PROC    NEAR
0000 B4 09                                mov     AH,9
0002 CD 21                                int     21h ; ,л§®ў дҒёҗёё DOS ї® їаҒ
                                алў ёо
0004 C3                                ret
0005                                WriteMsg   ENDP

                                ; Ғ®«®ў п їа®җҒҔҒа
0005                                Main        PROC    FAR
0005 1E                                push    DS ;\ ‘®ea ҒЁҒ ҔаҒб
                                з « PSP ў бвҒёҒ
0006 2B C0                                sub     AX,AX ; > Ҕ«п ї®б«ҒҔҒойҒҗ® ў
__Microsoft (R) Macro Assembler Version 5.10
                                10/22/20 13:08:5
                                Page      1-2

```

```

                                ®ббв ®ў«Ғёп ї®
0008 50                                push    AX ;/ €®- ҔҒ ret, § ўҒа
                                и ойҒ® їа®җҒҔҒаҒ.
0009 B8 ---- R                        mov     AX,DATA ; ‡ ЈаҒ§ё
                                бҒҗ-Ғв®Ј®
000C 8E D8                        mov     DS,AX ; аҒҗёбва
                                Ҕ ле.
000E BA 0000 R                        mov     DX, OFFSET HELLO ; ,лў®Ҕ
                                нёа їҒаў®®
0011 E8 0000 R                        call    WriteMsg ; бва®ёё їа
                                ЁўҒвбвўёп.
0014 BA 0010 R                        mov     DX, OFFSET GREETING ; ,лў®Ҕ
                                нёа ўв®а®®
0017 E8 0000 R                        call    WriteMsg ; бва®ёё їа
                                ЁўҒвбвўёп.
001A CB                                ret ; ,ле®Ҕ ў D
                                OS ї® €®- ҔҒ,
                                ; е®ҔпйҒ®
                                бп ў 1-®- б«®ўҒ PSP.
001B                                Main        ENDP

```

001B CODE ENDS
END Main

__Microsoft (R) Macro Assembler Version 5.10

10/22/20 13:08:5
Symbols-1

Segments and Groups:

N a m e	Length	Align	Combine	Class
ASTACK	0018	PARA		STACK
CODE	001B	PARA		NONE
DATA	002C	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

Текст файла HELLO2.ASM

```
; HELLO2 - "зГЎ п ĩa®Ja ¬¬ N2 « Ÿ.a Ÿ.#1 ĩ® ¤ĖĖжĖĭ«ĖГ
"БаеĖвГĖвга Ė®¬ĭмовГа "

;          ĩa®Ja ¬¬ ĖĖĭ®«м$гГв ĩa®жГ¤гaг ¤«п ĭГз вĖ бва®ĖĖ
;
;      '...Љ''  ĩĥĥĥĥĥĥĥ>

EOFLine EQU '$'          ; ĥĭaГ¤Г«ГĖГ бĖ¬Ÿ®«м®© Ė®бв вл
                        ;      "Љ®Гж бва®ĖĖ"

;  'вГĖ ĩa®Ja ¬¬л

AStack    SEGMENT  STACK
            DW 12 DUP(?)    ; ĥвŸ®¤Ėвбп 12 б«®Ÿ ĭ ¬пвĖ
AStack    ENDS

;  „ лГ ĩa®Ja ¬¬л

DATA      SEGMENT

;  „ĖaГĖвĖŸл ®ĭĖĖ Ėп ¤ лe

HELLO      DB 'Hello Worlds!', 0AH, 0DH, EOFLine
GREETING   DB 'Student from 9382 - Pya Son Khwa$'
DATA      ENDS

;  Љ®¤ ĩa®Ja ¬¬л

CODE      SEGMENT

            ASSUME CS:Code DS:DATA SS:AStack

;  ĩa®жГ¤га ĭГз вĖ бва®ĖĖ
```

```

WriteMsg PROC NEAR

    mov     AH,9

    int     21h    ; ,л$®Ÿ дг€жЁЁ DOS ĩ® ĩаГалŸ Ёо

    ret

WriteMsg ENDP

; ĩ®«®Ÿ п ĩа®жГга

Main PROC FAR

    push    DS      ;\  '®ea ГЁГ маГб з « PSP Ÿ бвГГГ
    sub     AX,AX    ; > «п ĩ®б«ГгагойГЈ® Ÿ®ббв ®Ÿ«ГЁп ĩ®
    push    AX      ;/ €®¬ «Г ret, $ ŸГгаи ойГ© ĩа®жГгаг.
    mov     AX,DATA      ; ‡ Јаг$€ бГЈ¬Гв®Ј®
    mov     DS,AX      ; аГЈЁбва « ле.
    mov     DX, OFFSET HELLO ; ,лŸ®« нСа ĩГай®©
    call    WriteMsg   ; бва®€Ё ĩаЁŸГвбвŸЁп.
    mov     DX, OFFSET GREETING ; ,лŸ®« нСа Ÿв®а®©
    call    WriteMsg   ; бва®€Ё ĩаЁŸГвбвŸЁп.
    ret              ; ,ле®« Ÿ DOS ĩ® €®¬ «Г,
                    ;     е®«пйГ©бп Ÿ 1-®¬

б«®ŸГ PSP.

Main ENDP

CODE ENDS

END Main

```