

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. 1383

Седова Э.А.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Получение навыков трансляции, выполнения и отладки программ на языке Ассемблера.

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.

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.

Отчет по работе должен содержать:

- 1) текст задания;
- 2) тексты исходных файлов программ hello1 и hello2;
- 3) тексты файлов диагностических сообщений hello1.lst и hello2.lst;
- 4) протокол работы на компьютере, включающий основные действия по пунктам 1 - 6 и протоколы пошагового исполнения каждой из программ под

управлением отладчика в виде таблицы 1 (черновики протоколов должны быть подписаны преподавателем).

5) выводы по работе.

Выполнение работы.

Начальное содержимое сегментных регистров (CS) = 1A05, (DS) = 19F5, (ES) = 19F5, (SS) = 1A0C.

Таблица 1- исполнение файла hello1.asm

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0010	Mov ax, @data	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	BA0000	(IP) = 0015	(IP) = 0018
0018	Mov ah, 9	B409	(AX) = 1A07 (IP) = 0018	(AX) = 0907 (IP) = 001A
001A	int 21h	CD21	(IP) = 001A	(IP) = 001C
001C	Mov ah, 4ch	B44C	(AX) = 0907 (IP) = 001C	(AX) = 4C07 (IP) = 001E
001E	int 21h	CD21	(AX) = 4C07 (CX) = 0067 (DS) = 1A07 (IP) = 001E	(AX) = 0000 (CX) = 0000 (DS) = 19F5 (IP) = 0010

Начальное содержимое сегментных регистров (CS) = 1A0A, (DS) = 19F5, (ES) = 19F5, (SS) = 1A05.

Таблица 2- исполнение файла hello2.asm

Адрес команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти	
			До выполнения	После выполнения
0005	push DS	1E	Stack: +0 0000 +2 0000 +4 0000 +6 0000 (SP) = 0018 (IP) = 0005	Stack: +0 19F5 +2 0000 +4 0000 +6 0000 (SP) = 0016 (IP) = 0006
0006	sub AX,AX	2BC0	(IP) = 0006	(IP) = 0008
0008	push AX	50	Stack: +0 19F5 +2 0000 +4 0000 +6 0000 (IP) = 0008 (SP) = 0016	Stack: +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 0009 (SP) = 0014
0009	mov AX,DATA	B8071A	Stack: (IP) = 0009 (AX) = 0000	Stack: (IP) = 000C (AX) = 1A07
000C	mov DS,AX	8ED8	Stack: (IP) = 000C (DS) = 19F5	Stack: (IP) = 000E (DS) = 1A07

000E	mov DX, OFFSET HELLO	BA0000	(IP) = 000E	(IP) = 0011
0011	call WriteMsg	E8ECFF	Stack: +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 0011 (SP) = 0014	Stack: +0 0014 +2 0000 +4 19F5 +6 0000 (IP) = 0000 (SP) = 0012
0000	mov AH,9	B409	(IP) = 0000 (AX) = 1A07	(IP) = 0002 (AX) = 0907
0002	int 21h	CD21	(IP) = 0002	(IP) = 0004
0004	ret	C3	Stack: +0 0014 +2 0000 +4 19F5 +6 0000 (IP) = 0004 (SP) = 0012	Stack: +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 0014 (SP) = 0014
0014	mov DX, OFFSET GREETING	BA1000	(IP) = 0014 (DX) = 0000	(IP) = 0017 (DX) = 0010
0017	call WriteMsg	E8E6FF	Stack: +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 0017 (SP) = 0014	Stack: +0 001A +2 0000 +4 19F5 +6 0000 (IP) = 0000 (SP) = 0012
0000	mov AH,9	B409	(IP) = 0000	(IP) = 0002

0002	int 21h	CD21	(IP) = 0001	(IP) = 0004
0004	ret	C3	Stack: +0 001A +2 0000 +4 19F5 +6 0000 (IP) = 0004 (SP) = 0012	Stack: +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 001A (SP) = 0014
001A	ret	CB	Stack: +0 0000 +2 19F5 +4 0000 +6 0000 (IP) = 001A (SP) = 0014 (CS) = 1A0A	Stack: +0 0000 +2 0000 +4 0000 +6 0000 (IP) = 0000 (SP) = 0018 (CS) = 19F5
0000	Int 20	CD20	(AX) = 0907 (DX) = 0010 (IP) = 0000 (CS) = 19F5 (DS) = 1A07	(AX) = 0000 (DX) = 0000 (IP) = 0005 (CS) = 1A0A (DS) = 19F5

Программный код см. в приложении А.

Файлы диагностических сообщений см. в приложении Б.

Выводы.

В ходе выполнения работы был получен навыки трансляции отладки и запуска программ на языке Ассемблера.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: hello1.asm

```
DOSSEG ; Задание сегмента под DOS
.MODEL SMALL ; Модель памяти - SMALL
.STACK 100h ; Отвести под Стек 256 байт
.DATA ; начало сегмента данных
Greeting LABEL BYTE ; Текст приветствия
DB Вас приветствует ст.гр.1383 - Седова Э.А.',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 ; # функция DOS печати строки
int 21h ; вывод на экран приветствия
mov ah, 4ch ; # функции DOS завершения программы
int 21h ; завершение программы и выход в DOS
END
```

Название файла: hello2.asm

```
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 1383 - Sedova E.A.$'
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          ; \ Созрание 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 А

ФАЙЛЫ ДИАГНОСТИЧЕСКИХ СООБЩЕНИЙ

Название файла: hello1.lst

Microsoft (R) Macro Assembler Version 5.10
17:18:22

10/3/22

Page 1-1

```

; HELLO1.ASM - СíPíCЪPsC%PμPSPSP°CЦ PIPμCЪCÍPё
CЦ CíC±PμP±PSPsPN° PíCЪPsPíCЪP°PjPjC< P»P°P±.CЪP
°P±. N1
;
PíPs PrPёCÍC±PёPíP»PёPSPμ "PђCЪ
C...PёC, PμPеC, CíCЪP° PеPsPjPíCЪCЪC, PμCЪP°"
; *****
*****
; PќP°P·PSP°C±PμPSPёPμ: PμCЪPsPíCЪP°PjPjP° C„Ps
CЪPjPёCЪCíPμC, Pё PIC<PIPsPrPёC, PSP° CќPеCЪP°P
S PíCЪPёPIPμC, CÍC, PIPёPμ
;
PíPsP»CЪP·PsPIP°C, PμP»CЦ CÍ PíPsP
jPsC%CЪCЪ C„CíPSPеC±PёPё P"PhPŸ "P'C<PIPsPr CÍC
,CЪPsPеPё"
;
(PSPsPjPμCЪ 09 PíCЪPμCЪC<PIP°PSPё
Pμ 21h), PеPsC, PsCЪP°CЦ:
;
- PsP±PμCÍPíPμC±PёPIP°PμC, PIC<P
IPsPr PSP° CќPеCЪP°PS CÍC,CЪPsPеPё CÍPёPjPIPsP»
PsPI,
;
P·P°PеP°PSC±PёPIP°CЪC%PμPN°CÍCЦ
P·PSP°PеPsPj "$";
;
- C,CЪPμP±CíPμC, P·P°PrP°PSPёCЦ
PI CЪPμPíPёCÍC,CЪPμ ah PSPsPjPμCЪP° C„CíPSPеC±P
ёPё=09h,

```

```
; P° PI CЪPµPiPëCfC,CЪPµ dx - C
fPjPµC%PµPSPëCµ P°PrCЪPµCfP° PIC<PIPsPrPëPjPs
PN°
; CfC,CЪPsPePë;
; - PëCfPiPsP»CЪP·CfPµC, CЪPµPiPëC
fC,CЪ ax Pë PSPµ CfPsC...CЪP°PSCµPµC, PµPiPs
; CfPsPrPµCЪPµPëPjPsPµ.
; *****
*****
```

```
DOSSEG
; P-P°PrP°PSPëPµ CfPµPiPjPµPSC,PsPI PiPsPr P"Ph
PŸ
```

```
.MODEL SMALL
; PЪPsPrPµP»CЪ PiP°PjCµC,Pë-SMALL(PЪP°P»P°Cµ)
.STACK 100h
; PhC,PIPµCfC,Pë PiPsPr PŸC,PµPe 256 P±P°PN°C,
.DATA
; PkP°C±P°P»Ps CfPµPiPjPµPSC,P° PrP°PSPSC<C...
Greeting LABEL BYTE
; PŸPµPeCfC, PiCЪPëPIPµC,CfC,PIPëCµ
```

```
0000
0000 D0 92 D0 B0 D1 81 DB 'P'P°Cf PiCЪPëPIPµC,CfC,PICfPµC,
CfC,.PiC
```

```
Б.1383 - PŸPµPrPsPIP° P.Pђ.',13,10,'$'
```

```
20 D0 BF D1 80 D0
B8 D0 B2 D0 B5 D1
82 D1 81 D1 82 D0
B2 D1 83 D0 B5 D1
82 20 D1 81 D1 82
2E D0 B3 D1 80 2E
31 33 38 33 20 2D
```

```
_Microsoft (R) Macro Assembler Version 5.10
17:18:22
```

10/3/22

Page 1-2

```
20 D0 A1 D0 B5 D0
B4 D0 BE D0 B2 D0
B0 20 D0 AD 2E D0
90 2E 0D 0A 24
```

```
.CODE ; PkP°C
+P°P»Ps CfPµPiPjPµPSC,P° PePsPrP°
0000 B8 ---- R mov ax, @data ; P-P°
PiCЪCfP·PeP° PI DS P°PrCЪPµCfP° PSP°C±P°P»P°
0003 8E D8 mov ds, ax ;
CfPµ
PiPjPµPSC,P° PrP°PSPSC<C...
0005 BA 0000 R mov dx, OFFSET Greeting ; P-P°
PiCЪCfP·PeP° PI dx CfPjPµC%PµPSPëCµ
; P°PrC
ЪPµCfP° C,PµPeCfC,P° PiCЪPëPIPµC,CfC,PIPëCµ
0008 DisplayGreeting:
0008 B4 09 mov ah, 9 ; #
C,,
CfPSPeC±PëPë P"PhPŸ PiPµC±P°C,Pë CfC,CЪPsPePë
000A CD 21 int 21h ;
PIC<P
```

```

        IPSPr PSP° CKPeCBP°PS PiCBPëPIPµC,CfC,PIPëCµ
000C B4 4C          mov ah, 4ch ; #
C,,
        CfpSPeCfPëPë P"PñPŸ P·P°PIPµCBCEPµPSPëCµ PiCBPs
        PiCBP°PjPjC<
000E CD 21          int 21h ;
P·P°P
        IPµCBCEPµPSPëPµ PiCBPsPiCBP°PjPjC< Pë PIC<C...PsP
        r PI P"PñPŸ
        END

```

Microsoft (R) Macro Assembler Version 5.10
17:18:22

10/3/22

Symbols-1

Segments and Groups:

	N a m e	Length	Align	Combine	Class
DGROUP	GROUP			
_DATA	0047	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

47992 + 459268 Bytes symbol space free

0 Warning Errors
0 Severe Errors

Название файла: hello2.lst

Microsoft (R) Macro Assembler Version 5.10
02:52:53

10/9/22

Page 1-1

```

; HELLO2 - PJC+PµP+PSP°Cµ PiCBPsPiCBP°PjPjP° N2
P»P°P±.CBP°P±.#1 PiPs PrPëCfCfPëPiP»PëPSPµ "P

```

```

        ħCĤC...PĕC, PμPeC, CíCĤP° PePsPjPíCĤCĤC, PμCĤP°"
        ;          PμCĤPsPíCĤP°PjPjP° PĕCíPíPsP»CĤP·CíP
        μC, PíCĤPsC†PμPrCíCĤCí PrP»Cĭ PíPμC†P°C, Pĕ CíC,
        CĤPsPePĕ
        ;
        ;          PŸP•PĭPŸPŸ PμP PĥP"P PĥPĥPĥP«

= 0024          EOFLine EQU '$'          ;
PhPíCĤPμPrPμP»PμPSP
        ĕPμ CíPĕPjPIPsP»CĤPSPsPNº PePsPSCíC, P°PSC, C<
        ;          "PĭPsPSPμC† CíC
        ,CĤPsPePĕ"

        ; PŸC, PμPe PíCĤPsPíCĤP°PjPjC<

ASSUME CS:CODE, SS:AStack

0000          AStack      SEGMENT      STACK
0000 000C[          DW 12 DUP('!')          ;
PhC, PIPsPrPĕC, CíC          ĭ 12 CíP»PsPI PíP°PjCĭC, Pĕ
        0021          ]

0018          AStack      ENDS

        ; P"P°PSPSC<Pμ PíCĤPsPíCĤP°PjPjC<

0000          DATA          SEGMENT

        ; P"PĕCĤPμPeC, PĕPIC< PsPíPĕCíP°PSPĕCĭ PrP°PSPS
        C<C...

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 1383 - Sedova
E.A.$'
        74 20 66 72 6F 6D
        20 31 33 38 33 20
        2D 20 53 65 64 6F
        76 61 20 45 2E 41
        2E 24

0030          DATA          ENDS

        ; PĭPsPr PíCĤPsPíCĤP°PjPjC<

0000          CODE          SEGMENT
        ; PμCĤPsC†PμPrCíCĤP° PíPμC†P°C, Pĕ CíC, CĤPsPePĕ
0000 WriteMsg PROC NEAR
0000 B4 09          mov AH, 9
0002 CD 21          int 21h ; P'C<P·PsPI
C„CíPSPĕC†PĕP

        ĕ DOS PíPs PíCĤPμCĤC<PIP°PSPĕCĤ

0004 C3          ret
0005 WriteMsg ENDP

```

```

; P`PsP»PsPIPSP°CŲ PiCbPsC†PμPrCfCbP°
0005      Main      PROC  FAR
0005  1E          push  DS          ;\  PŲPsC...CbP°PSPμPSPë
Pμ P°PrCbPμCfP° PSP°C†P°P»P° PSP PI CfC,PμPePμ
0006  2B C0      sub    AX,AX      ; > PrP»CŲ
PiPsCfP»PμP

rCfCbC%PμPiPs PIPsCfCfC,P°PSPsPIP»PμPSPëCŲ PiPs
0008  50          push  AX          ;/  PePsPjP°PSPrPμ ret
, P·P°PIPμCbCëP°CbC%PμPN° PiCbPsC†PμPrCfCbCf.
0009  B8 ---- R  mov    AX,DATA      ; P-P°PiCbC
fP·PeP° CfPμPiPjPμPSC,PSPsPiPs
000C  8E D8      mov    DS,AX          ;
CbPμPiPëC

fC,CbP° PrP°PSPSC<C...
000E  BA 0000 R  mov    DX, OFFSET HELLO ; P'C<PIPsP
r PSP° CkPeCbP°PS PiPμCbPIPsPN°
0011  E8 0000 R  call   WriteMsg        ; CfC,CbPsP
ePë PiCbPëPIPμC,CfC,PIPëCŲ.
0014  BA 0010 R  mov    DX, OFFSET GREETING ; P'C<PIPsP
r PSP° CkPeCbP°PS PIC,PsCbPsPN°
0017  E8 0000 R  call   WriteMsg        ; CfC,CbPsP
ePë PiCbPëPIPμC,CfC,PIPëCŲ.
001A  CB          ret                ; P'C<C...PsP
r PI DOS PiPs PePsPjP°PSPrPμ,
; PSP°C...PsP

rCŲC%PμPN°CfCŲ PI 1-PsPj CfP»PsPIPμ 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	0030	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

47986 + 459271 Bytes symbol space free

0 Warning Errors
0 Severe Errors