

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 Ассембле-
ра.

Студент гр. 8381

Михайлов Д.А.

Преподаватель

Кирияничиков А.В,

Санкт-Петербург

2020

Цель работы.

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

Ход работы.

Часть 1.

1. Загрузил файлы hello1.asm, hello2.asm, masm.exe, link.exe, afdpro.com, lib.exe из каталога \лаборат_работы в каталог c:\MASM.
2. Запустил программу DOSBox, смонтировал виртуальный диск c: в каталоге \MASM при помощи mount c:\MASM.
3. Просмотрел программу в режиме редактирования, изучил структуру и реализации каждого сегмента программы. Строку-приветствие преобразовал в соответствии со своими личными данными.
4. В DOS перешёл на виртуальный диск при помощи команды c:
5. Протранслировал программу с помощью строки:
 > masm hello1.asm
По ходу трансляции создается объектный файл Hello1.obj. Во время её выполнения ошибок не было обнаружено.
6. Скомпоновал загрузочный модуль Hello1.exe с помощью строки:
 > link Hello1.obj
В результате работы линковщика создается загрузочный модуль Hello1.exe.
7. Загрузил русскую кодовую таблицу символов путём набора строки:
 > keyb ru 866
8. Запустил программу в автоматическом режиме путем набора строки:
 > Hello1.exe
9. Вывод программы:
 > Вас приветствует ст.гр.9382 — Михайлов Д.А..

10. Выполнил запуск программы Hello1.exe в пошаговом режиме с фиксацией используемых регистров и ячеек памяти до и после выполнения каждой команды, используя отладчик и соответственно команду:

> afd 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.

1. 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 (DS) = 19F5 (DX) = 0000 (IP) = 0010	(AX) = 1A07 (DS) = 19F5 (DX) = 0000 (IP) = 0013
0013	MOV DS, AX	8ED8	(AX) = 1A07 (DS) = 19F5 (DX) = 0000 (IP) = 0013	(AX) = 1A07 (DS) = 1A07 (DX) = 0000 (IP) = 0015
0015	MOV DX, 0000	BA0000	(AX) = 1A07 (DS) = 1A07 (DX) = 0000 (IP) = 0015	(AX) = 1A07 (DS) = 1A07 (DX) = 0000 (IP) = 0018
0018	MOV AH,09	B409	(AX) = 1A07 (DS) = 1A07 (DX) = 0000 (IP) = 0018	(AX) = 0907 (DS) = 1A07 (DX) = 0000 (IP) = 001A
001A	INT 21	CD21	(AX) = 0907 (DS) = 1A07 (DX) = 0000 (IP) = 001A	(AX) = 0907 (DS) = 1A07 (DX) = 0000 (IP) = 001C
001C	MOV AH,4C	B44C	(AX) = 0907 (DS) = 1A07 (DX) = 0000 (IP) = 001C	(AX) = 4C07 (DS) = 1A07 (DX) = 0000 (IP) = 001E
001E	INT 21	CD21	(AX) = 4C07 (DS) = 1A07 (DX) = 0000 (IP) = 001E	(AX) = 0000 (DS) = 19F5 (DX) = 0000 (IP) = 0010

Часть 2.

1. Просмотрел программу Hello2.asm в режиме редактирования, изучил ее структуру и реализацию каждого сегмента программы. Строки-приветствия преобразовывал в соответствии со своими личными данными.

2. Выполнил трансляцию программы Hello2.asm с помощью транслятора MASM и команды:

>masm Hello2.asm

В результате чего получился объектный файл Hello2.obj. В процессе трансляции ошибок не обнаружено.

3. Используя линковщик LINK, создал загрузочный модуль Hello2.exe. Использовал команду:

>link Hello2.obj

4. Выполнил программу Hello2.exe в автоматическом режиме и убедился в том, что она работает корректно: в консоль выводится:

>Hello worlds!

>Greetings from student of 9382 Mikhailov Dmitry.

5. Запустил программу Hello2.exe в пошаговом режиме, используя отладчик afd с фиксацией содержимого используемых регистров и ячеек памяти до и после выполнения команд.

Содержимое сегментных регистров до старта программы: CS:11B2, DS:119C, ES:119C, SS:11AC, HS:119C, FS:119C

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) = 119C (SP)=0018 (IP) = 0005 Stack +0 0000 +2 0000 +4 0000	(DS) = 119C (SP)=0016 (IP) = 0006 Stack +0 119C +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 119C +2 0000 +4 0000	(AX) = 0000 (SP)=0014 (IP) = 0009 Stack +0 0000 +2 119C +4 0000
0009	MOV AX, 11AE	B8AE11	(AX) = 0000 (IP) = 0009	(AX) = 11AE (IP) = 000C
000C	MOV DS, AX	8ED8	(AX) = 11AE (DS) = 119C (IP) = 000C	(AX) = 11AE (DS) = 11AE (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 119C +4 0000	(SP)=0012 (IP) = 0000 Stack +0 0014 +2 0000 +4 119C
0000	MOV AH,09	B409	(AX) = 11AE (IP) = 0000	(AX) = 09AE (IP) = 0002
0002	INT 21	CD21	(IP) = 0002	(IP) = 0004
0004	RET	C3	(SP)=0012 (IP) = 0004 Stack +0 0014 +2 0000 +4 119C	(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 119C +4 0000	(SP)=0012 (IP) = 0000 Stack +0 001A +2 0000 +4 119C
0000	MOV AH,09	B409	(AX) = 09AE (IP) = 0000	(AX) = 09AE (IP) = 0002
0002	INT 21	CD21	(IP) = 0002	(IP) = 0004
0004	RET	C3	(SP)=0012 (IP) = 0004 Stack +0 001A +2 0000 +4 119C	(SP)=0014 (IP) = 001A Stack +0 0000 +2 119C +4 0000
001A	RET Far	CB	(SP)=0014 (IP) = 001A Stack +0 0000 +2 119C +4 0000	(SP)=0018 (IP) = 0000 Stack +0 0000 +2 0000 +4 0000

0000	INT 20	CD20	(AX) = 09AE (DS) = 11AE (IP) = 0000	(AX) = 0000 (DS) = 119C (IP) = 0005
------	--------	------	---	---

Выводы.

В результате выполнения лабораторной работы была освоена трансляция, изучено, как происходит выполнение и отладка программ на языке Ассемблера, а также разобраны структуры приведенных в работе программ.

Приложение.

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 Михайлов
Дмитрий',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 *hello2.asm*

```
EOFLine EQU '$'
AStack SEGMENT STACK
DW 12 DUP(?)
AStack ENDS
DATA SEGMENT
HELLO DB 'Здравствуйте!', 0AH, 0DH, EOFLine
GREETING DB 'Вас приветствует ст.гр.9382 - Михайлов
Дмитрий.', 0AH, 0DH, EOFLine
```

```

DATA      ENDS
CODE      SEGMENT
          ASSUME CS:Code DS:DATA SS:STACK
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

```