

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 процессора Intel X86.

Студентка гр. 9381

Сергиенкова А.А.

Преподаватель

Кирияничиков В.М.

Санкт-Петербург

2020

Цель работы.

Освоение отладки, трансляции и выполнения программ на языке Ассемблера.

Задание.

Часть 1.

1. Загрузить файл `hello1.asm` из каталога `\лаборат_работы` в каталог `\MASM`.
2. Просмотреть программу в режиме редактирования, разобраться в структуре и реализации каждого сегмента программы. Непонятные фрагменты прояснить у преподавателя. Строку-приветствие преобразовать в соответствии со своими личными данными.

3. Протранслировать программу с помощью строки

```
> masm имя_файла.asm
```

с созданием объектного файла `имя_файла.obj` и файла диагностических сообщений (файла листинга) `имя_файла.lst`. Объяснить и исправить синтаксические ошибки, если они будут обнаружены транслятором и представлены в файле листинга. Повторить трансляцию программы до получения объектного файла(модуля).

4. Скомпоновать загрузочный модуль (`имя_файла.exe`) с помощью строки

```
> link имя_файла.obj
```

с созданием загрузочного модуля (`имя_файла.exe`) и файла карты памяти (`имя_файла.map`). По карте памяти оценить размещение и длину сегментов программы.

5. Выполнить программу в автоматическом режиме путем набора строки

```
> имя_файла.exe
```

и убедиться в ее работоспособности.

6. Выполнить программу `hello1` в пошаговом режиме под управлением отладчика:

> afd имя_файла.exe

с фиксацией содержимого используемых регистров и ячеек памяти до и после выполнения команды.

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.

Ход работы.

1. Загрузка файла HELLO1.ASM в каталог MASM.
2. Просмотр и изменение программы в режиме редактирования.
(Изменены личные данные.)
3. Трансляция программы с созданием объектного файла hello1.obj и файла диагностических сообщений list.lst с помощью строки `masm hello1.asm`.

```
Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount f /Users/anastasiasergienkova/desktop/tools
Drive F is mounted as local directory /Users/anastasiasergienkova/desktop/tools/

Z:\>f:

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

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

    47998 + 461309 Bytes symbol space free

    0 Warning Errors
    0 Severe Errors

F:\>_
```

4. Линковка загрузочного модуля hello1.exe с созданием загрузочного модуля hello1.exe и файла карты памяти hello1.map.

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

47998 + 461309 Bytes symbol space free

0 Warning Errors
0 Severe Errors

F:\>link hello1.obj

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
Libraries [.LIB]:

F:\>keyb ru 866
Keyboard layout ru loaded for codepage 866

F:\>hello1.exe
Вас приветствует ст.гр.9383 - Сергиенкова А.А.
```

5. Выполнение программы в автоматическом режиме путем набора строки hello1.exe.

```
F:\>hello1.exe
Вас приветствует ст.гр.9383 - Сергиенкова А.А.
```

6. Выполнение программы в пошаговом режиме под управлением отладчика afd. Результаты представлены в таблице 1.

AX 0000	SI 0000	CS 1A05	IP 0010	Stack +0 E4D9	Flags 7202
BX 0000	DI 0000	DS 19F5		+2 0F25	
CX 0051	BP 0000	ES 19F5	HS 19F5	+4 9BD8	OF DF IF SF ZF AF PF CF
DX 0000	SP 0100	SS 1A0B	FS 19F5	+6 1F30	0 0 1 0 0 0 0 0

CMD >				1															
				DS:0000 CD 20 FF 9F 00 EA F0 FE															
				DS:0008 AD DE 1B 05 C5 06 00 00															
				DS:0010 18 01 10 01 18 01 92 01															
				DS:0018 01 01 01 00 02 FF FF FF															
				DS:0020 FF FF FF FF FF FF FF FF															
				DS:0028 FF FF FF FF EB 19 C0 11															
				DS:0030 A2 01 14 00 18 00 F5 19															
				DS:0038 FF FF FF FF 00 00 00 00															
				DS:0040 05 00 00 00 00 00 00 00															
				DS:0048 00 00 00 00 00 00 00 00															

2				0 1 2 3 4 5 6 7 8 9 A B C D E F																
DS:0000				CD 20 FF 9F 00 EA F0 FE AD DE 1B 05 C5 06 00 00																= я.ъЁ■ н ..†...
DS:0010				18 01 10 01 18 01 92 01 01 01 01 00 02 FF FF FF															Т.
DS:0020				FF FF FF FF FF FF FF FF FF FF FF FF EB 19 C0 11															ы. L.
DS:0030				A2 01 14 00 18 00 F5 19 FF FF FF FF 00 00 00 00																В.....й.
DS:0040				05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00															

1 Step	2 ProcStep	3 Retrieve	4 Help ON	5 BRK Menu	6	7 ↑	8 ↓	9 ←	10 →
--------	------------	------------	-----------	------------	---	-----	-----	-----	------

Значения регистров до выполнения программы:

Сегментный регистр	Начальное значение
CS	1A05
DS	19F5
ES	19F5
SS	1A0B

HS = 19F5

FS = 19F5

Stack	+0	E4D9
	+2	0F25
	+4	9BD8
	+6	1F30

Таблица 1.

Адрес команды	Символический код команды	16-ричный команды	Содержимое регистров и ячеек памяти	
			до выполнения команды	после выполнения команды
0010	MOV AX,1A07	B8071A	AX = 0000 IP = 0010	AX = 1A07 IP = 0013
0013	MOV DS,AX	8ED8	AX = 1A07 DS = 19A5 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	IP = 001E	IP = 0010

Часть 2.

1. Файл hello2.asm загружен в папку MASM.
2. Строка в файле отредактирована под личные данные.
3. Трансляция программы с созданием объектного файла hello2.obj и файла диагностических сообщений list.lst с помощью строки masm hello2.asm.

```
Z:\>mount f /Users/anastasiasergienkova/desktop/tools
Drive F already mounted with local directory /Users/anastasiasergienkova/desktop/
/tools/

Z:\>f:

F:\>masm hello2.asm
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.

Object filename [hello2.OBJ]:
Source listing [NUL.LST]: list2
Cross-reference [NUL.CRF]:
hello2.asm(28): warning A4001: Extra characters on line

      48006 + 461301 Bytes symbol space free

      1 Warning Errors
      0 Severe Errors

F:\>
```

4. Линковка загрузочного модуля hello2.exe с созданием загрузочного модуля hello1.exe и файла карты памяти hello2.map.
5. Выполнение программы в автоматическом режиме путем набора строки hello1.exe.

```
F:\>link hello2.obj

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]:

F:\>keyb ru 866
Keyboard layout ru loaded for codepage 866

F:\>hello2.exe
Hello Worlds!
Student from 9383 Сергиенкова А.А.-
F:\>
```

6. Выполнение программы в пошаговом режиме под управлением отладчика afd. Результаты представлены в таблице 2.

Сегментный регистр	Начальное значение
CS	1A0B
DS	19F5
ES	19F5
SS	1A05

Таблица 2.

Адрес команды	Символический код команды	16-ричный команды	Содержимое регистров и ячеек памяти	
			до выполнения команды	после выполнения команды
0005	PUSH DS	1E	IP = 0005 SP = 0018 STACK: +0 0000 +2 0000 +4 0000 +6 0000	IP = 0006 SP = 0016 STACK: +0 19F5 +2 0000 +4 0000 +6 0000
0006	SUB AX,AX	2BC0	AX = 0000 IP = 0006	AX = 0000 IP = 0008
0008	PUSH AX	50	SP = 0016 IP = 0008 STACK: +0 19F5 +2 0000 +4 0000 +6 0000	SP = 0014 IP = 0009 STACK: +0 0000 +2 19F5 +4 0000 +6 0000
0009	MOV AX,11AE	B8AE11	AX = 0000 IP = 0009	AX = 1A07 IP = 000C
000C	MOV DS,AX	8ED8	DS = 19F5 IP = 000C	DS = 1A07 IP = 000E

000E	MOV DX,0000	BA0000	DX = 0000 IP = 000E	DX = 0000 IP = 0011
0011	CALL 0000	E8ECFF	IP = 0011 SP = 0014 STACK: +0 0000 +2 19F5 +4 0000 +6 0000	IP = 0000 SP = 0012 Stack: +0 0014 +2 0000 +4 19F5 +6 0000
0000	MOV AH,09	B409	AX = 1A07 IP = 0000	AX = 0907 IP = 0002
0002	INT 21	CD21	IP = 0002	IP = 0004
0004	RET	C3	IP = 0004 SP = 0012 STACK: +0 0014 +2 0000 +4 19F5 +6 0000	IP = 0014 SP = 0014 STACK: +0 0000 +2 19F5 +4 0000 +6 0000
0014	MOV DX,0010	BA1000	DX = 0000 IP = 0014	DX = 0010 IP = 0017
0017	CALL 0000	E8E6FF	IP = 0017 SP = 0014 STACK: +0 0000 +2 19F5 +4 0000 +6 0000	IP = 0000 SP = 0012 STACK: +0 001A +2 0000 +4 19F5 +6 0000
0000	MOV AH,09	B409	AX = 0907 IP = 0000	AX = 0907 IP = 0002
0002	INT 21	CD21	IP = 0002	IP = 0004
0004	RET	C3	IP = 0004 SP = 0012 STACK: +0 001A +2 0000	IP = 001A SP = 0014 STACK: +0 0000 +2 19F5

			+4 19F5 +6 0000	+4 0000 +6 0000
001A	RET FAR	CB	IP = 001A SP = 0014 CS = 1A0B STACK: +0 0000 +2 19F5 +4 0000 +6 0000	IP = 0000 SP = 0018 CS = 19F5 STACK: +0 0000 +2 0000 +4 0000 +6 0000
0000	INT 20	CD20		

```

AX 0000 SI 0000 CS 1A0B IP 0005 Stack +0 0000 Flags 7202
BX 0000 DI 0000 DS 19F5          +2 0000
CX 0000 BP 0000 ES 19F5 HS 19F5   +4 0000 OF DF IF SF ZF AF PF CF
DX 0000 SP 0018 SS 1A05 FS 19F5   +6 0000 0 0 1 0 0 0 0 0

CMD >
Program terminated OK
0005 1E      PUSH    DS
0006 2BC0    SUB     AX,AX
0008 50      PUSH    AX
0009 B8071A  MOV     AX,1A07
000C 8ED8    MOV     DS,AX
000E BA0000  MOV     DX,0000
0011 E8ECFF  CALL    0000
0014 BA1000  MOV     DX,0010
DS:0000  CD 20 FF 9F 00 EA FF FF  AD DE 1B 05 C5 06 00 00
DS:0008  AD DE 1B 05 C5 06 00 00
DS:0010  18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF
DS:0018  01 01 01 00 02 FF FF FF
DS:0020  FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 E4 11
DS:0028  FF FF FF FF EB 19 E4 11
DS:0030  A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00
DS:0038  FF FF FF FF 00 00 00 00
DS:0040  05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
DS:0048  00 00 00 00 00 00 00 00

2 0 1 2 3 4 5 6 7 8 9 A B C D E F
DS:0000  CD 20 FF 9F 00 EA FF FF  AD DE 1B 05 C5 06 00 00  = Я.ъ н.+.
DS:0010  18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF  .....Т. ....
DS:0020  FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 E4 11  ..... ы.ф.
DS:0030  A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00  В.....і. ....
DS:0040  05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  .....

1 Step 2ProcStep 3Retrieve 4Help ON 5BRK Menu 6 7↑ 8↓ 9← 10⇒

```

Вывод

В ходе лабораторной работы была освоена трансляция, отладка и выполнение программ на языке Ассемблера. Результатом выполнения лабораторной работы являются программы Hello1.exe и Hello2.exe.

HELLO1.ASM

```
; HELLO1.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 'Вас приветствует ст.гр.9383 - Сергиенкова А.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      ; # 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
```

HELLO2.ASM

; 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

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 9383 Сергиенкова А.А.- \$'

DATA ENDS

; Код программы

CODE SEGMENT

ASSUME CS:Code DS:DATA SS:AStack

; 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

```

```

Microsoft (R) Macro Assembler Version 5.10                               9/16/20 18:58:56
                                                                           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 82 A0 E1 20 AF E0 DB 'Вас приветствует ст.гр.9383 – Сергиенков
а А.А.',13,10,'$'
A8 A2 A5 E2 E1 E2
A2 E3 A5 E2 20 E1
E2 2E A3 E0 2E 39
33 38 33 20 2D 20
91 A5 E0 A3 A8 A5
AD AA AE A2 A0 20
80 2F 80 2F 0D 0A

```

```

0000 82 A0 E1 20 AF E0  DB 'Вас приветствует ст.гр.9383 - Сергиенков
      а А.А.',13,10,'$'
      AB A2 A5 E2 E1 E2
      A2 E3 A5 E2 20 E1
      E2 2E A3 E0 2E 39
      33 38 33 20 2D 20
      91 A5 E0 A3 A8 A5
      AD AA AE A2 A0 20
      80 2E 80 2E 0D 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
0005 BA 0000 R      mov dx, OFFSET Greeting ; Загр
Microsoft (R) Macro Assembler Version 5.10 9/16/20 18:58:56
                                           Page      1-2

      узка в dx смещения
                                           ; адрес
      а текста приветствия
000B      DisplayGreeting:
000B B4 09          mov ah, 9              ; # 09
      нкции DOS печати строки
000A CD 21          int 21h                ; вывод
      на экран приветствия
000C B4 4C          mov ah, 4Ch            ; # 4C
      нкции DOS завершения программы
000E CD 21          int 21h                ; завер
      шение программы и выход в DOS
      END
Microsoft (R) Macro Assembler Version 5.10 9/16/20 18:58:56
                                           Symbols-1

Segments and Groups:
.....
      N a m e          Length  Align  Combine Class
.....
GROUP . . . . .      GROUP
_DATA . . . . .      0031  WORD  PUBLIC 'DATA'
_STACK . . . . .      0100  PARA  STACK 'STACK'
_TEXT . . . . .      0010  WORD  PUBLIC 'CODE'

Symbols:
      N a m e          Type  Value  Attr
.....
DISPLAYGREETING . . . . .  L NEAR  000B  _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

47998 + 461309 Bytes symbol space free

0 Warning Errors
0 Severe Errors

```

```

Microsoft (R) Macro Assembler Version 5.10                                9/16/20 50:38:56
                                                                 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

0000      AStack      SEGMENT STACK
0000 000C[      DW 12 DUP(?)      ; Отводится 12 слов п
; амяти
; ???
; ]

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 9383 Сергиенкова А.А
.- $'
74 20 66 72 6F 6D
20 39 33 38 33 20
91 A5 E0 A3 AB A5
AD AA AE A2 A0 20
80 2E 80 2E 2D 20
24
0035      DATA      ENDS

; Код программы

0000      CODE      SEGMENT
; ASSUME C5:Code DS:DATA SS:AStack
hello2.asm(28): warning A4001: Extra characters on line
; 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

; Головная процедура
0005      Main      PROC FAR

Microsoft (R) Macro Assembler Version 5.10                                9/16/20 50:38:56
                                                                 Page      1-2

```

```

0005      Main      PROC  FAR
Microsoft (R) Macro Assembler Version 5.10                               9/16/20 50:38:56
                                                                           Page      1-2

0005 1E          push  DS      ;\ Сохранение адреса
начала PSP в стеке
0006 2B C0          sub  AX,AX    ; > для последующего в
остановления по
0008 50          push  AX      ;/ команде get, 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          get              ; 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                               9/16/20 50:38:56
                                                                           Symbols-1

```

Segments and Groups:

N a m e	Length	Align	Combine	Class
ASTACK	0018	PARA	STACK	
CODE	001B	PARA	NONE	
DATA	0035	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

```

48046 + 461301 Bytes symbol space free

```

0 Warning Errors
0 Severe  Errors

```