

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: Трансляция, отладка и выполнение программ на языке
Ассемблера

Студентка гр. 0383

Петровская Е.С.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

Цель работы.

Изучение трансляции, отладки и выполнения 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. 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.

Выполнение работы.

Были просмотрены и протранслированы программы hello1.asm и hello2.asm, скомпонованы их загрузочные модули, произведен запуск программ в автоматическом режиме. Работа обеих программ под управлением отладчика расписана в таблицах 1 и в соответственно.

```
C:\>hello1.exe
You're welcomed by the student of the 0383 group - Petrovskaya E.S.
```

Рис.1 — результат работы программы hello1.exe в автоматическом режиме

```
C:\>hello2.exe
Hello Worlds!
Student from 0383 - Petrovskaya Evgeniya
```

Рис.2 — результат работы программы hello2.exe в автоматическом режиме

Таблица 1 – Результаты прогона программы hello1 под управлением отладчика

Начальное содержимое сегментных регистров		(CS) = 1A05	(DS) = 19F5	(ES) = 19F5	(SS) = 1A0C
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	(AX) = 4C07	

			(IP) = 001C	(IP) = 001E
001E	INT 21	CD21	(IP) = 001E	

Таблица 2 – Результаты прогона программы hello2 под управлением отладчика

Начальное содержимое сегментных регистров		(CS) = 1A0B	(DS) = 19F5	(ES) = 19F5	(SS) = 1A05
Адрес Команды	Символический код команды	16-ричный код команды	Содержимое регистров и ячеек памяти		
			до выполнения	после выполнения	
0005	PUSH DS	1E	(AX) = 0000 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0018 (IP) = 0005 Stack +0 0000	(AX) = 0000 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0016 (IP) = 0006 Stack +0 19F5	
0006	SUB AX, AX	2BC0	(AX) = 0000 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0016 (IP) = 0006 Stack +0 19F5	(AX) = 0000 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0016 (IP) = 0008 Stack +0 19F5	
0008	PUSH AX	50	(AX) = 0000 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0016 (IP) = 0008 Stack +0 19F5	(AX) = 0000 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0014 (IP) = 0009 Stack +0 0000 Stack +2 19F5	

0009	MOV AX, 1A07	B8071A	(AX) = 0000 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0014 (IP) = 0009 Stack +0 0000 Stack +2 19F5	(AX) = 1A07 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0014 (IP) = 000C Stack +0 0000 Stack +2 19F5
000C	MOV DS, AX	8ED8	(AX) = 1A07 (DX) = 0000 (DS) = 19F5 (CS) = 1A0B (SP) = 0014 (IP) = 000C Stack +0 0000 Stack +2 19F5	(AX) = 1A07 (DX) = 0000 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 000E Stack +0 0000 Stack +2 19F5
000E	MOV DX, 0000	BA0000	(AX) = 1A07 (DX) = 0000 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 000E Stack +0 0000 Stack +2 19F5	(AX) = 1A07 (DX) = 0000 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 0011 Stack +0 0000 Stack +2 19F5
0011	CALL 0000	E8ECFF	(AX) = 1A07 (DX) = 0000 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 0011 Stack +0 0000 Stack +2 19F5	(AX) = 0907 (DX) = 0000 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 0014 Stack +0 0000 Stack +2 19F5

0014	MOV DX, 0010	BA1000	(AX) = 0907 (DX) = 0000 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 0014 Stack +0 0000 Stack +2 19F5	(AX) = 0907 (DX) = 0010 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 0017 Stack +0 0000 Stack +2 19F5
0017	CALL 0000	E8E6FF	(AX) = 0907 (DX) = 0010 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 0017 Stack +0 0000 Stack +2 19F5	(AX) = 0907 (DX) = 0010 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 001A Stack +0 0000 Stack +2 19F5
001A	RET Far	CB	(AX) = 0907 (DX) = 0010 (DS) = 1A07 (CS) = 1A0B (SP) = 0014 (IP) = 001A Stack +0 0000 Stack +2 19F5	(AX) = 0907 (DX) = 0010 (SP) = 0018 (CS) = 19F5 (IP) = 0000 Stack +0 0000 Stack +2 0000
0000	INT 20	CD20	(AX) = 0907 (DX) = 0010 (SP) = 0018 (CS) = 19F5 (IP) = 0000 Stack +0 0000	Программа завершена

Выводы.

В ходе выполнения лабораторной работы произошло ознакомление с базовыми функциями эмулятора DOSBOX, особенностями трансляции, компоновки и запуска программ на языке Ассемблера, а также их отладки с помощью утилиты ADF PRO.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: 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 'You`re welcomed by the student of the 0383 group -
Petrovskaya E.S.',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
```

Название файла: 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
```

```

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 0383 - Petrovskaya Evgeniya$'
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      ;\ 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

```

Название файла: hello1.lst

```

#Microsoft      (R)      Macro      Assembler      Version      5.10
9/15/21 07:51:36

```

Page

1-1

[illegible]

* * * * *

; ԺՀԺՃԾ»ՆԺ·ԺՃԺ²Ժ°ՆԺԺԾ»Ն Ն ԺՀԺՃԾ
 ՇԵՃՆՆՆ ՆՆԺԾԺ°ՆԺՃԺՃ ԺԺԺ; "ԺՆԺ²ԺՃԺՃ ՆՆ
 ՆԺՃԺ°ԺՃ"

$$\begin{aligned} & - \partial \ddot{Y} \partial \pm \partial \mu \ddot{N} \partial \zeta \partial \mu \ddot{N} \partial \ddot{Z} \partial^2 \partial \circ \partial \mu \ddot{N} \partial^2 \ddot{N} \partial \\ & {}^2 \partial \ddot{Y} \partial \ddot{Z} \partial \alpha \partial^\circ \ddot{N} \ddot{I} \partial^\circ \ddot{N} (\partial^\circ \partial \alpha \ddot{N}) \ddot{N} (\ddot{N} (\partial \ddot{Y} \partial^\circ \partial \ddot{Z} \ddot{N}) \partial \ddot{Z} \partial \mathbb{C} \partial^2 \partial \ddot{Y} \partial \\ & \partial \ddot{Y} \partial^2, \end{aligned}$$

; - ÑÑĐµĐ±ÑĐµÑ Đ·Đ°ĐŽĐ°ĐæĐŽÑ
Đ² ÑĐµĐ³ĐŽÑÑÑĐµ ah ĐæĐŸĐÆĐµÑĐ° ÑÑĐæĐ°ÑĐ
ŽĐŽ=09h,

; ÑÑÑĐŸĐ°ĐŽ;

;

* * * * *

; $\partial\partial^\circ\partial\check{Z}\partial^\circ\partial\alpha\partial\check{Z}\partial\mu$ $\check{N}\partial\mu\partial^3\partial\mathcal{E}\partial\mu\partial\alpha\check{N}\partial\check{Y}\partial^2$ $\partial\zeta\partial\check{Y}\partial\check{Z}$ $\partial\partial$

```

        Di
        .MODEL    SMALL
        ;  ÐÐŸÐŽÐµÐ»Ñ Ð¿Ð°ÐœÑÑÐŽ - SMALL(ÐÐ°Ð»Ð°Ñ)
        .STACK    100h
        ;  ÐÑÐ²ÐµÑÑÐŽ Ð¿ŸÐŽ Ð¿ÑÐµÐ° 256 Ð±Ð°¹Ñ
        .DATA
        ;  ÐÐ°ÑÐ°Ð»»Ÿ ÑÐµÐ³ÐœÐµÐœÑÐ° ÐŽÐ°ÐœÐœÑÑ
0000      Greeting LABEL BYTE
        ;  ÐœÐµÐ°ÑÑ Ð¿ÑŽÐ²ÐµÑÑÑÐ²ÐŽÑ
0000      59 6F 75 60 72 65      DB 'You`re welcomed by the student of
the 03
                                83 group - Petrovskaya E.S.',13,10,'$'
20 77 65 6C 63 6F
6D 65 64 20 62 79
20 74 68 65 20 73
74 75 64 65 6E 74
20 6F 66 20 74 68
65 20 30 33 38 33
20 67 72 6F 75 70

```

1-2

```

20 2D 20 50 65 74
72 6F 76 73 6B 61
79 61 20 45 2E 53
2E 0D 0A 24

.CODE ; 0000h
0000 B8 ---- R    mov ax, @data ; 0000h
0003 8E D8        mov ds, ax ; 0003h
; 0003h
0005 BA 0000 R    mov dx, OFFSET Greeting ; 0005h
0008                DisplayGreeting:
0008 B4 09        mov ah, 9 ; 0008h
; 0008h
000A CD 21        int 21h ; 000Ah
000C B4 4C        mov ah, 4ch ; 000Ch
; 000Ch
000E CD 21        int 21h ; 000Eh

```

END

```
#Microsoft      (R)      Macro      Assembler      Version      5.10
9/15/21 07:51:36
```

Symbol

s-1

Segments and Groups:

	N a m e	Length	Align	Combine	Class
DGROUP	GROUP			
_DATA	0046	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 + 459266 Bytes symbol space free

0 Severe Errors

Название файла: hello2.lst

```
#Microsoft (R) Macro Assembler Version 5.10
9/15/21 07:47:40
```

Page

1-1

```
; HELLO2 - Ð£ÑÐµÐ±ÐœÐ°Ñ Ð¿ÑÐŸÐ³ÑÐ°ÐœÐœÐ° N2
Ð»Ð°Ð±Ð°.ÑÐ°Ð±Ð°.#1 Ð¿ÐŸ Ð¿¿¿ÑÑÐ¿¿¿Ð¿Ð»Ð¿¿ÐœÐµ "Ð
ÑÑÐ¿¿ÑÐµÐ°ÑÑÑÐ° Ð°ÐŸŸÐœÐ¿ÑÑÑÐµÑÑÐ°"
; ÐÑÐŸÐ³ÑÐ°ÐœÐœÐ° Ð¿ÑÑÐ¿ÐŸÐ»ÑÐ°ÑÐ
ÑÑ Ð¿¿ÑÐŸÑÐµÐ¿¿ÑÑÑ Ð¿¿»Ñ Ð¿¿µÑÑÐ°ÑÐ¿ ÑÑ
ÑÐŸÐ°Ð¿
;
; ÐœÐœÐ;Ðœ ÐÐ ÐÐÐ ÐÐÐÐ«
```

```

= 0024          EOFLine    EQU    '$'          ;

```

ÐÐ¿ÑÐµÐŹÐµÐ»»ÐµÐœÐ

ŽĐμ ŃĐŽĐƉĐ²ĐŸĐ»ŃĐœĐŸĐ¹ Đ°ĐŸĐœŃŃĐ°ĐœŃŃ
 ; "ĐĐŸĐœĐμŃ ŃŃ
 ŃĐŸĐ°ĐŽ"

; $\mathfrak{D} \vdash \tilde{N} \mathfrak{D} \mu \mathfrak{D}^{\circ}$ $\mathfrak{D} \vdash \tilde{N} \mathfrak{D} \ddot{Y} \mathfrak{D}^3 \tilde{N} \mathfrak{D}^{\circ} \mathfrak{D} \mathfrak{C} \mathfrak{D} \mathfrak{C} \tilde{N}$

```
ASSUME CS:CODE, SS:AStack
```

```

0000          AStack      SEGMENT      STACK
0000 000C[                DW      12  DUP('!')          ;

```

ĐÑĐ²ĐỖĐŽĐŽÑÑÑ

12 ÑĐ»ĐŸĐ² Đ¿Đ°Đ€ÑÑĐŽ

0021

1

```
0018             AStack     ENDS
```

; ÐÐ°ÐæÐæÑÐµ Ð¸ÑÐŸÐ³ÑÐ°°ÐæÐæÑ

0000	DATA	SEGMENT
------	------	---------

;
ÑÑ

```

0000 48 65 6C 6C 6F 20  HELLO          DB 'Hello Worlds!', 0AH,
0DH,E0FLine

```

57 6F 72 6C 64 73

21 0A 0D 24

```

0010 53 74 75 64 65 6E  GREETING      DB  'Student  from 0383 -
Petrovskaya Evgeniya$'

```

74 20 66 72 6F 6D

20 30 33 38 33 20


```

        2D 20 50 65 74 72  

        6F 76 73 6B 61 79  

        61 20 45 76 67 65  

        6E 69 79 61 24  

0039          DATA          ENDS  
  

; ÐÐŸÐŽ Ð¿ÑÐŸÐ³ÑÐ°ÐœÐœÑ  
  

0000          CODE          SEGMENT  

; ÐÑÐŸÑÐµÐŽÑÑÐ° Ð¿ÐµÑÑÐ°ÑŽ ÑÑÑÐŸÐ°ÐŽ  

0000 WriteMsg PROC NEAR  

0000 B4 09 mov AH,9  

0002 CD 21 int 21h ; ÐÑÐ·ÐŸÐ²  

ÑÑÐœÐ°ÑÐŽÐŽÐŽ  

Ž DOS Ð¿ÐŸ Ð¿ÑÐµÑÑÑÐ²Ð°ÐœÐŽÑ

```

1-2

```

0004 C3          ret
0005          WriteMsg ENDP

; ԾԾԿԾ»ԾԿԾ²ԾԾԾ°Ն ԾՀՆԾԿՆԾԾԾՆՆԾ°
0005 Main      PROC FAR
0005 1E          push DS          ; \ ԾԿԿՆՆԾ°ԾԾԾԾԾԾԾ
ԾԾ Ծ°ԾՆՆԾԾԾ° ԾԾԾ°ՆԾԾ°Ծ»Ծ° PSP Ծ² ՆՆԾԾԾ°ԾԾ
0006 2B C0          sub AX,AX          ; > ԾՃԾ»Ն
ԾՀԾԿՆՆԾ»ԾԾԾ
ŽՆՆՆԾԾԾ³ԾԿ Ծ²ԾԿՆՆՆԾ°ԾԾԾԾ²Ծ»ԾԾԾԾՃՆ ԾՀԾԿ
0008 50          push AX          ; / Ծ°ԾԿԾԾԾ°ԾԾՃԾԾ
ret
, Ծ·Ծ°Ծ²ԾԾՆՆԾ°ՆՆԾԾԾ¹ ԾՀՆԾԿՆԾԾՆՆՆ.
0009 B8 ---- R    mov AX,DATA          ; ԾԾ°Ծ³ՆՆ
Ծ·Ծ°Ծ° ՆԾԾԾ³ԾԾԾԾՆՆԾԾԿԾԾ³ԾԿ
000C 8E D8          mov DS,AX          ;
ՆԾԾԾ³ԾՃՆ
ՆՆԾ° ԾՃԾ°ԾԾԾՆՆ.
000E BA 0000 R    mov DX, OFFSET HELLO ; ԾՆԾ²ԾԿԾ
Ž ԾԾԾ° ՆԾԾ°Ն(Ծ°ԾԾ ԾՀԾԾՆ(Ծ²ԾԿԾ¹
0011 E8 0000 R    call WriteMsg          ; ՆՆՆԾԿԾ
°ԾՃ ԾՀՆՆՆԾ²ԾԾՆՆՆԾ²ԾՃՆ.
0014 BA 0010 R    mov DX, OFFSET GREETING ; ԾՆԾ²ԾԿԾ
Ž ԾԾԾ° ՆԾԾ°Ն(Ծ°ԾԾ Ծ²Ն(ԾԿՆ(ԾԿԾ¹
0017 E8 0000 R    call WriteMsg          ; ՆՆՆԾԿԾ
°ԾՃ ԾՀՆՆՆԾ²ԾԾՆՆՆԾ²ԾՃՆ.
001A CB          ret          ; ԾՆՆԾԿԾ
Ž Ծ² DOS ԾՀԾԿ Ծ°ԾԿԾԾԾ°ԾԾՃԾԾ,
; ԾԾԾ°ՆԾԿԾ
ŽՆՆՆԾԾ¹ՆՆ Ծ² 1-ԾԿԾ ՆԾ»ԾԿԾ²ԾԾ 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		0039	PARA	NONE

Symbols:

	N a m e	Type	Value	Attr
EOFLINE		NUMBER		0024
GREETING		L BYTE		0010 DATA
HELLO		L BYTE		0000 DATA
0016 MAIN		F PROC		0005 CODE Length =
0005 WRITEMSG		N PROC		0000 CODE Length =
@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

```