



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Лабораторна робота №1

Системне програмне забезпечення

Виконав
студент групи ІТ-03:

Чабан А.Є.

Перевірив:

Стельмах О.П.

Київ 2021

Тема: створення програм на асемблері

Завдання:

1. Для програми, наведеної вище, створити файл типу .asm. Ця програма не має засобів виводу даних, тому правильність її виконання треба перевірити за допомогою td.exe.
2. Скомпілювати програму, включивши потрібні опції для налагоджувача та створення файлу лістингу типу .lst.
3. Ознайомитись зі структурою файлу .lst. За вказівкою викладача, для певної команди асемблера розглянути структуру машинної команди і навести її у звіті.
4. Скомпонувати .obj-файл програми. Включити опції для налагодження та створення .map-файлу.
5. Занести до звіту адреси початку та кінця всіх сегментів з .map-файлу.
6. Завантажити до налагоджувача td.exe одержаний .exe-файл програми.
7. У вікні CPU у полі DUMP знайти початкову адресу сегмента даних та записати його до звіту. Знайти масиви SOURCE та DEST. Дані у масиві SOURCE подаються у шістнадцятковій системі.
8. У покроковому режимі за допомогою клавіші F7 виконати програму. Одержані результати у масиві DEST показати викладачеві.

Код програми:

```
STSEG SEGMENT PARA STACK "STACK"
DB 64 DUP ( "STACK" )
STSEG ENDS
DSEG SEGMENT PARA PUBLIC "DATA"
SOURCE DB 10, 20, 30, 40
DEST DB 4 DUP ( "?" )
DSEG ENDS
CSEG SEGMENT PARA PUBLIC "CODE"
MAIN PROC FAR
ASSUME CS: CSEG, DS: DSEG, SS: STSEG
; адреса повернення
PUSH DS
MOV AX, 0 ; або XOR AX, AX
PUSH AX
```

```
; ініціалізація DS
MOV AX, DSEG
MOV DS, AX
; обнуляємо масив
MOV DEST, 0
MOV DEST+1, 0
MOV DEST+2, 0
MOV DEST+3, 0
; пересилання
MOV AL, SOURCE
MOV DEST+3, AL
MOV AL, SOURCE+1
MOV DEST+2, AL
MOV AL, SOURCE+2
MOV DEST+1, AL
MOV AL, SOURCE+3
MOV DEST, AL
RET
MAIN ENDP
CSEG ENDS
END MAIN
```

Компіляція програми:

```
F:\>tasm /l LAB1C2
Turbo Assembler Version 3.2i Copyright (c) 1988, 1992 Borland International
Serial No:   Tester:

Assembling file:  LAB1C2.ASM
Error messages:   None
Warning messages: None
Passes:           1
Remaining memory: 465k
```

Вміст файлу лістингу:

```

1 Turbo Assembler Version 3.2 09/08/21 13:55:08 Page 1
2 LAB1C2.ASM
3
4
5
6 1 0000 STSEG SEGMENT PARA STACK "STACK"
7 2 0000 40*(53 54 41 43 4B) DB 64 DUP ( "STACK" )
8 3 0140 STSEG ENDS
9 4 0000 DSEG SEGMENT PARA PUBLIC "DATA"
10 5 0000 0A 14 1E 28 SOURCE DB 10, 20, 30, 40
11 6 0004 04*(3F) DEST DB 4 DUP ( "?" )
12 7 0008 DSEG ENDS
13 8 0000 CSEG SEGMENT PARA PUBLIC "CODE"
14 9 0000 MAIN PROC FAR
15 10 ASSUME CS: CSEG, DS: DSEG, SS: STSEG
16 11 ; адреса повернення
17 12 0000 1E PUSH DS
18 13 0001 B8 0000 MOV AX, 0 ; або XOR AX, AX
19 14 0004 50 PUSH AX
20 15 ; ініціалізація DS
21 16 0005 B8 0000 MOV AX, DSEG
22 17 0008 8E D8 MOV DS, AX
23 18 ; обнуляємо масив
24 19 000A C6 06 0004r 00 MOV DEST, 0
25 20 000F C6 06 0005r 00 MOV DEST+1, 0
26 21 0014 C6 06 0006r 00 MOV DEST+2, 0
27 22 0019 C6 06 0007r 00 MOV DEST+3, 0
28 23 ; пересилання
29 24 001E A0 0000r MOV AL, SOURCE
30 25 0021 A2 0007r MOV DEST+3, AL
31 26 0024 A0 0001r MOV AL, SOURCE+1
32 27 0027 A2 0006r MOV DEST+2, AL
33 28 002A A0 0002r MOV AL, SOURCE+2
34 29 002D A2 0005r MOV DEST+1, AL
35 30 0030 A0 0003r MOV AL, SOURCE+3
36 31 0033 A2 0004r MOV DEST, AL
37 32 0036 CB RET
38 33 0037 MAIN ENDP
39 34 0037 CSEG ENDS
40 35 END MAIN

```

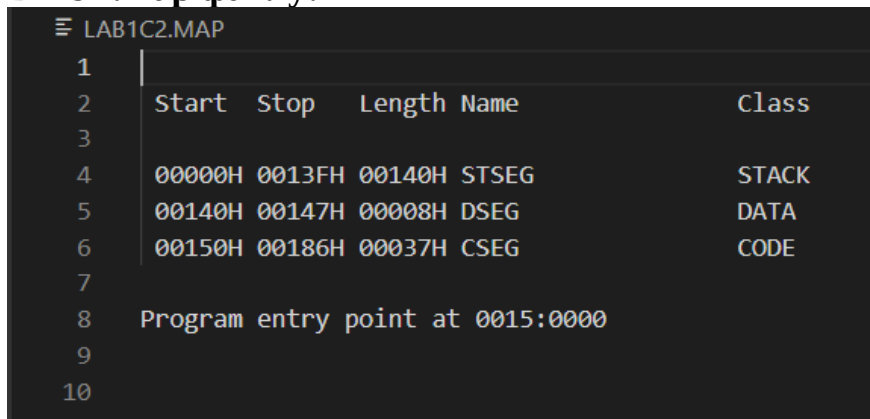
```

41 Turbo Assembler Version 3.2 09/08/21 13:55:08 Page 2
42 Symbol Table

```

Symbol Name	Type	Value
??DATE	Text	"09/08/21"
??FILENAME	Text	"LAB1C2 "
??TIME	Text	"13:55:08"
??VERSION	Number	0314
@CPU	Text	0101H
@CURSEG	Text	CSEG
@FILENAME	Text	LAB1C2
@WORDSIZE	Text	2
DEST	Byte	DSEG:0004
MAIN	Far	CSEG:0000
SOURCE	Byte	DSEG:0000
Groups & Segments	Bit Size Align	Combine Class
CSEG	16 0037 Para	Public CODE
DSEG	16 0008 Para	Public DATA
STSEG	16 0140 Para	Stack STACK

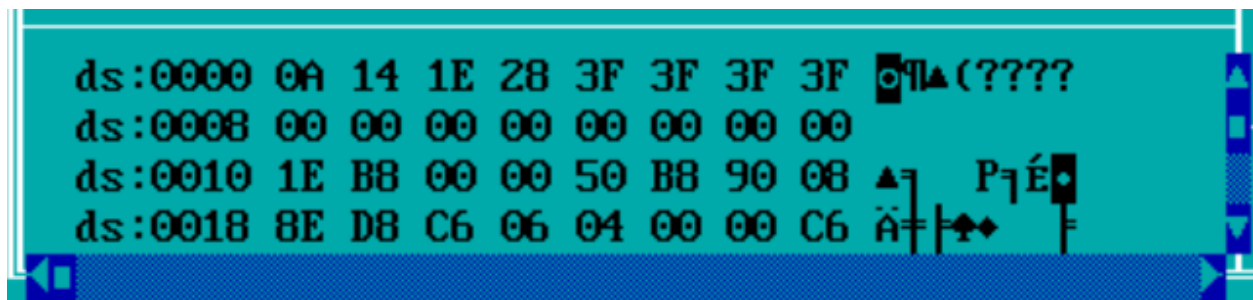
Вміст .map файлу:



LAB1C2.MAP

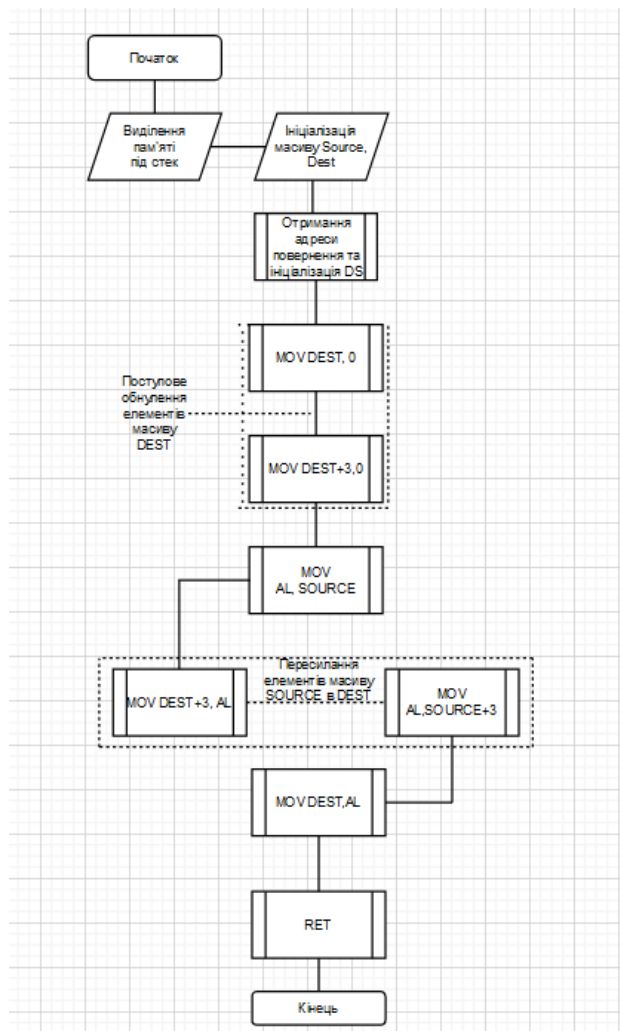
1	
2	Start Stop Length Name Class
3	
4	00000H 0013FH 00140H STSEG STACK
5	00140H 00147H 00008H DSEG DATA
6	00150H 00186H 00037H CSEG CODE
7	
8	Program entry point at 0015:0000
9	
10	

Вміст дампу пам'яті:



ds:0000	0A	14	1E	28	3F	3F	3F	3F	07	1A	(????)
ds:0008	00	00	00	00	00	00	00	00	00		
ds:0010	1E	B8	00	00	50	B8	90	08	1A	77	P7É
ds:0018	8E	D8	C6	06	04	00	00	C6	1A	77	77

Схема функціонування програми:



Висновок:

1. В текстовому редакторі було створено файл типу .asm.
2. Скомпілював програму, включивши потрібні опції для налагоджувача та створення файлу лістингу типу .lst.
3. Ознайомився зі структурою файлу .lst. Розглянув структури машинних команд.
4. Після усунення помилок, скомпонував .obj-файл програми, включивши опції для налагодження та створення .map-файлу.
5. Відкрив файл карти пам'яті (.map-файл) та подивився на адреси початку та кінця всіх сегментів програми.
6. Завантажив програму налагоджувача td.exe та мій одержаний .exe-файл програми.

7. У вікні CPU у полі DUMP подивився на початкову адресу сегмента даних.
В

сегменті даних знайшов масиви SOURCE та DEST. Дані у масиві SOURCE
подаються

у шістнадцятковій системі.

8. У покроковому режимі за допомогою клавіші F7 виконав програму.

Програма

коректно виконує поставлену задачу.