

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної
техніки
Кафедра інформатики та програмної інженерії

Звіт до комп'ютерного практикуму з дисципліни
«Системне програмне забезпечення»

Прийняв
асистент кафедри ІІІ
Пархоменко А.В.
“19” Березня 2023 р.

Виконав
Студент групи ІІІ-14
Хільчук А.В.

Комп'ютерний практикум №1

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.

Виконання:

Текст 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
```

```
PUSH DS  
MOV AX, 0  
PUSH AX
```

```

MOV AX, DSEG
MOV DS, AX

MOV DEST, 0
MOV DEST+1, 0
MOV DEST+2, 0
MOV DEST+3, 0

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

```

Введені та отримані результати:

Вміст .lst файлу:

Turbo Assembler Version 4.0 02/25/23 18:52:40 Page 1
KM1.asm

```

1 0000          STSEG SEGMENT PARA STACK
"STACK"
2 0000 40*(53 54 41 43 4B) DB 64 DUP ("STACK")
3 0140          STSEG ENDS
4
5 0000          DSEG SEGMENT PARA PUBLIC "DATA"
6 0000 0A 14 1E 28 SOURCE DB 10, 20, 30, 40
7 0004 04*(3F)      DEST DB 4 DUP ("??")
8 0008          DSEG ENDS
9
10 0000         CSEG SEGMENT PARA PUBLIC "CODE"
11
12 0000         MAIN PROC FAR

```

13	ASSUME CS: CSEG, DS: DSEG, SS:
STSEG	
14	
15 0000 1E	PUSH DS
16 0001 B8 0000	MOV AX, 0
17 0004 50	PUSH AX
18	
19 0005 B8 0000s	MOV AX, DSEG
20 0008 8E D8	MOV DS, AX
21	
22 000A C6 06 0004r 00	MOV DEST, 0
23 000F C6 06 0005r 00	MOV DEST+1, 0
24 0014 C6 06 0006r 00	MOV DEST+2, 0
25 0019 C6 06 0007r 00	MOV DEST+3, 0
26	
27 001E A0 0000r	MOV AL, SOURCE
28 0021 A2 0007r	MOV DEST+3, AL
29 0024 A0 0001r	MOV AL, SOURCE+1
30 0027 A2 0006r	MOV DEST+2, AL
31 002A A0 0002r	MOV AL, SOURCE+2
32 002D A2 0005r	MOV DEST+1, AL
33 0030 A0 0003r	MOV AL, SOURCE+3
34 0033 A2 0004r	MOV DEST, AL
35 0036 CB	RET
36 0037	MAIN ENDP
37 0037	CSEG ENDS
38	END MAIN

Turbo Assembler Version 4.0 02/25/23 18:52:40 Page 2
Symbol Table

Symbol Name	Type Value
??DATE	Text "02/25/23"
??FILENAME	Text "KM1"
??TIME	Text "18:52:40"
??VERSION	Number 0400
@CPU	Text 0101H
@CURSEG	Text CSEG
@FILENAME	Text KM1

@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:

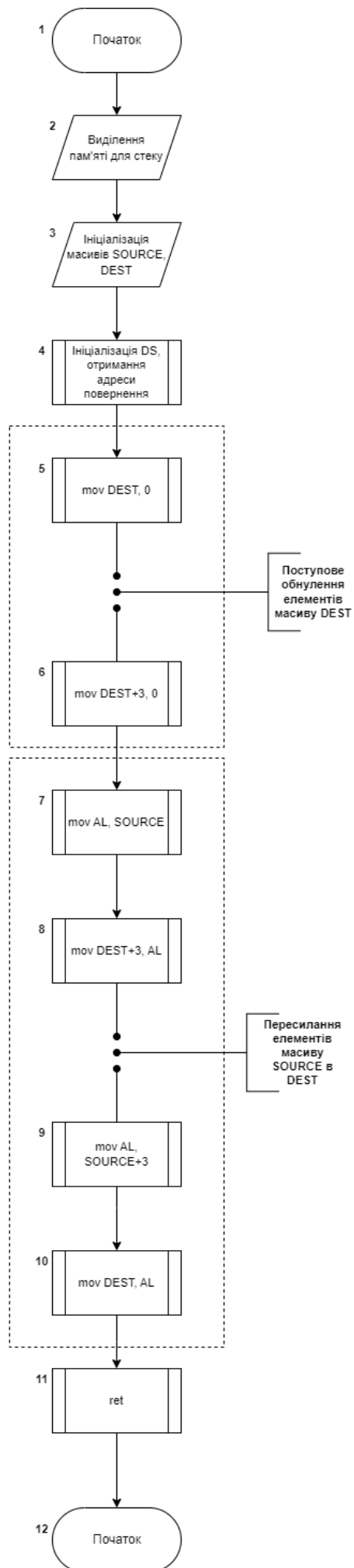
Start	Stop	Length	Name	Class
00000H	0013FH	00140H	STSEG	STACK
00140H	00147H	00008H	DSEG	DATA
00150H	00186H	00037H	CSEG	CODE

Address	Publics by Name
---------	-----------------

Address	Publics by Value
---------	------------------

Program entry point at 0015:0000

Схема функціонування програми:



Вікно DUMP:

```
ds:0000 0A 14 1E 28 3F 3F 3F 3F 07 00 00 00
ds:0008 00 00 00 00 00 00 00 00
ds:0010 1E B8 00 00 50 B8 D1 52 07 00 00 00
ds:0018 8E D8 C6 06 04 00 00 C6 07 00 00 00
```

До виконання програми:

Масив SOURCE:

```
ds:0000 0A 14 1E 28 3F 3F 3F 3F 07 00 00 00
ds:0008 00 00 00 00 00 00 00 00
ds:0010 1E B8 00 00 50 B8 D1 52 07 00 00 00
ds:0018 8E D8 C6 06 04 00 00 C6 07 00 00 00
```

Масив DEST

```
ds:0000 0A 14 1E 28 3F 3F 3F 3F 07 00 00 00
ds:0008 00 00 00 00 00 00 00 00
ds:0010 1E B8 00 00 50 B8 D1 52 07 00 00 00
ds:0018 8E D8 C6 06 04 00 00 C6 07 00 00 00
```

Після виконання програми

Масив DEST

```
ds:0000 0A 14 1E 28 28 1E 14 0A 07 00 00 00
ds:0008 00 00 00 00 00 00 00 00
ds:0010 1E B8 00 00 50 B8 D1 52 07 00 00 00
ds:0018 8E D8 C6 06 04 00 00 C6 07 00 00 00
```

Висновок:

Отож, у ході виконання лабораторної роботи було проаналізовано поставлене завдання, репліковано код, зазначений у методичних матеріалах, та назначено файлу з ним тип .asm. Було скопійовано програму із зазначенням у параметрах опції для налагодження, а також назви відповідних результуючих файлів. Ознайомлено зі структурою .lst файлу та структурою машинних команд. Опісля було скомпоновано .obj файл за допомогою TLINK із зазначенням назв вихідних файлів та опцій для відладки та формування .map файлу; оглянуто вміст даного файлу. Урешті-решт було відкрито отриманий .exe файл у налагоджувачі та проглянуто у вікні CPU сектор DUMP та знайдено оголошені та ініціалізовані масиви SOURCE та DEST; проведено покрокове виконання програми та запевнено в правильності її роботи.