НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

*Звіти до комп’ютерних практикумів дисципліни*

«Системне програмне забезпечення»

**Прийняв Виконав**

**доцент кафедри ІПІ Студент групи ІП-11**

**Лісовиченко О.І. Головня О.Р.**

**“06” березня 2023 р.**

Київ – 2023

**Комп’ютерний практикум №1**

**Тема**: Створення програм на асемблері

**Завдання**:

1. Для програми, наведеної нище, 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.

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.

8. У покроковому режимі за допомогою клавіші F7 виконати програму. Одержані  результати у масиві DEST показати викладачеві.

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 2.51 03/01/23 13:40:48 Page 1

program.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 0000 DSEG SEGMENT PARA PUBLIC "DATA"

5 0000 0A 14 1E 28 SOURCE DB 10, 20, 30, 40

6 0004 04\*(3F) DEST DB 4 DUP ( "?" )

7 0008 DSEG ENDS

8 0000 CSEG SEGMENT PARA PUBLIC "CODE"

9 0000 MAIN PROC FAR

10 ASSUME CS: CSEG, DS: DSEG, SS: STSEG

11

12 0000 1E PUSH DS

13 0001 B8 0000 MOV AX, 0 ; або XOR AX, AX

14 0004 50 PUSH AX

15

16 0005 B8 0000s MOV AX, DSEG

17 0008 8E D8 MOV DS, AX

18

19 000A C6 06 0004r 00 MOV DEST, 0

20 000F C6 06 0005r 00 MOV DEST+1, 0

21 0014 C6 06 0006r 00 MOV DEST+2, 0

22 0019 C6 06 0007r 00 MOV DEST+3, 0

23

24 001E A0 0000r MOV AL, SOURCE

25 0021 A2 0007r MOV DEST+3, AL

26 0024 A0 0001r MOV AL, SOURCE+1

27 0027 A2 0006r MOV DEST+2, AL

28 002A A0 0002r MOV AL, SOURCE+2

29 002D A2 0005r MOV DEST+1, AL

30 0030 A0 0003r MOV AL, SOURCE+3

31 0033 A2 0004r MOV DEST, AL

32 0036 CB RET

33 0037 MAIN ENDP

34 0037 CSEG ENDS

35 END MAIN

Turbo Assembler Version 2.51 03/01/23 13:46:48 Page 2

Symbol Table

Symbol Name Type Value

??DATE Text "02/20/23"

??FILENAME Text "program "

??TIME Text "13:46:48"

??VERSION Number 0205

@CPU Text 0101H++

@CURSEG Text CSEG

@FILENAME Text PROGRAM

@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:**

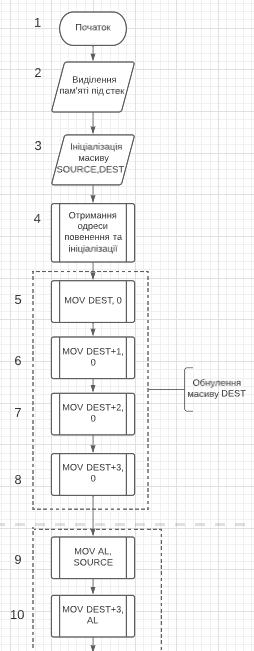
Start Stop Length Name Class

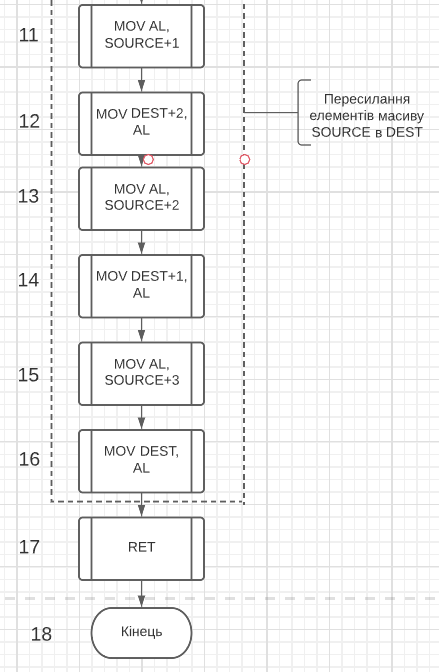
00000H 0013FH 00140H STSEG STACK

00140H 00147H 00008H DSEG DATA

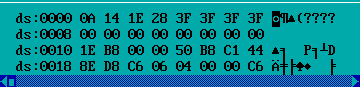
00150H 00186H 00037H CSEG CODE

**Схема функціонування програми**

****

****

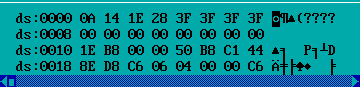
**Вікно DUMP**



**До виконання програми:**

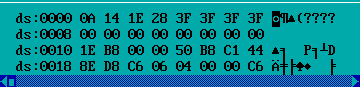
**Масив SOURCE:**

**(перші 4 звурху)**



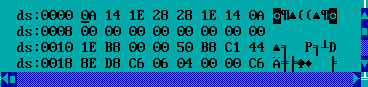
**Масив DEST**

**(останні 4 зверху)**



**Після виконання програми**

**Масив DEST**



**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 памяті (.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 даниз знайшов масиви SOURCE та DEST.

8. У покроковому режимі за допомогою клавіші F7 виконав програму. Програма  коректно виконує поставлену задачу.