МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ



**Дніпропетровський національний університет  
залізничного транспорту імені академіка В. Лазаряна**

Кафедра «Комп’ютерні інформаційні технології»

**Лабораторна робота №4**

**з дисципліни «архітектура комп’ютера»**

**на тему: «Реалізація розгалужень мовою Асемблер.»**

Виконав: студент гр. ПЗ1911

Сіньков Г.О.

Прийняла: ас. каф. КІТ

Нежуміра О. І.

Дніпро, 2019

**Лабораторна робота № 4**

Тема. Реалізація циклічних алгоритмів мовою Асемблера.

Мета. Набути практичних навичок розробки програм циклічної структури.

**1.Постановка задачи**

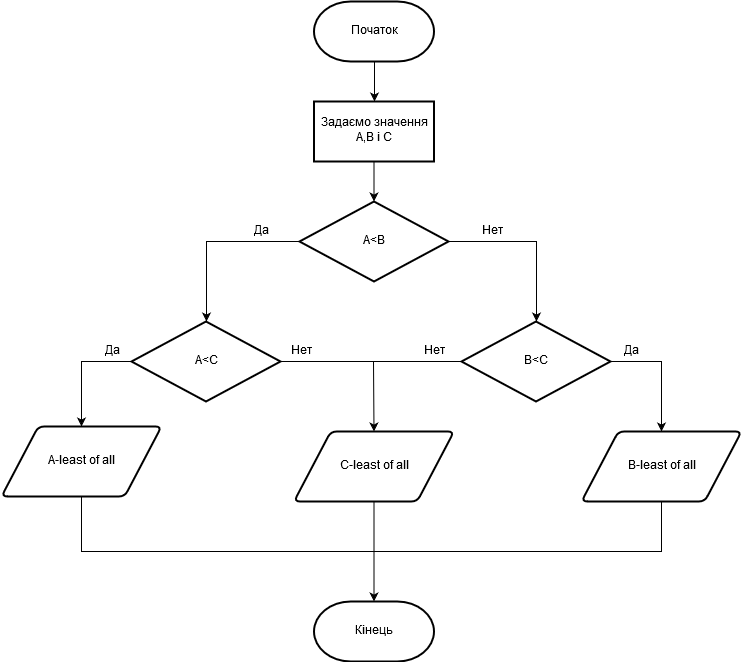
1.1 Загальне завдання

Завдання наведено в таблиці. Вибір завдання здійснюється згідно з порядковим номером у списку підгрупи. Вхідні дані задаються у сегменті даних.

1.2. Індивідуальне завдання

|  |  |
| --- | --- |
| № п/п | Зміст завдання |
| 7 | Для трьох довільних чисел вказати найменше. Результат подається у вигляді повідомлення. |

**2. Алгоритм розв’язання задачі**



**3. Лістинг програми**

Turbo Assembler Version 3.0 11/26/19 22:06:17 Page 1

lab4.asm

1 ; сегмент даних

2 0000 data segment para public 'data'

3 0000 0D 0A 41 2D 6C 65 61+ A\_answer db 13,10,'A-least of all',13,10,13,10,'$'

4 73 74 20 6F 66 20 61+

5 6C 6C 0D 0A 0D 0A 24

6 0015 0D 0A 42 2D 6C 65 61+ B\_answer db 13,10,'B-least of all',13,10,13,10,'$'

7 73 74 20 6F 66 20 61+

8 6C 6C 0D 0A 0D 0A 24

9 002A 0D 0A 43 2D 6C 65 61+ C\_answer db 13,10,'C-least of all',13,10,13,10,'$'

10 73 74 20 6F 66 20 61+

11 6C 6C 0D 0A 0D 0A 24

12 003F 19 A db 25

13 0040 39 B db 57

14 0041 9A C db 154

15 0042 data ends

16 ; сегмент стека

17 0000 stk segment stack

18 0000 0100\*(??) db 256 dup (?) ;поле для стека розміром 256 байт

19 0100 stk ends

20 ; сегмент команд

21 0000 code segment para public 'code'

22 assume cs:code, ds:data, ss:stk

23 0000 start:

24 0000 B8 0000s mov ax, data ;ініціалізація сегментного регістра ds

25 0003 8E D8 mov ds, ax ; на програмний сегмент даних data

26

27

28 0005 A0 003Fr mov al, A

29 0008 8A 26 0040r mov ah, B

30

31 000C 3A C4 cmp al, ah ; A<B

32 000E 77 0E ja \_c1 ; проверка знака

33

34 0010 8A 26 0041r mov ah, C

35 0014 3A C4 cmp al, ah ; A<C

36 0016 77 13 ja \_else

37 0018 BA 0000r lea dx, A\_answer

38 001B EB 11 90 jmp \_exit

39

40 001E \_c1:

41 001E A0 0041r mov al, C

42 0021 3A E0 cmp ah, al ; B<C

43 0023 77 06 ja \_else

44 0025 BA 0015r lea dx, B\_answer

45 0028 EB 04 90 jmp \_exit

46

47 002B \_else:

48 002B BA 002Ar lea dx, C\_answer

49

50 002E \_exit:

51 002E B4 09 mov ah, 09h ; блок вивелення повідомлення

52 0030 CD 21 int 21h

53 0032 B8 4C00 mov ax, 4C00h

54 0035 CD 21 int 21h

55 0037 code ends

56 end start

Turbo Assembler Version 3.0 11/26/19 22:06:17 Page 2

Symbol Table

Symbol Name Type Value

??DATE Text "11/26/19"

??FILENAME Text "lab4 "

??TIME Text "22:06:17"

??VERSION Number 0300

@CPU Text 0101H

@CURSEG Text CODE

@FILENAME Text LAB4

@WORDSIZE Text 2

A Byte DATA:003F

A\_ANSWER Byte DATA:0000

B Byte DATA:0040

B\_ANSWER Byte DATA:0015

C Byte DATA:0041

C\_ANSWER Byte DATA:002A

START Near CODE:0000

\_C1 Near CODE:001E

\_ELSE Near CODE:002B

\_EXIT Near CODE:002E

Groups & Segments Bit Size Align Combine Class

CODE 16 0037 Para Public CODE

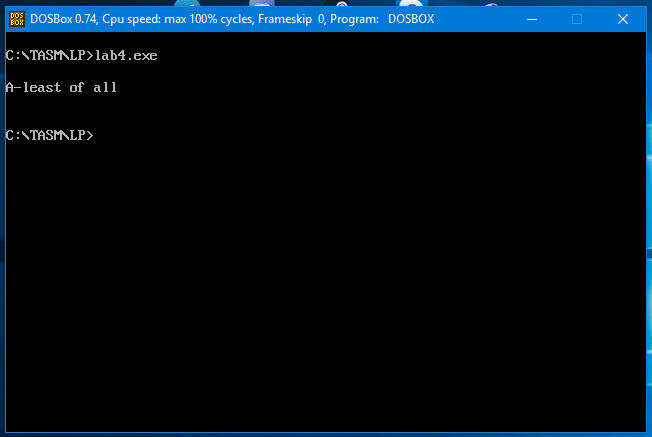
DATA 16 0042 Para Public DATA

STK 16 0100 Para Stack

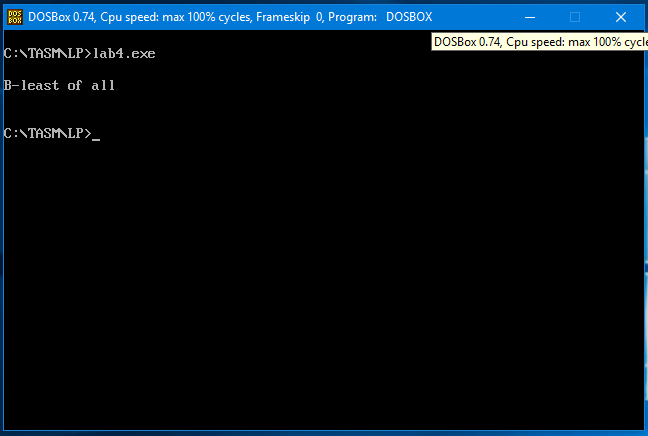
|  |  |  |  |
| --- | --- | --- | --- |
| № | Вхідні дані | Умови переходу | Очікувані результати |
| 1 | A = 25, B = 57, C = 154 | A<B(да)  A<C(да) | A – менше всіх |
| 2 | A = 57, B = 25, C = 154 | A<B(ні)  B<C(да) | B – менше всіх |
| 3 | A = 57, B = 154, C = 25 | A<B(ні, але по умові в програмі так)  B<C(ні) | C – менше всіх |

**5. Результати виконання програми**

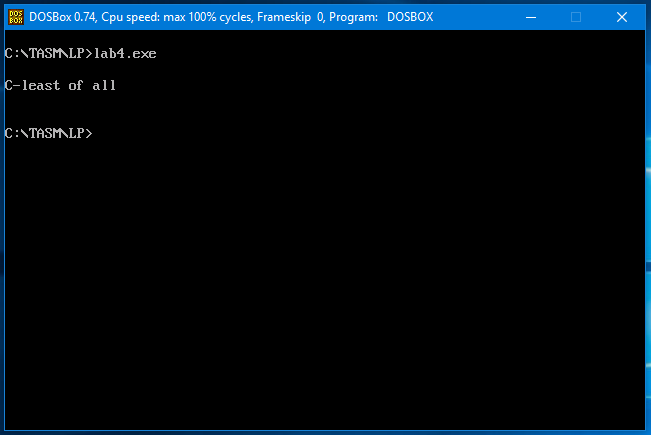
Перший тест:



Другий тест:



Третій тест:



**6. Висновок:** при створені коду с реалізацією розгалуженням потрібно спочатку правильно встановити умови та команду, яка відповідає умові заданої в задачі. В даній задачі я для власної зручності використовував команди умовного переходу протилежні для моєї умови, для того щоб при виконані умови програма не переходила на якусь мітку, а читалась поспіль команди.