

Міністерство освіти і науки України
Національний університет «Запорізька політехніка»
кафедра програмних засобів

ЗВІТ
з лабораторної роботи №2
з дисципліни «Архітектура програмування та низькорівневе програмування»
на тему: «Структури програм на мові Ассемблера»
Варіант №13

Виконав:
студентка групи КНТ-132

Кочева Мілена

Прийняв:
ст. викладач

О. І. Качан

Запоріжжя
2023

1. Мета та завдання до роботи

1. Провести практичне ознайомлення з компілятором TASM та лінкувальником TLINK і ознайомитися з їх аргументами, параметрами та способом використання.

2. У текстовому редакторі створити початковий код на мові асемблера для exe-програм з наступними вимогами до коду:

- реалізувати виведення до консолі будь-якого друкованого символу;
- на етапі завершення виконання програми реалізувати повернення до операційної системи коду помилки зі значенням згідно варіанту як у ЛР №1 (регістр BX);
- обов'язково у потрібних місцях коду прописувати коментарі;
- у якості імен сегментів та точки входу використовувати зрозумілі, не довгі але оригінальні назви.

3. У текстовому редакторі створити початковий код на мові асемблера для com-програм з наступними вимогами до коду:

- створити дві версії коду com-програм:
 - а) без використання модульних процедур;
 - б) з використанням модульної процедури;
- реалізувати виведення до консолі будь-якого друкованого символу;
- на етапі завершення виконання програми реалізувати повернення до операційної системи коду помилки зі значенням згідно варіанту як у ЛР №1 (регістр BX);
- обов'язково у потрібних місцях коду прописувати коментарі;
- у якості імен сегментів та точки входу використовувати зрозумілі, не довгі але оригінальні назви.

4. Над створеними початковими кодами програм виконати процеси компіляції, лінкування та позбавитись від помилок (якщо вони є).

5. Виконати демонстрацію структури програми у відлагоджувачі DEBUG:

- дизасемблювати виконуваний код програми;
- вказати на місця початку/кінця наступних частин програми: PSP, CODE

Segment, DATA Segment, STACK Segment;

- знайти у виконуваному коді символ, який повинен бути виведено до консолі;
- знайти у виконуваному коді значення, яке має бути передано до змінної коду помилки операційної системи.

6. Виконати запуск програм у консолі з демонстрацією виведеного символу та кодом помилки. Перевірити коректність значення коду помилки згідно варіанту.

7. Оформити звіт згідно ДСТУ 3008:2015 та завантажити поряд з ZIP-архівом власноруч створених початкових кодів програм. Документ звіту та архів повинні мати у назві Група_Прізвище_НомерЛР.

$$BX = 160 - N = 160 - 13 = 147$$

2. Виконана робота

Код до програми яку в майбутньому будемо компілювати:



```
редактирование laba2-1.asm - Far
C:\TASM\laba2-1.asm
.model small

sseg segment para stack 'stack'
    db 256 dup(?)
sseg ends

dseg segment para public 'data'
    M db 'M'
dseg ends

csegment segment para public 'code'
    assume cs:cseg,ss:sseg,es:nothing

start proc far

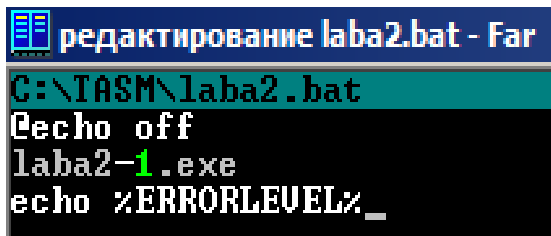
    assume ds:dseg
    mov bx,dseg
    mov ds,bx

    mov ah,02h
    mov dl,M
    int 21h

    mov ah,4Ch
    mov al,147
    int 21h

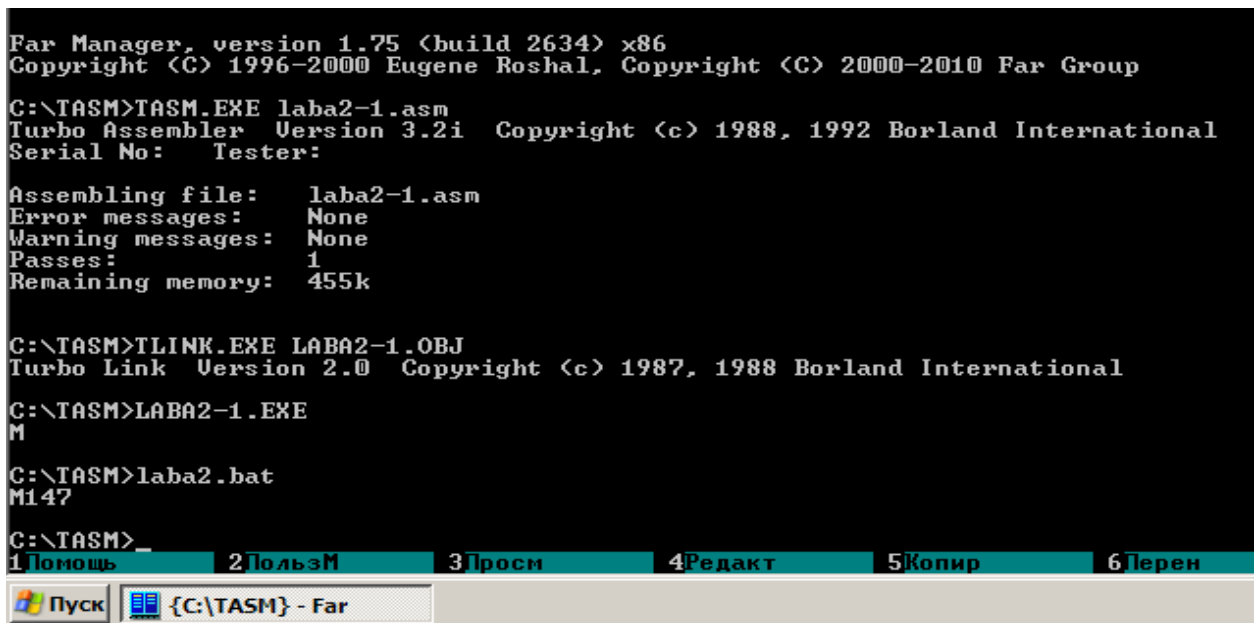
start endp
cseg ends
end start
```

Вміст файлу типу .bat, який буде виводити задану літеру в кодї вже з помилкою:



```
редактирование laba2.bat - Far
C:\TASM\laba2.bat
Echo off
laba2-1.exe
echo %ERRORLEVEL%_
```

Компіляція через TASM та LINK, та запуск файлу типу .exe та .bat:



```
Far Manager, version 1.75 (build 2634) x86
Copyright (C) 1996-2000 Eugene Roshal, Copyright (C) 2000-2010 Far Group

C:\TASM>TASM.EXE laba2-1.asm
Turbo Assembler Version 3.2i Copyright (c) 1988, 1992 Borland International
Serial No: Tester:

Assembling file: laba2-1.asm
Error messages: None
Warning messages: None
Passes: 1
Remaining memory: 455k

C:\TASM>TLINK.EXE LABA2-1.OBJ
Turbo Link Version 2.0 Copyright (c) 1987, 1988 Borland International

C:\TASM>LABA2-1.EXE
M

C:\TASM>laba2.bat
M147

C:\TASM>_
1.Помощь 2.ПользаМ 3.Просм 4.Редакт 5.Копир 6.Перен
Пуск {C:\TASM} - Far
```

Программа без використання модульних процедур:

```
редактирование laba2-2.asm - Far
C:\TASM\laba2-2.asm
cseg segment para public 'code'
    assume ss:cseg,ds:cseg,ss:cseg
START:
    M db 'M'

    mov ah,02h
    mov dl,M
    int 21h

    mov dl,13
    int 21h

    mov dl,10
    int 21h

    mov ah,4Ch
    mov al,147
    int 21h

cseg ends
end START
```

Програма з використанням модульної процедури

```
редактирование laba2-3.asm - Far
C:\TASM\laba2-3.asm
.model small

cseg segment para public 'code'
    assume ss:cseg,cs:cseg,ds:cseg
    org 100h

START:
    jmp short milena
    M db 'M'

milena proc near

    mov ah,02h
    mov dl,M
    int 21h

    mov ah,4Ch
    mov al,147
    int 21h

    call knt
    ret

milena endp

knt proc near

    mov ah,02h
    mov dl,13
    int 21h

    mov ah,4Ch
    mov dl,10
    int 21h

knt endp
cseg ends
end START
```

Дизасемблюємо виконуваний код програми:

```
C:\WINDOWS\system32\cmd.exe - debug c:\tasm\laba2-1.exe
Microsoft Windows XP [Версия 5.1.2600]
(C) Корпорация Майкрософт, 1985-2001.

C:\Documents and Settings\user>debug c:\tasm\laba2-1.exe
-u
0B74:0000 BB760B      MOV     BX,0B76
0B74:0003 8EDB      MOV     DS,BX
0B74:0005 B402      MOV     AH,02
0B74:0007 8A160000    MOV     DL,[0000]
0B74:000B CD21      INT     21
0B74:000D B44C      MOV     AH,4C
0B74:000F B093      MOV     AL,93
0B74:0011 CD21      INT     21
0B74:0013 0000      ADD     [BX+SI],AL
0B74:0015 0000      ADD     [BX+SI],AL
0B74:0017 0000      ADD     [BX+SI],AL
0B74:0019 0000      ADD     [BX+SI],AL
0B74:001B 0000      ADD     [BX+SI],AL
0B74:001D 0000      ADD     [BX+SI],AL
0B74:001F 004D26    ADD     [DI+26],CL
-
```

CODE Segment:

```
C:\WINDOWS\system32\cmd.exe - debug c:\tasm\laba2-1.exe
0B74:0015 0000      ADD     [BX+SI],AL
0B74:0017 0000      ADD     [BX+SI],AL
0B74:0019 0000      ADD     [BX+SI],AL
0B74:001B 0000      ADD     [BX+SI],AL
0B74:001D 0000      ADD     [BX+SI],AL
0B74:001F 004D26    ADD     [DI+26],CL
-d cs:0
0B74:0000 BB 76 0B 8E DB B4 02 8A-16 00 00 CD 21 B4 4C B0      .v.....!.L.
0B74:0010 93 CD 21 00 00 00 00 00-00 00 00 00 00 00 00      ..!.....>
0B74:0020 4D 26 88 55 04 EB 3E 3C-04 75 0F 2E A1 04 94 40      M&.U..>.u....@
0B74:0030 26 89 45 04 26 8C 5D 06-EB 2B 26 89 75 04 26 8C      &.E.&.l..+&.u.&.
0B74:0040 4D 06 50 26 F6 47 02 01-74 04 B0 04 EB 09 26 F6      M.P&.G..t....&.
0B74:0050 47 02 02 74 05 B0 02 E8-13 01 58 26 F6 47 02 10      G..t.....X&.G..
0B74:0060 74 03 E8 E2 00 5F C3 2E-C6 06 D0 94 00 50 26 8B      t.....P&.
0B74:0070 07 0B C0 75 18 50 53 52-57 2E C7 06 F7 93 09 00      ...u.PSRW.....
-d
0B74:0080 B4 FF B0 03 E8 5B FF 5F-5A 5B 58 EB 30 A9 00 10      .....[. _Z[X.0...
0B74:0090 74 12 2E C7 06 F7 93 00-00 E8 80 03 2E 83 3E F7      t.....>.
0B74:00A0 93 09 75 19 A9 00 08 74-17 2E C7 06 F7 93 00 00      ..u....t.....
0B74:00B0 E8 8E 04 2E 83 3E F7 93-09 75 02 EB 03 EB 6F 90      .....>...u....o.
0B74:00C0 A9 00 80 74 12 2E C7 06-F7 93 00 00 E8 2F 01 2E      ..t...../..
0B74:00D0 83 3E F7 93 09 75 57 A9-00 40 74 12 2E C7 06 F7      .>...uW...@t....
0B74:00E0 93 00 00 E8 F4 00 2E 83-3E F7 93 09 75 40 A9 00      .....>...u@..
0B74:00F0 01 74 15 2E C7 06 F7 93-00 00 E8 9C 05 E8 1F 06      .t.....
```

DATA Segment:

```
-d ds:0
0B64:0000 CD 20 FF 9F 00 9A F0 FE-1D F0 4F 03 63 05 8A 03 . . . . .0.c...
0B64:0010 63 05 17 03 63 05 52 05-01 01 01 00 02 FF FF FF c...c.R.....
0B64:0020 FF FF FF FF FF FF FF FF-FF FF FF FF 27 0B 4C 01 . . . . .'.L.
0B64:0030 3A 0A 14 00 18 00 64 0B-FF FF FF FF 00 00 00 00 :.....d.....
0B64:0040 05 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
0B64:0050 CD 21 CB 00 00 00 00 00-00 00 00 00 20 20 20 20 .!.....
0B64:0060 20 20 20 20 20 20 20 20-00 00 00 00 20 20 20 20 .....
0B64:0070 20 20 20 20 20 20 20 20-00 00 00 00 00 00 00 00 .....
-d
0B64:0080 00 0D 63 3A 5C 74 61 73-6D 5C 6C 61 62 61 32 2D ..c:\tasm\laba2-
0B64:0090 31 2E 65 78 65 0D A5 20-A4 AB EF 20 A1 A2 A7 AE 1.exe.....
0B64:00A0 A2 AE A3 AE 20 AF AE E0-E2 A0 20 A2 A2 AE A4 A0 ....
0B64:00B0 2D AC EB A2 AE A4 A0 20-53 42 2C 20 AD A0 AF E0 -.....SB,....
0B64:00C0 A8 AC A5 E0 3A 0D EC 0D-E0 EB A5 0D 00 00 00 00 .....
0B64:00D0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0B64:00E0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0B64:00F0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
```

STACK Segment:

```
C:\WINDOWS\system32\cmd.exe - debug c:\tasm\laba2-1.exe
0B74:00A0 93 09 75 19 A9 00 08 74-17 2E C7 06 F7 93 00 00 ..u...t.....
0B74:00B0 E8 8E 04 2E 83 3E F7 93-09 75 02 EB 03 EB 6F 90 .....>...u...o
0B74:00C0 A9 00 80 74 12 2E C7 06-F7 93 00 00 E8 2F 01 2E ..t...../...
0B74:00D0 83 3E F7 93 09 75 57 A9-00 40 74 12 2E C7 06 F7 .>...uW...et...
0B74:00E0 93 00 00 E8 F4 00 2E 83-3E F7 93 09 75 40 A9 00 .....>...u@...
0B74:00F0 01 74 15 2E C7 06 F7 93-00 00 E8 9C 05 E8 1F 06 .t.....
-d ss:0
0B77:0000 26 89 45 04 26 8C 5D 06-EB 2B 26 89 75 04 26 8C &.E.&.l...+&.u.&.
0B77:0010 4D 06 50 26 F6 47 02 01-74 04 B0 04 EB 09 26 F6 M.P&.G...t...&.
0B77:0020 47 02 02 74 05 B0 02 E8-13 01 58 26 F6 47 02 10 G...t...X&.G...
0B77:0030 74 03 E8 E2 00 5F C3 2E-C6 06 D0 94 00 50 26 8B t.....P&...
0B77:0040 07 0B C0 75 18 50 53 52-57 2E C7 06 F7 93 09 00 ....u.PSRW.....
0B77:0050 B4 FF B0 03 E8 5B FF 5F-5A 5B 58 EB 30 A9 00 10 .....l...Z[X.0...
0B77:0060 74 12 2E C7 06 F7 93 00-00 E8 80 03 2E 83 3E F7 t.....>...
0B77:0070 93 09 75 19 A9 00 08 74-17 2E C7 06 F7 93 00 00 ..u...t.....
-d
0B77:0080 E8 8E 04 2E 83 3E F7 93-09 75 02 EB 03 EB 6F 90 .....>...u...o
0B77:0090 A9 00 80 74 12 2E C7 06-F7 93 00 00 E8 2F 01 2E ..t...../...
0B77:00A0 83 3E F7 93 09 75 57 A9-00 40 74 12 2E C7 06 F7 .>...uW...et...
0B77:00B0 93 00 00 E8 F4 00 2E 83-3E F7 93 09 75 40 A9 00 .....>...u@...
0B77:00C0 01 74 15 2E C7 06 F7 93-00 00 E8 9C 05 E8 1F 06 .t.....
0B77:00D0 2E 83 3E F7 93 09 75 26-A9 00 02 74 12 2E C7 06 ..>...u&...t...
0B77:00E0 F7 93 00 00 E8 82 05 2E-83 3E F7 93 09 75 0F A9 .....>...u...
0B77:00F0 00 20 74 0A 2E C7 06 F7-93 00 00 E8 E2 01 00 00 .t.....
-d
0B77:0100 3E D0 94 01 75 0F 2E 83-3E F7 93 00 75 07 2E C7 .>...u...>...u...
0B77:0110 06 F7 93 09 00 58 C3 50-56 2E 8A 04 0A C0 74 1A .....X.PU.....t.
0B77:0120 3C 3A 75 0D 2E 80 7C 01-00 75 06 2E C6 04 00 EB <:u...!...u.....
0B77:0130 09 E8 E6 06 73 01 46 46-EB DF 5E 58 C3 56 52 8A .....s.FF...^X.UR.
0B77:0140 D0 2E 8A 04 E8 D3 06 72-0C 0A C0 74 0C E8 0C 00 .....r...t....
0B77:0150 2E 88 04 EB 01 46 46 EB-E8 5A 5E C3 3C 80 73 0C .....FF...Z^.<.s.
0B77:0160 3C 61 72 45 3C 7A 77 41-24 DF EB 3D 53 06 57 8D <arE<zwA$...=S.W.
0B77:0170 3E C2 94 80 FA 04 74 04-8D 3E BD 94 2E 38 15 74 .>.....t...>...8.t
-d
0B77:0180 17 50 51 52 0E 07 B4 65-8A C2 BB FF FF B9 05 00 .PQR...e.....
0B77:0190 BA FF FF CD 21 5A 59 58-2E 8B 5D 01 2E 8E 45 03 ....!ZYX...l...E.
0B77:01A0 43 43 2C 80 26 D7 5F 07-5B C3 50 2E 80 0E 03 94 CC,..&...l.P....
0B77:01B0 80 2E 80 26 03 94 FD 2E-8A 04 3C 2B 74 0A 3C 2D ...&.....<+t.<-
0B77:01C0 75 07 2E 80 0E 03 94 02-46 E8 02 00 58 C3 50 51 u.....F...X.PQ
0B77:01D0 52 56 33 C9 33 D2 53 2E-8A 04 0A C0 74 42 E8 F1 RU3.3.S.....tB..
0B77:01E0 00 72 39 32 E4 8B E8 D1-E2 D1 D1 E8 D2 00 72 2C .r92.....r,
0B77:01F0 8B DA 8B C1 D1 E2 D1 D1-E8 C5 00 72 1F D1 E2 D1 .....r....
```


3. Висновки:

Таким чином ми провели практичне ознайомлення з компілятором TASM та лінкувальником TLINK і я ознайомилась з їх аргументами, параметрами та способом використання.