

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

ІКНІ
Кафедра ПЗ

ЗВІТ

до лабораторної роботи № 6

на тему: *“Опрацювання рядка символів засобами асемблера мікропроцесорів x86. Робота з файлами”*
з дисципліни: “Архітектура комп’ютера”

Лектор:

доцент кафедри ПЗ
Крук О.Г.

Виконав:

студент групи ПЗ-22
Коваленко Д.М.

Прийняв:

доцент кафедри ПЗ
Крук О.Г.

«_____» _____ 2022 р.
 Σ =

Тема. Опрацювання рядка символів засобами асемблера мікропроцесорів x86. Робота з файлами.

Мета. Освоїти команди асемблера для роботи з рядками символів; опанувати функції Win32 для роботи з файлами; розвинути навички складання програми для опрацювання рядка символів та програми для створення, записування і читання текстового файла; відтранслювати і виконати в режимі відлагодження програми, складені відповідно до свого індивідуального завдання.

Індивідуальне завдання

ПЗ-21		ПЗ-22		ПЗ-23	
№	Послідовність	№	Послідовність	№	Послідовність
1	П ₃ , П ₅ , П ₁ , П ₄ , П ₇ , П ₂ , П ₆	1	П ₂ , П ₇ , П ₁ , П ₄ , П ₅ , П ₃ , П ₆	1	П ₇ , П ₆ , П ₁ , П ₄ , П ₃ , П ₂ , П ₅
2	П ₅ , П ₁ , П ₄ , П ₇ , П ₂ , П ₆ , П ₃	2	П ₇ , П ₁ , П ₄ , П ₅ , П ₃ , П ₆ , П ₂	2	П ₆ , П ₁ , П ₄ , П ₃ , П ₂ , П ₅ , П ₇
3	П ₁ , П ₄ , П ₇ , П ₂ , П ₆ , П ₃ , П ₅	3	П ₁ , П ₄ , П ₅ , П ₃ , П ₆ , П ₂ , П ₇	3	П ₁ , П ₄ , П ₃ , П ₂ , П ₅ , П ₇ , П ₆
4	П ₄ , П ₇ , П ₂ , П ₆ , П ₃ , П ₅ , П ₁	4	П ₄ , П ₅ , П ₃ , П ₆ , П ₂ , П ₇ , П ₁	4	П ₄ , П ₃ , П ₂ , П ₅ , П ₇ , П ₆ , П ₁
5	П ₇ , П ₂ , П ₆ , П ₃ , П ₅ , П ₁ , П ₄	5	П ₅ , П ₃ , П ₆ , П ₂ , П ₇ , П ₁ , П ₄	5	П ₃ , П ₂ , П ₅ , П ₇ , П ₆ , П ₁ , П ₄
6	П ₆ , П ₃ , П ₅ , П ₁ , П ₄ , П ₇ , П ₂	6	П ₆ , П ₂ , П ₇ , П ₁ , П ₄ , П ₅ , П ₃	6	П ₅ , П ₇ , П ₆ , П ₁ , П ₄ , П ₃ , П ₂
7	П ₃ , П ₁ , П ₅ , П ₄ , П ₇ , П ₂ , П ₆	7	П ₂ , П ₁ , П ₇ , П ₄ , П ₅ , П ₃ , П ₆	7	П ₇ , П ₁ , П ₆ , П ₄ , П ₃ , П ₂ , П ₅
8	П ₁ , П ₅ , П ₄ , П ₇ , П ₂ , П ₆ , П ₃	8	П ₁ , П ₇ , П ₄ , П ₅ , П ₃ , П ₆ , П ₂	8	П ₁ , П ₆ , П ₄ , П ₃ , П ₂ , П ₅ , П ₇

Хід роботи

Код програми

```
.686
.model flat ,stdcall
.stack

.DATA

text2      BYTE 105 DUP (?)
text3      BYTE 75  DUP (?)

field1     BYTE 22  DUP (?)
field2     BYTE 22  DUP (?)
field3     BYTE 22  DUP (?)
field4     BYTE 22  DUP (?)
field5     BYTE 22  DUP (?)
field6     BYTE 22  DUP (?)
field7     BYTE 22  DUP (?)

field1Len  DWORD ?
field2Len  DWORD ?
field3Len  DWORD ?
field4Len  DWORD ?
field5Len  DWORD ?
field6Len  DWORD ?
field7Len  DWORD ?

spacesTotal  DWORD 0
wordStart    DWORD 0

.CODE
spacesLen PROC start,DWORD
mov EDI, OFFSET text
add EDI, start
mov ESI, EDI
mov AL, ' '
mov ECX, 75
```

```

repe scasb
dec EDI
sub EDI, ESI
mov EAX, EDI
mov EBX, EAX
add EBX, start
ret
spacesLen ENDP

fieldLen PROC start :DWORD
mov EAX, start
mov wordStart, EAX
mov EDI, OFFSET text
add EDI, start
mov ESI, EDI
mov AL, ' '
mov ECX, 75
repne scasb
dec EDI
sub EDI, ESI
mov EAX, EDI
mov EBX, EAX
add EBX, start
ret
ret
fieldLen ENDP

main PROC
invoke fieldLen, 0
mov field1Len, EAX

mov ECX, field1len
mov ESI, OFFSET text
mov EDI, OFFSET field1
rep movsb

invoke spacesLen, EBX
add spacesTotal, EAX

invoke fieldLen, EBX
mov field2Len, EAX

mov ECX, field2len
mov ESI, OFFSET text
add ESI, wordStart
mov EDI, OFFSET field2
rep movsb

invoke spacesLen, EBX
add spacesTotal, EAX

invoke fieldLen, EBX
mov field3Len, EAX

mov ECX, field3len
mov ESI, OFFSET text
add ESI, wordStart
mov EDI, OFFSET field3
rep movsb

invoke spacesLen, EBX
add spacesTotal, EAX

```

```

invoke fieldLen, EBX
mov field4Len, EAX

mov ECX, field4len
mov ESI, OFFSET text
add ESI, wordStart
mov EDI, OFFSET field4
rep movsb

invoke spacesLen, EBX
add spacesTotal, EAX

invoke fieldLen, EBX
mov field5Len, EAX

mov ECX, field5len
mov ESI, OFFSET text
add ESI, wordStart
mov EDI, OFFSET field5
rep movsb

invoke spacesLen, EBX
add spacesTotal, EAX

invoke fieldLen, EBX
mov field6Len, EAX

mov ECX, field6len
mov ESI, OFFSET text
add ESI, wordStart
mov EDI, OFFSET field6
rep movsb

invoke spacesLen, EBX
add spacesTotal, EAX

invoke fieldLen, EBX
mov field7Len, EAX

mov ESI, OFFSET text
add ESI, wordStart
mov EDI, OFFSET field7
mov ECX, field7len
rep movsb

invoke spacesLen, EBX
add spacesTotal, EAX

mov AL, ' '
mov EDI, OFFSET text2

mov ECX, 1
rep stosb
mov ESI, OFFSET field1
mov ECX, field1len
rep movsb

mov ECX, 7
rep stosb
mov ESI, OFFSET field7
mov ECX, field7len
rep movsb

```

```

mov ECX, 4
rep stosb
mov ESI, OFFSET field4
mov ECX, field4len
rep movsb

mov ECX, 5
rep stosb
mov ESI, OFFSET field5
mov ECX, field5len
rep movsb

mov ECX, 3
rep stosb
mov ESI, OFFSET field3
mov ECX, field3len
rep movsb

mov ECX, 6
rep stosb
mov ESI, OFFSET field6
mov ECX, field6len
rep movsb

mov ECX, 2
rep stosb
mov ESI, OFFSET field2
mov ECX, field2len
rep movsb
ret
main ENDP

END main

```

Висновки

Під час виконання лабораторної роботи я освоїв команди асемблера для роботи з рядками символів; опанував функції Win32 для роботи з файлами; розвинув навички складання програми для опрацювання рядка символів та програми для створення, записування і читання текстового файлу; відтранлював і виконати в режимі відлагодження програми, складені відповідно до свого індивідуального завдання.