

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ
УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ
ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»

Лабораторна робота №7

з дисципліни: «Архітектура комп'ютера»

ПІДПРОГРАМИ АРХІТЕКТУРИ IA-32 (X86) У REAL ADDRESS MODE

СПЕЦІАЛЬНОСТІ

121 – Інженерія програмного забезпечення

Виконав: Степаненко А.Ю., Салимоненко В.О., Дурдинець О.Т.

Підпис: _____

Кількість балів: _____ Оцінка _____

Група: IT-01

Викладач: Бердник Ю. М.

Підпис: _____

Київ 2020

Мета лабораторної роботи:

Мета лабораторної роботи полягає у створенні підпрограми архітектури IA-32(X86) у Real Address mode

Завдання для ЛР 7

Лабораторна робота передбачає проведення експерименту, що дозволяє розкрити механізм виклику підпрограм і описати використання команд call, ret. Визначення стану регістрів при виконанні підпрограм, стану пам'яті, стеку. Подальша збірка фрагментів лістингу програми і поточного стану МПС з метою розкриття механізму виклику підпрограм і використання пам'яті. Для цього необхідно написати програму, що реалізує текстовий інтерфейс і підпрограми. Завдання.

У нас був варіант 8:

Таблиця 2.


Сполучення букв для розробки інтерфейсу користувача

Функції	Номер варіанту																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Літера для виклику функції обчислення виразу	q	r	y	i	a	G	l	c	n	.	2	5	8	e	t	u	i	o
Літера для виклику функції включення звуку (тривалість звучання с)	W	T	U	O	S	H	Z	V	M	/	3	6	9	D	H	J	K	L
	1	2	3	4	5	4	3	2	1	2	4	7	8	9	1	3	4	5
Літера для виходу з програми	e	y	i	p	d	k	x	b	,	l	4	7	0	c	n	m	,	.
Літера для пошуку найбільшого значення (парний варіант)		a		b		c		d		f		g		h		r		o
Літера для пошуку найменшого значення (непарний варіант)	q		w		e		t		r		y		u		o		p	

1. $((a_1+a_2)*a_3/a_4+a_5)$	$a_1=-7, a_2=3, a_3=2, a_4=4, a_5=1$
2. $((a_1+a_2)*a_3/a_4+a_5)$	$a_1=-7, a_2=3, a_3=2, a_4=4, a_5=2$
3. $((a_1+a_2)*a_3/a_4+a_5)$	$a_1=-7, a_2=3, a_3=2, a_4=4, a_5=3$
4. $((a_1-a_2)*a_3*a_4+a_5)$	$a_1=-1, a_2=1, a_3=2, a_4=2, a_5=3$
5. $((a_1-a_2)*a_3*a_4+a_5)$	$a_1=-1, a_2=2, a_3=1, a_4=2, a_5=3$
6. $((a_1-a_2)*a_3*a_4+a_5)$	$a_1=-1, a_2=1, a_3=1, a_4=2, a_5=3$
7. $((a_1-a_2)+a_3)/a_4*a_5)$	$a_1=-2, a_2=3, a_3=1, a_4=2, a_5=3$
8. $((a_1-a_2)+a_3)/a_4*a_5)$	$a_1=-2, a_2=3, a_3=1, a_4=2, a_5=3$
9. $((a_1-a_2)+a_3)/a_4*a_5)$	$a_1=-2, a_2=3, a_3=1, a_4=2, a_5=3$
10. $(a_1-a_2*a_3/a_4+a_5)$	$a_1=-6, a_2=3, a_3=2, a_4=2, a_5=1$
11. $(a_1-a_2*a_3/a_4+a_5)$	$a_1=-6, a_2=3, a_3=2, a_4=2, a_5=1$
12. $(a_1-a_2*a_3/a_4+a_5)$	$a_1=-6, a_2=3, a_3=2, a_4=2, a_5=1$
13. $((a_1-a_2)/a_3 - a_4)*a_5)$	$a_1=-3, a_2=3, a_3=2, a_4=1, a_5=2$
14. $((a_1-a_2)/a_3 - a_4)*a_5)$	$a_1=-3, a_2=3, a_3=2, a_4=1, a_5=2$
15. $((a_1-a_2)/a_3 - a_4)*a_5)$	$a_1=-3, a_2=3, a_3=2, a_4=1, a_5=2$
16. $((a_1+a_2)/a_3 + a_4) - a_5)$	$a_1=-8, a_2=4, a_3=2, a_4=1, a_5=1$

Реалізація:

Для спрощення був використаний .bat файл який дозволяє запускати нашу програму

```
C: > TASM >  vatnik.bat  
1  @echo off  
2  
3  set arg=%lab7  
4  tasm %arg%.asm  
5  tlink %arg%.obj  
6  %arg%.exe
```

Код для програми:

C: > TASM > ~~asm~~ lab7.asm

```
1 ;Програма під час початку роботи виводить повідомлення.
2 ;В залежності від обраного повідомлення вона :
3 ;1. Викликає функцію для видачі звуку.
4 ;2. Викликає функцію для обчислення результату виразу
5 ;3. Забезпечує вихід з програми
6 ; Під час реалізації функцій
7 ; Програма читає зі стандартного вводу (клавіатури) строку.
8 ; Розроблено на кафедрі АУТС 3.10.2011 року.
9 ;-----
10 IDEAL
11 MODEL small
12 STACK 512
13
14 DATASEG
15 Array dw 1253h, 5739h ,1537h ,8674h, 3752h, 1964h ,2145h ,6831h, 2363h, 2988h ,5482h ,8762h, 1321h, 4895h ,2641h ,0913h ;1
16 dw 5437h, 5627h ,2894h ,3613h, 7865h, 3178h ,8937h ,2164h, 8970h, 5210h ,2615h ,1542h, 7847h, 8263h ,5895h ,4785h ;2
17 dw 4587h, 2361h ,1423h ,1485h, 1369h, 7855h ,1452h ,4685h, 1217h, 9868h ,5232h ,7874h, 5854h, 4578h ,5362h ,2585h ;3
18 dw 1454h, 1530h ,7454h ,8566h, 3226h, 4754h ,1563h ,2573h, 1984h, 8763h ,4129h ,6214h, 8742h, 8962h ,8753h ,8513h ;4
19 dw 2325h, 5784h ,1451h ,6986h, 5743h, 5842h ,8521h ,8969h, 3214h, 4789h ,5263h ,5478h, 5412h, 9534h ,7513h ,2196h ;5
20 dw 6587h, 6542h ,5236h ,5745h, 4523h, 9998h ,7812h ,9648h, 2684h, 3572h ,6255h ,3458h, 1452h, 1457h ,3256h ,8547h ;6
21 dw 1478h, 2486h ,1268h ,3589h, 2452h, 5896h ,6587h ,2359h, 1452h, 6556h ,2547h ,5221h, 2189h, 3278h ,9641h ,5438h ;7
22 dw 4785h, 5244h ,4787h ,5226h, 3247h, 1263h ,1593h ,6547h, 2581h, 3278h ,4258h ,9587h, 4257h, 4215h ,6521h ,9854h ;8
23 dw 2561h, 3274h ,3025h ,5846h, 5810h, 6309h ,8574h ,2653h, 1239h, 8956h ,8751h ,8563h, 8451h, 8712h ,2548h ,3024h
24 dw 2781h, 8756h ,8965h ,5412h, 7845h, 3264h ,9856h ,2187h, 3201h, 4858h ,9526h ,6325h, 3478h, 3218h ,2056h ,3057h
25 dw 0478h, 3205h ,9853h ,2103h, 2076h, 6254h ,7846h ,3558h, 2556h, 4851h ,3259h ,2563h, 1478h, 4512h ,3258h ,3259h
26 dw 1542h, 0845h ,0256h ,2482h, 6842h, 3621h ,3259h ,2412h, 3698h, 4874h ,5721h ,5624h, 3552h, 2541h ,2984h ,2563h;
27 dw 2565h, 4152h ,2585h ,3658h, 3256h, 1278h ,5326h ,3025h, 9856h, 2644h ,3265h ,2158h, 4856h, 5680h ,1475h ,0245h
28 dw 5045h, 1588h ,3256h ,1856h, 6325h, 3256h ,0542h ,4521h, 6235h, 2845h ,5562h ,1205h, 3482h, 4784h ,1542h ,1252h
29 dw 2586h, 6589h ,5523h ,4157h, 3584h, 2015h ,4785h ,2635h, 1451h, 4826h ,1243h ,1148h, 6825h, 4521h ,1536h ,5429h
30 dw 4852h, 5374h ,3945h ,1542h, 2752h, 7635h ,2356h ,2784h, 4587h, 5642h ,6319h ,5493h, 4862h, 2255h ,3278h ,5476h; 01FEh
31 Len dw 100h
32 string db 254 ;змінна для строки - string,
33 str_len db 0 ;
34 db 254 dup ('*') ; Буфер заповнюється '*' для кращого налаштування
35 ;---- Змінні для виводу системних команд
36 system_message_1 DB "Input command and press enter\ " , '$'
37 system_message_2 DB "Program end" , '$'
38 ;---- Змінні для виводу команд під час управління програмою
39 display_message_0 DB "-----programm for lab 3 is running !!!-----", 13, 10, '$'
40 display_message_1 DB "c - for count", 13, 10, '$'
41 display_message_2 DB "V - for beep", 13, 10, '$'
42 display_message_3 DB "b - for exit", 13, 10, '$'
43 display_message_3_3 DB "d - for max ", 13, 10, '$'
44 display_message_4 DB "-----programm for lab is END !!! -----", 13, 10, '$'
45 display_message_5 DB "Press any key for beep -----", 13, 10, '$'
46 ;--- Змінні що використовувалися під час налаштування програми
47 message DB ?
48 test_message_1 DB "!!! count DISPLAY", 13, 10, '$'
49 ;----Константи для функції звуку
50 NUMBER_CYCLES EQU 2000
51 FREQUENCY EQU 600
52 PORT_B EQU 61H
53 COMMAND_REG EQU 43H ; Адреса командного регістру
54 CHANNEL_2 EQU 42H ; Адреса каналу 2
55 simbol db ?
56
57 value1 db -2
58 value2 db 3
59 value3 db 1
60 value4 db 2
61 value5 db 3
62 ;////////////////////////////////////#////////////////////////////////////
63 CODESEG
64
65 Start:
66 mov ax, @data ; ax <- @data
67 mov ds, ax ; ds <- ax
68 mov es, ax ; es <- ax
```

C: > TASM > **asm** lab7.asm

```
72  Main_cikle: ; Основний цикл програми для інтерфейсу користувача
73  ;-----
74  call display_foo_main
75  ;-----
76  mov ah, 0ah ; ah <- 0ah
77  mov dx, offset string ; пересилка в dx начала буфера
78  int 21h
79
80  xor ax, ax
81  mov bx, offset string ;пересилка в bx начала буфера для
82  ;реалізації адресації зі зміщенням
83  mov ax, [bx+1] ;занесення в ax чисельного значення
84  ;символу ASCII, що відповідає
85  ;знаку,
86  ;який введено з клавіатури
87  shr ax, 8 ;зсув в регістрі ax для виконання
88  ;cmp
89  cmp ax, 063h ; c ascii =63h ; Вибір відповідної функції
90  je Count ; На лекції 3!!!
91  cmp ax, 056h ; V ascii =56h
92  je Beep
93  cmp ax, 064h ; d ascii =64h
94  je sort
95  cmp ax, 062h ; b ascii =62h
96  je Exit
97  jmp Main_cikle
98  ;-----
99  Count:
100  mov dx, offset display_message_5 ; Закоментовані повідомлення у ході налаштування
101  call display_foo
102  call math
103  jmp Main_cikle
104  ;-----
105  Beep:
106  ; any foo for sound ; виклик функції звуку
107  mov dx, offset display_message_5
108  call display_foo
109  call zvukF1
110  jmp Main_cikle
111
112  ;-----
113  sort:
114  ; any foo for sound ; виклик функції звуку
115  mov dx, offset display_message_5
116  call display_foo
117  call BubbleSort
118  jmp Main_cikle
119  ;-----
120  ; Стандартний вихід з програми
121  Exit:
122  mov dx, offset display_message_4
123  call display_foo
124  mov ah, 04Ch ;
125  int 21h ;
126  ;-----SUB-1 display_foo_main-----
127  PROC display_foo_main
128  mov ah, 0 ;Функція відображає інтерфейс
129  ;користувача
130  mov al, 3
131  int 10h
132  mov dx, offset display_message_0
133  call display_foo
134  mov dx, offset display_message_1
135  call display_foo
136  mov dx, offset display_message_2
137  call display_foo
138  mov dx, offset display_message_3
139  call display_foo
```



```

C: > TASM > ASM lab7.asm
126 ;-----SUB-1 display_foo_main-----
127 PROC display_foo_main
128     mov ah, 0 ;Функція відображає інтерфейс
129     ;користувача
130     mov al, 3
131     int 10h
132     mov dx, offset display_message_0
133     call display_foo
134     mov dx, offset display_message_1
135     call display_foo
136     mov dx, offset display_message_2
137     call display_foo
138     mov dx, offset display_message_3
139     call display_foo
140     mov dx, offset display_message_3_3
141     call display_foo
142     mov dx, offset system_message_1
143     call display_foo
144     ret
145 ENDP display_foo_main
146 ;-----SUB-2 display_foo-----
147 PROC display_foo
148     mov ah, 9
149     int 21h
150     xor dx, dx
151     ret
152 ENDP display_foo
153 ;-----SUB-3 zvukF1-----
154 PROC zvukF1
155 lab2:
156
157     int 16h ; Зберігає отримане значення з клавіатури в змінній
158     mov [simvol], al ; simvol
159     cmp [simvol], 'e' ; Перевірка на відповідність і встановлення прапору ознаки 0
160     jz Exit ;
161     ; Перехід на Exit: у випадку відповідності
162
163     ;Встановлення частоти 440 гц
164     ;--- дозвіл каналу 2 встановлення порту В мікросхеми 8255
165     IN AL, PORT_B ;Читання
166     OR AL, 3 ;Встановлення двох молодших бітів
167     OUT PORT_B, AL ;пересилка байта в порт В мікросхеми 8255
168     ;--- встановлення регістрів порту вводу-виводу
169     MOV AL, 10110110B ;біти для каналу 2
170     OUT COMMAND_REG, AL ;байт в порт командний регістр
171     ;--- встановлення лічильника
172     MOV AX, 2705 ;лічильник = 1190000/440
173     OUT CHANNEL_2, AL ;відправка AL
174     MOV AL, AH ;відправка старшого байту в AL
175     OUT CHANNEL_2, AL ;відправка старшого байту
176
177     ;--- пауза 2 секунды
178     mov cx, 80
179 classic_loop:
180     mov bx, cx
181     mov ah, 86h
182     xor cx, cx
183     mov dx, 25000
184     int 15h
185     mov cx, bx
186     loop classic_loop
187
188     ;--- виключення звуку
189     IN AL, PORT_B ;отримуємо байт з порту В
190     AND AL, 1111100B ;скидання двох молодших бітів
191     OUT PORT_B, AL ;пересилка байтів в зворотному напрямку
192     ret
193 ENDP zvukF1

```

C: > TASM > ~~asm~~ lab7.asm

```
195 PROC math
196     mov ax, -2
197     mov bx, 3
198     sub ax, bx
199     mov bx, 1
200     add ax,bx
201     mov bl, 2
202     mov dx, 1h
203     idiv bl ;AL (приватне), Ah (залишок)
204     mov dx,ax
205     mov cl,3h
206     imul cl
207     mov dl,al
208     mov al,dh
209     imul cl
210     mov ah,al
211     mov al,dl
212
213     call output
214     ret
215 ENDP math
216
217 PROC output
218     ;ax - число
219     mov [ES:0234h], ' '
220     mov [ES:0233h], ' '
221     mov [ES:0232h], ' '
222     mov [ES:0231h], ' '
223     mov di,0230h ;es:di - адрес буфера приемника
224     push cx ;сохраняем регистры
225     push dx
226     push bx
227     mov bx,10 ;основание системы
228     XOR CX,CX ;в cx будет количество цифр в десятичном числе
229 m1:   XOR dx,dx
230     DIV bx ;делим число на степени 10
231     PUSH DX ;и сохраняем остаток от деления(коэффициенты при степенях) в стек
232     INC CX
233     TEST AX,AX
234     JNZ m1
235 m2:   POP AX
236     ADD AL,'0' ;преобразовываем число в ASCII символ
237     STOSb ;сохраняем в буфер
238     LOOP m2 ;все цифры
239     pop bx ;восстанавливаем регистры
240     POP dx
241     POP cx
242     mov [ES:022Dh], '$'
243     mov [ES:0235h], '$'
244     mov dx, 230h ;Пересилання адреси рядка символів message в регістр dx
245     mov ah,09h;Завантаження числа 09h до регістру ah
246     int 21h;Переривання
247
248     mov cx, 80
249 classic_loop2:
250     mov bx, cx
251     mov ah,86h
252     xor cx, cx
253     mov dx,25000
254     int 15h
255     mov cx, bx
256     loop classic_loop2
257
258     ret
259 ENDP output
260
261 PROC BubbleSort
262     lea si, Array
```

C: > TASM > ASM lab7.asm

```
261 PROC BubbleSort
262 lea si, Array
263 mov cx, Len
264     push    bx
265     push    cx
266     push    dx
267     push    si
268     push    di
269
270     mov     bx,     si
271     mov     dx,     cx
272     dec     dx
273     shl     dx,     1
274     dec     cx
275     mov     si,     0
276 ForI:
277     mov     di,     dx
278 ForJ:
279     mov     ax,     [bx+di-2]
280     cmp     ax,     [bx+di]
281     jbe     NextJ
282     xchg    ax,     [bx+di]
283     xchg    ax,     [bx+di-2]
284     xchg    ax,     [bx+di]
285 NextJ:
286     sub     di,     2
287     cmp     di,     si
288     ja      ForJ
289     add     si,     2
290     loop    ForI
291     mov     ax,     [DS:01FEh]
292     call    output
293     pop     di
294     pop     si
295     pop     dx
296     pop     cx
297     pop     bx
298     ret
299 ENDP BubbleSort
300
301 END Start
```

Ось меню нашої програми:

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: LAB7
-----programm for lab 3 is running !!!-----
c - for count
U - for beep
b - for exit
d - for max
Input command and press enter\ _
```

Результат після натискання клавіші c:

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: LAB7
-----programm for lab 3 is running !!!-----
c - for count
U - for beep
b - for exit
d - for max
Press any key for beep -----
250 _
```

Результат після натискання клавіші d:

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: LAB7
-----programm for lab 3 is running !!!-----
c - for count
U - for beep
b - for exit
d - for max
Press any key for beep -----
39320
```

Результат після натискання клавіші b:

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
-----programm for lab 3 is running !!!-----
c - for count
U - for beep
b - for exit
d - for max
-----programm for lab is END !!! -----
F:\>
```

Github репозиторій: <https://github.com/VadymT95/computer-architecture>

GitHub Анатолія: <https://github.com/Stepanenko-Anatolii>

GitHub Вадима: <https://github.com/VadymT95>

GitHub Олександра: <https://github.com/ParZZ1vaLl>

Висновки:

Під час виконання лабораторної роботи ми створили підпрограму архітектури IA-32(X86) у Real Address mode а також набули твердих навичок і знань технологічної основи розробки ПЗ на Асемблері, у ході якої застосовуються знання архітектури комп'ютерів.