

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №3
по дисциплине «Организация ЭВМ и системы»
Тема: Представление и обработка целых чисел. Организация
ветвящихся процессов

Студента гр. 1383

Панов М.Ю.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Изучение ветвлений на ассемблере и методов обработки целых чисел.

Задание.

(f3, f5, f3)

Разработать на языке Ассемблера программу, которая по заданным целочисленным значениям параметров a , b , i , k вычисляет: а) значения функций $i1 = f1(a,b,i)$ и $i2 = f2(a,b,i)$; б) значения результирующей функции $res = f3(i1,i2,k)$, где вид функций $f1$ и $f2$ определяется из табл. 2, а функции $f3$ - из табл.3 по цифрам шифра индивидуального задания ($n1,n2,n3$), приведенным в табл.4. Значения a , b , i , k являются исходными данными, которые должны выбираться студентом самостоятельно и задаваться в процессе исполнения программы в режиме отладки. При этом следует рассмотреть всевозможные комбинации параметров a , b и k , позволяющие проверить различные маршруты выполнения программы, а также различные знаки параметров a и b

Выполнение работы.

- 1) формула $7 - 4i$ не нуждается в оптимизации
- 2) формула $8 - 6i$ можно представить в виде $8 - 4i - 2i$
- 3) формула $20 - 4i$ не нуждается в оптимизации
- 4) формула $-(6i - 6)$ можно представить в виде $6 - 4i - 2i$

Так же для уменьшения количества кода заранее считается $2i$.

Тесты:

Номер	Входные данные	Результат
1	$a = 2$ $b = 1$ $k = 0$ $i = 2$	$i1 = -1$ $i2 = 12$ $res = 11$

2	a = 2 b = 1 k = 1 i = 2	i1 = -1 i2 = 12 res = -1
3	a = 1 b = 2 k = 1 i = 2	i1 = -4 i2 = -6 res = -6
4	a = 1 b = 1 k = 1 i = 2	i1 = -4 i2 = -6 res = -6
5	a = -1 b = 1 k = 0 i = 1	i1 = 7 i2 = 0 res = 7
6	a = 1 b = 1 k = 1 i = -1	i1 = 14 i2 = 12 res = 12

Выводы.

В ходе работы были изучены ветвления и обработка с целых чисел на языке ассемблер.

Исходный код

dosseg

.MODEL small

```
;      / 7 - 4*i , при a>b
;f3 = <
;      \ 8 -6*i , при a<=b
;
;      / 20 - 4*i , при a>b
;f5 = <
;      \ -(6*i - 6), при a<=b
;
;
;      / |i1 + i2|, при k=0
;f3 = <
;      \ min(i1,i2), при k/=0
```

a EQU 1

b EQU 1

k EQU 1

i EQU 2

.STACK

.DATA

f1 dw 0

f2 dw 0

res dw 0

.CODE

mov ax, i ; ax = i
shl ax, 1 ; ax = 2*i

mov cx, a
cmp cx, b
jle less

;ax отвечает за результат f3
;dx отвечает за результат f5

; a>b
neg ax ; -2i
shl ax, 1 ; ax = -4i
mov dx, ax ; dx = -4i
add ax, 7 ; ax = -4i +7
add dx, 20 ; dx = -4i +20
jmp finish1

; a<=b
less:
add ax, i ; ax = 3i
shl ax, 1 ; ax = 6i
neg ax ; ax = -6i
mov dx, ax ; dx = -6i
add ax, 8 ; ax = -6i + 8
add dx, 6 ; dx = -6i + 6

finish1:

mov [f1], ax **;i1**
mov [f2], dx **;i2**

mov cx, k
cmp cx, 0
je zer0

;k != 0

cmp ax, dx
jle finish2 **; if i1 < i2**
mov dx, ax **; i1 = i2**
jmp finish2

zer0:

add ax, dx **; i1 + i2**
cmp ax, 0
jge finish2 **; if i1 + i2 <= 0**
neg ax
jmp finish2

finish2:

mov [res], ax
END

ЛИСТИНГ:

Microsoft (R) Macro Assembler Version 5.10

11/6/22 05:12:43

Page 1-1

ï»¿dosseg

lb3.ASM(1): warning A4001: Extra characters on line

.MODEL small

; / 7 - 4*i , D;Ñ€D, a>b

;f3 = <

; \ 8 -6*i , D;Ñ€D, a<=b

;

; / 20 - 4*i , D;Ñ€D, a>b

;f5 = <

; \ -(6*i - 6), D;Ñ€D, a<=b

;

;

; / |i1 + i2|, D;Ñ€D, k=0

;f3 = <

; \ min(i1,i2), D;Ñ€D, k/=0

= 0001 a EQU 1

= 0001 b EQU 1

= 0001 k EQU 1

= 0001

i EQU 1

.STACK

.DATA

0000 0000 f1 dw 0
0002 0000 f2 dw 0
0004 0000 res dw 0

.CODE

0000 B8 0001 mov ax, i ; ax = i
0003 D1 E0 shl ax, 1 ; ax = 2*i

0005 B9 0001 mov cx, a
0008 83 F9 01 cmp cx, b
000B 7E 0F jle less

;ax $\neq \frac{3}{4}i, i^2 \neq i, i \neq i, i \neq i$

$\neq i, f \neq i, i \neq i, i \neq i$

, f3

;dx $\neq \frac{3}{4}i, i^2 \neq i, i \neq i, i \neq i$

$\neq i, f \neq i, i \neq i, i \neq i$

, f5

; a>b

000D F7 D8 neg ax ; -2i
000F D1 E0 shl ax, 1 ; ax = -4i

0011 8B D0	mov dx, ax	; dx = -4i
0013 05 0007	add ax, 7	; ax = -4i +7
0016 83 C2 14	add dx, 20	; dx = -4i +20
0019 EB 10 90	jmp finish1	

Microsoft (R) Macro Assembler Version 5.10

11/6/22 05:12:43

Page 1-2

; a<=b

001C	less:	
001C 05 0001	add ax, i	; ax = 3i
001F D1 E0	shl ax, 1	; ax = 6i
0021 F7 D8	neg ax	; ax = -6i
0023 8B D0	mov dx, ax	; dx = -6i
0025 05 0008	add ax, 8	; ax = -6i + 8
0028 83 C2 06	add dx, 6	; dx = -6i + 6

002B	finish1:	
	; mov [f1], ax	;i1
	;mov [f2], dx	;i2

002B B9 0001	mov cx, k
002E 83 F9 00	cmp cx, 0
0031 74 09	je zer0

;k != 0

0033	3B C2	cmp ax, dx	
0035	7E 11	jle finish2	; if i1 < i2
0037	8B D0	mov dx, ax	; i1 = i2
0039	EB 0D 90	jmp finish2	

003C		zer0:	
003C	03 C2	add ax, dx	; i1 + i2
003E	3D 0000	cmp ax, 0	
0041	7D 05	jge finish2	; if i1 + i2
			<= 0
0043	F7 D8	neg ax	
0045	EB 01 90	jmp finish2	

0048		finish2:	
0048	A3 0004 R	mov [res], ax	

END

Microsoft (R) Macro Assembler Version 5.10 11/6/22 05:12:43

Symbols-1

Segments and Groups:

N a m e	Length	Align	Combine	Class
DGROUP.....				GROUP
_DATA	0006	WORD	PUBLIC	'DATA'

```

STACK ..... 0400 PARA STACK 'STACK'
_TEXT ..... 004B WORD PUBLIC 'CODE'

```

Symbols:

N a m e	Type	Value	Attr
A	NUMBER	0001	
B	NUMBER	0001	
F1	L WORD	0000	_DATA
F2	L WORD	0002	_DATA
FINISH1	L NEAR	002B	_TEXT
FINISH2	L NEAR	0048	_TEXT
I	NUMBER	0001	
K	NUMBER	0001	
LESS	L NEAR	001C	_TEXT
RES	L WORD	0004	_DATA
ZER0	L NEAR	003C	_TEXT
@CODE	TEXT	_TEXT	
@CODESIZE	TEXT	0	
@CPU	TEXT	0101h	
@DATASIZE	TEXT	0	

@FILENAME TEXT lb3

@VERSION TEXT 510

91 Source Lines

91 Total Lines

28 Symbols

48020 + 459240 Bytes symbol space free

1 Warning Errors

0 Severe Errors