

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

ОТЧЕТ
по лабораторной работе №4
по дисциплине «Организация ЭВМ и систем»
Тема: Представление и обработка целых чисел. Организация ветвящихся
процессов

Студент гр. 9382

Дерюгин Д.А.

Преподаватель

Ефремов М.А

Санкт-Петербург

2020

Цель работы.

Изучить принцип представления целых чисел на языке ассамблера, а также научиться пользоваться процессами ветвления.

Задание.

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 .

Вариант 6(1.7.6)

$15-2*i$, при $a>b$

$f1 = <$

$3*i+4$, при $a\leq b$

$-(4*i-5)$, при $a>b$

$f2 = <$

$10-3*i$, при $a\leq b$

$|i1-i2|$, при $k<0$

$f3 = <$

$\max(7,|i2|)$, при $k\geq 0$

Выполнение работы.

- 1) Объявлены сегменты стека и данных с переменными.
- 2) Сравниваем a и b и выполняем код, который подходит под условие.

3) Вычисляем f_1 и f_2

4) Сравниваем k и вычисляем f_3

Тестирование

Результаты тестирования

Табл1.

Входные данные	Выходные данные
a: 1 b: 2 i: 3 k: 4	f1: 13 f2: 4 f3: 7
a: 2 b: 1 i: 3 k: -1	f1: 9 f2: -7 f3: 16
a: 1 b: 2 i: 3 k: -1	f1: 13 f2: 4 f3: 17
a: 2 b: 1 i: -5 k: 1	f1: 25 f2: 25 f3: 25

Вывод.

В ходе выполнения работы были изучены принципы обработки целых чисел и принципы организации ветвящихся процессов.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЕ КОД ПРОГРАММЫ

```
AStack SEGMENT STACK
```

```
DW 12 DUP(?)
```

```
AStack ENDS
```

```
DATA SEGMENT
```

```
a DW 5
```

```
b DW 6
```

```
i DW 7
```

```
k DW 9
```

```
i1 DW ?
```

```
i2 DW ?
```

```
res DW ?
```

```
DATA ENDS
```

```
CODE SEGMENT
```

```
ASSUME CS:CODE, DS:DATA, SS:AStack
```

```
Main PROC FAR
```

```
push ds
```

```
sub ax, ax
```

```
push ax
```

```
mov ax, DATA
```

```
sub ax, ax
```

```
mov ds, ax
```

```
sub ax, ax
```

```
mov ax, a
```

```
mov bx, b
```

```
mov cx, i
```

```
cmp ax, bx
```

```
ja agb
```

```
    ; a<=b
```

```
    add i1, cx; i1 = i
```

```
    add bx, cx; bx = i
```

```
    shl cx, 1; cx = i*2
```

```
    add bx, cx; i1 = i+ i*2
```

```
    add i1, bx; i1 = i+i*2
```

```
    neg bx; cx = -i*2 - i
```

```
    add i2, bx; i2 = -i-i*2
```

```
    add i2, 10; i2 = -i-i*2 + 10
```

```
    add i1, 4; i1 = i+i*2 + 4
```

```
    jmp checkk
```

```
; a > b
```

```
agb:
```

```
    shl cx, 1; cx = i*2
```

```
    neg cx; cx = -i*2
```

```
    add i1, cx; i1 = -i*2
```

```
    shl cx, 1; cx = -i*4
```

```
    add cx, 5; cx = -i*4 + 5
```

```
add i2, cx; i2 = -i*4+5
add i1, 15; i1 = -i*2 + 15
jmp checkk
```

```
checkk:
mov ax, k; ax = k
cmp ax, 0
jb klz
;k>=0
mov bx, i2; bx = i2
cmp bx, 0
jb lesszero
;i2>=0
cmp bx, 7
jb maxseven
mov res, bx
ret
```

```
klz:
;k<0
mov cx, i1; cx = i1
mov dx, i2; dx = i2
neg dx;; dx = -i2
add cx, dx; cx = i1-i2
cmp cx, 0
jb lesszero
; cx>=0
mov res, cx
ret
```

```
lesszero:
neg cx
```

```
maxseven:
mov bx, 7
```

```
Main ENDP
CODE ENDS
END Main
```