

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 систем»
Тема: Представление и обработка целых чисел. Организация
ветвящихся процессов
Вариант 8

Студент гр. 1383

Куликов М.Д.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Изучение представления и обработки целых чисел и организации ветвящихся процессов.

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 .

Выполнение работы.

В ходе выполнения работы были реализованы необходимые функции, зависящие от переменных a,b,k . Для вычисления значений переменных были использованы команды `add`, `sub`, `shl` (побитовый сдвиг влево). Для сравнения значений и последующего перехода по меткам в процедуре `Main` были использованы команды `cmp` (сравнение), `jg` (переход когда 1 операнд больше 2 операнда), `jl` (переход когда 1 операнд меньше 2 операнда).

После написания программы она была протранслирована, после чего были исправлены все критические ошибки и предупреждения.

Выводы.

В ходе работы было изучено представление и обработка целых чисел и написана программа, удовлетворяющая условиям лабораторной работы.

ПРИЛОЖЕНИЕ А
ИСХОДНЫЙ КОД ПРОГРАММЫ

a EQU 1

b EQU 2

i EQU 3

k EQU 4

AStack SEGMENT STACK

DW 12 DUP(?)

AStack ENDS

DATA SEGMENT

i1 dw 0

i2 dw 0

res dw 0

DATA ENDS

; / - $(4*i+3)$, при $a>b$

; (f2)f1 = <

; \ $6*i - 10$, при $a \leq b$

;

; / $7 - 4*i$, при $a>b$

; (f3)f2 = <

; \ $8 - 6*i$, при $a \leq b$

;

; / $|i1| - |i2|$, при $k<0$

; (f8)f3 = <

; \ $\max(4, |i2|-3)$, при $k \geq 0$

CODE SEGMENT

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

Main PROC FAR

push DS

sub AX,AX

push AX

mov AX,DATA

mov DS,AX

mov ax,i

shl ax,1

mov i1,ax

add ax,i

mov i2,ax

mov ax,a

cmp ax,b

jg if_a_greater

shl i2,1

mov dx,i2

mov i1,dx

sub i1,10

neg i2

add i2,8

jmp func_3

if_a_greater:

shl i1,1

add i1,3

neg i1

```
mov ax,i1
mov i2,ax
add i2,10
```

```
func_3:
    cmp i1,0
    jg i1_after_neg
    neg i1
```

```
i1_after_neg:
    cmp i2,0
    jg k_comparing
    neg i2
```

```
k_comparing:
    mov dx,k ;переношу в dx так как k нельзя использовать в cmp с нулем
    cmp dx,0
    jl k_less
    sub i2,3
    cmp i2,4
    jl i2_less
    mov ax,i2
    mov res,ax
    ret
```

```
i2_less:
    mov res,4
    ret
```

```
k_less:
    mov ax,i1
    sub ax,i2
    mov res,ax
```

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