**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.**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 1303 |  | Хабибуллина А.М. |
| Преподаватель |  | Ефремов М.А. |

Санкт-Петербург

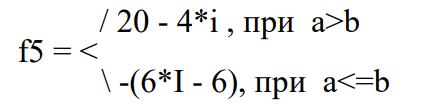
2022

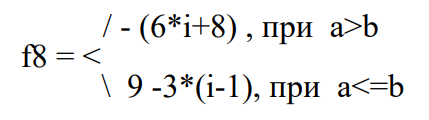
## **Цель работы.**

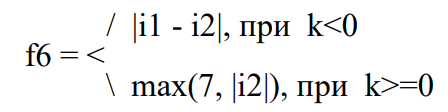
* 1. Разработать на языке Ассемблера программу, которая по заданным целочисленным значениям параметров вычисляет значения функций.
  2. Задание.
  3. Разработать на языке Ассемблера программу, которая по заданным целочисленным значениям параметров a, b, i, k вычисляет:
  4. а) значения функций i1 = f1(a,b,i) и i2 = f2(a,b,i);
  5. b) 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.





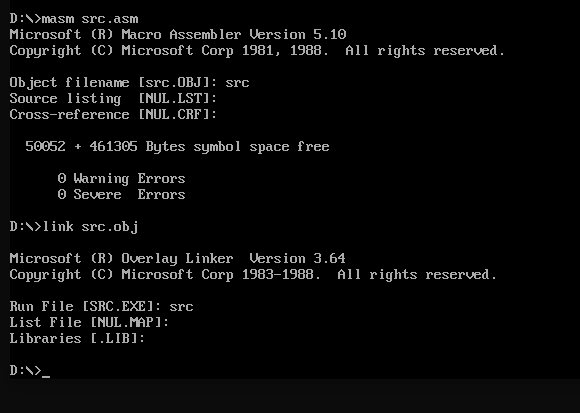


Выполнение работы

1. Получен вариант набора функций, которые необходимо реализовать, приведенного в каталоге задания.
2. Так как запрещено использование процедур в программе, функции были реализованы при помощи фрагментов кода, размеченных метками, с безусловными переходами на них. Функции содержат ветвление. Их поведение зависит от состояния переменных a, b.

3. Программа протранслирована без ошибок.

Трансляция программы:



* 1. Выводы

В ходе выполнения лабораторной работы были получены навыки разработки программы с заданными целочисленными значениями на языке программирования Ассемблер.

1. Приложение А  
   Исходный код программы

Название файла: src.asm

AStack SEGMENT STACK

DW 2 DUP(?)

AStack ENDS

DATA SEGMENT

a DW ?

b DW ?

i DW ?

k DW ?

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

mov DS,ax

; f1: if a > b: i1 = 20 - 4 \* i else: i1 = 6 - 6 \* i

; f2: if a > b: i2 = -6 \* i - 8 else: i2 = 12 - 3 \* i

mov ax, i

shl ax, 1

add ax, i ; ax = 3 \* i

mov bx, ax

shl bx, 1 ; bx = 6 \* i

mov cx, a

cmp cx, b

jg Go1

Go2: ; a <= b

; f1

mov i1,6

sub i1, bx ;i1 = 6 - 6 \* i

; f2

mov i2,12

sub i2,ax ;i2 = 12 - 3 \* i

jmp f3

Go1: ; a > b

; f1

mov i1,20

add ax, i

sub i1, ax ;i1 = 20 - 4 \* i

; f2

mov i2, -8

sub i2,bx ;i2 = -6 \* i - 8

f3:

mov ax,i1

mov bx,i2

cmp bx,0

jge cmp\_k

pos:

neg bx

mov res, ax

cmp\_k:

mov cx, k

cmp cx, 0

jl k2 ; k < 0

k1: ; k >= 0 : res = max(7, |i2|)

cmp bx, 7

jg final

mov res, 7

jmp final

k2: ; k < 0 : res = |i1 - i2|

sub res, bx

cmp res, 0

jge final

neg res

final:

ret

Main ENDP

CODE ENDS

END Main