**Лабораторная работа №1**

Цель работы: научится производить арифметические операции в регистрах и в стеке, научится работать с операциями CMP и JMP

**Задание 1**

1. Вариант 28. 16 бит, дополнительный код. Выражение: a/b- c; Значения: [13, 18, 3], [15, -12, -18] ; Результат:1)-3(ост. 13) 2)17(ост. 3)
2. Присвоим исходные данные регистрам AX, BX, CX. Целочисленный результат оставим в регистре AX, а остаток от деления в DX
3. Запишем программу для двух наборов значений

%include "io.inc"

;a/b-c

;13, 18, 3

section .text

global CMAIN

CMAIN:

XOR DX, DX ;обнуляем регистр DX, так как старшие биты делимого берутся из него

;записываем числа в регистры

MOV AX, 0x000D

MOV BX, 0x0012

MOV CX, 0x0003

IDIV BX ;13/18=0(ост.13)

SUB AX, CX ;0-3=-3

PRINT\_DEC 2, AX ;выводим целую часть результата

NEWLINE

PRINT\_DEC 2, DX ;выводим остаток от деления

NEWLINE

;15, -12, -18

XOR DX, DX ;обнуляем регистр DX, так как старшие биты делимого берутся из него

;записываем числа в регистры

MOV AX, 0x000F

MOV BX, 0xFFF4

MOV CX, 0xFFEE

IDIV BX ;15/(-12)=-1(ост.3)

SUB AX, CX ;-1-(-18)=18-1=17

PRINT\_DEC 2, AX ;выводим целую часть результата

NEWLINE

PRINT\_DEC 2, DX ;выводим остаток от деления

RET

На выходе программы получаем ожидаемые значения

**Задание 2**

1. Вариант 28. 16 бит, дополнительный код. Выражение: a/b- c; Значения: [13, 18, 3], [15, -12, -18] ; Результат:1)-2,2(7); 2)16,75
2. Присвоим исходные данные переменным a, b, c, в виде чисел с плавающей точкой типа IEEE 754. Будем работать с этими переменными в стеке. Ответ запишем в переменную r, которой заранее присвоим значение 0
3. Запишем программы для двух наборов значений

%include "io.inc"

;13, 18, 3

; ответ: -2,2(7)

section .text

global CMAIN

CMAIN:

;заносим переменные в стек

FLD DWORD [a]

FLD DWORD [b]

FDIV ;a/b(13/18=0,7(2))

FLD DWORD [c]

FSUB ;a/b-c(0,7(2)-3=-2,2(7))

FST DWORD [r] ;результат записываем в переменную r

ret

section .data

a: DD 0x41500000

b: DD 0x41900000

c: DD 0x40400000

r: DD 0x00000000

И для второго набора значений:

%include "io.inc"

;15, -12, -18

;ответ:16,75

section .text

global CMAIN

CMAIN:

mov ebp, esp; for correct debugging

;заносим переменные в стек

FLD DWORD [a]

FLD DWORD [b]

FDIV ;a/b(15/(-12)=-1,25)

FLD DWORD [c]

FSUB ;a/b-c(-1,25-(-18)=16,75)

FST DWORD [r] ;результат записываем в переменную r

ret

section .data

a: DD 0x41700000

b: DD 0xC1400000

c: DD 0xC1900000

r: DD 0x00000000

В переменной r в обоих случаях будут верные значения

**Задание 3**

1. Вариант 8: Задание: Вычислить сумму только положительных чисел из трех заданных
2. Рассмотрим все числа по очереди и сравним их с нулем, если они меньше или равны нулю, то игнорируем их, если больше нуля, то складываем с свободным регистром
3. Запишем программы для двух наборов значений

%include "io.inc"

;вычислить сумму только положительных чисел из трех заданных

;13, 18, 3

; ответ: 34

section .text

global CMAIN

CMAIN:

;записываем числа в регистры

MOV AX, 0x000D

MOV BX, 0x0012

MOV CX, 0x0003

XOR DX, DX ;обнуляем регистр чтобы дальше считать в нем сумму положительных чисел

CMP AX, 0x0000 ;сравниваем число с нулем

JNG l1 ;если оно меньше или равно нулю, то игнорируем его

ADD DX, AX ;если оно больше нуля, то добавляем его к общей сумме

l1:

;аналогично с двумя другими числами

CMP BX, 0x0000

JNG l2

ADD DX, BX

l2:

CMP CX, 0x0000

JNG l3

ADD DX, CX

l3:

PRINT\_DEC 2, DX ;выводим результат

RET

И для второго набора значений:

%include "io.inc"

;вычислить сумму только положительных чисел из трех заданных

;15, -12, -18

;ответ: 15

section .text

global CMAIN

CMAIN:

;записываем числа в регистры

MOV AX, 0x000F

MOV BX, 0xFFF4

MOV CX, 0xFFEE

XOR DX, DX ;обнуляем регистр чтобы дальше считать в нем сумму положительных чисел

CMP AX, 0x0000 ;сравниваем число с нулем

JNG l1 ;если оно меньше или равно нулю, то игнорируем его

ADD DX, AX ;если оно больше нуля, то добавляем его к общей сумме

l1:

;аналогично с двумя другими числами

CMP BX, 0x0000

JNG l2

ADD DX, BX

l2:

CMP CX, 0x0000

JNG l3

ADD DX, CX

l3:

PRINT\_DEC 2, DX ;выводим результат

RET

На выходе обеих программ получаем верные результаты

Вывод: я научился рабоать с регистрами и стеком, выполнять простые арифметические операции