Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Отчет

Лабораторная работа №3

Выполнил студент гр. 150505: Горбачевский К.В

Проверил: Туровец Н.О.

Минск 2022

**Тема работы**: Целочисленные арифметические операции. Обработка массивов числовых данных.

**Цель работы**: Ознакомиться с арифметическими операциями над целочисленными данными, обработкой массивов чисел, ознакомиться с правилами оформления ассемблерных процедур.

**Вариант лабораторной работы:** Выполнить набор логических побитовых операций над двумя целыми числами, представленными в 10-ной системе счисления.

**Код программы**

data segment

error db "The entered number is not decimal!$"

overflowError db "Overflow! Enter numbers less than 65535$"

enterNumber db "Enter number: $"

orAnswer db "Answer OR: $"

andAnswer db "Answer AND: $"

xorAnswer db "Answer XOR: $"

frst db 50,?,50 dup ('$')

scnd db 50,?,50 dup ('$')

data ends

code segment

printString:

mov ah, 09h

int 21h

ret

overflow:

call newLine

mov dx, offset overflowError

call printString

jmp start

plsEnterNumberMessage:

mov ah, 9h

mov dx, offset enterNumber

int 21h

ret

enterNumberOperation:

mov ah,0aH ; read string from keybord

int 21h

ret

newLine:

mov ah, 02h

mov dl, 0ah

int 21h

mov dl, 0dh

int 21h

ret

notDecimal:

call newLine

mov ah, 09h

mov dl, offset error

int 21h

call newLine

jmp start

ret

minus:

mov dx, 0FFFFh

inc si

dec cx

jmp go

ret

stringToNumber:

mov bx, 0h

mov bl, [si+1]

mov cx, bx

mov di, 10

cmp [si+2], '-'

je minus

go:

xor ax,ax ;AX = 0

xor bx,bx ;BX = 0

cycle:

inc si

mov bl, [si+1] ; next symbol

cmp bl, '0' ; if bl < '0' not decimal

jl notDecimal

cmp bl, '9' ; if bl > '9' not decimal

jg notDecimal

sub bl, '0' ; bite bl from symbol code to number

push dx ; save dx (FFFF)

mul di ; ax = al\*di(10)

pop dx

jc overflow ; if CF=1

add ax,bx

jc overflow ; if CF=1

loop cycle

ret

op:

mov di, 10 ; for div 10

mov si, 0

makeAnswer:

cmp bp, 0

je printAnswer

xor dx,dx

mov ax, bp ; ax = number

div di ; dx = ax % 10, ax = ax/10

mov bp,ax ; bp = number/10

;add dx, '0' ; dx - symbol

push dx ; push symbol to stack

inc si

loop makeAnswer

ret

printAnswer:

cmp si, 0 ; exit check

je exit

mov cx, si

printSymbols:

pop dx ; pop last symbol from stack

mov ah, 02h

int 21h

loop printSymbols

jmp viaExit

exit:

mov dx, '0'

mov ah, 02h

int 21h

viaExit:

ret

OROR:

mov si, bx

mov di, dx

or si, di

mov bp, si

ret

ANDAND:

mov si, bx

mov di, dx

and si, di

mov bp, si

ret

XORXOR:

mov si, bx

mov di, dx

xor si, di

mov bp, si

ret

start:

mov ax, data

mov ds, ax

call plsEnterNumberMessage

mov dx, offset frst

call enterNumberOperation

mov si, offset frst

call stringToNumber

cmp dx, 0FFFFh

je setMinuss

jne notMinuss

setMinuss:

neg ax ; ax = -ax

push ax ; push number

mov dx, 0 ; dx = 0

jmp nextt

notMinuss:

push ax

nextt:

call newLine

call plsEnterNumberMessage

mov dx, offset scnd

call enterNumberOperation

mov si, offset scnd

call stringToNumber

cmp dx, 0FFFFh

je setMinus

jne notMinus

setMinus:

neg ax

push ax

mov dx, 0

jmp next

notMinus:

push ax

next:

call newLine

pop si ; si - second number

pop di ; di - first number

mov dx, offset orAnswer

call printString

mov es, si ; save numbers

mov sp, di ;

mov bx, si

mov dx, di

call OROR

call op

call newLine

mov dx, offset andAnswer

call printString

mov bx, es ; take numbers

mov dx, sp

call ANDAND

call op

call newLine

mov dx, offset xorAnswer

call printString

mov bx, es

mov dx, sp

call XORXOR

call op

int 20h

code ends

end start

**Вывод в консоли после компиляции**

