Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Конструирование программ

Отчёт

по лабораторной работе №7

на тему:

Целочисленные арифметические операции. Обработка массивов

числовых данных.

Студент Богданов А.С.

Принимающий Туровец Н.О.

МИНСК 2022

1. Вариант 3

Выполнить набор логических побитовых операций над двумя целыми

числами, представленными в 10-ной системе счисления.

2. Код программы

.8086

.model tiny

.code

org 100h

print macro out\_str ; макро вывода строки

mov ah,9

mov dx,offset out\_str

int 21h

endm

println macro ; макро вывода символа новой линии

print new\_line

endm

start:

; считываем 2 числа, заносим в стек

print enter\_first\_number

call InputInt

push ax

print enter\_second\_number

call InputInt

push ax

; выводим xor

print xor\_str

pop ax

pop bx

push bx

push ax

xor ax, bx

call OutInt

println

; выводим and

print and\_str

pop ax

pop bx

push ax

push bx

and ax, bx

call OutInt

println

; выводим or

print or\_str

pop ax

pop bx

or ax, bx

call OutInt

println

ret

InputInt proc

mov ah,0ah

xor di,di

mov dx,offset buff ; аддрес буфера

int 21h ; принимаем строку

println

; обрабатываем содержимое буфера

mov si,offset buff+2 ; берем аддрес начала строки

; длина строки > 0

mov cl, buff+1

cmp cl, 0

je er

xor ax,ax

mov bx,10 ; основание сc

lp:

xor cx, cx ; очищаем регистр cx

mov cl,[si] ; берем символ из буфера

cmp cl,0dh ; проверяем не последний ли он

jz endin

; если символ не последний, то проверяем его на правильность

cmp cl,'0' ; если введен неверный символ <0

jl er

cmp cl,'9' ; если введен неверный символ >9

ja er

sub cl,'0' ; делаем из символа число

mul bx ; умножаем на 10

add ax,cx ; прибавляем к остальным

inc si ; указатель на следующий символ

jmp lp ; повторяем

er: ; если была ошибка, то выводим сообщение об этом и выходим

mov dx, offset error

mov ah,09

int 21h

int 20h

; все символы из буфера обработаны число находится в ax

endin:

ret

InputInt endp

OutInt proc

xor cx, cx

mov bx, 10 ; основание сс. 10 для десятеричной и т.п.

oi2:

xor dx,dx

div bx

; Делим число на основание сс. В остатке получается последняя цифра.

; Сразу выводить её нельзя, поэтому сохраним её в стэке.

push dx

inc cx

; А с частным повторяем то же самое, отделяя от него очередную

; цифру справа, пока не останется ноль, что значит, что дальше

; слева только нули.

test ax, ax

jnz oi2

; Теперь приступим к выводу.

mov ah, 02h

oi3:

pop dx

; Извлекаем очередную цифру, переводим её в символ и выводим.

add dl, '0'

int 21h

; Повторим ровно столько раз, сколько цифр насчитали.

loop oi3

ret

OutInt endp

buff db 6,7 Dup(?)

error db "Incorrect input$"

enter\_first\_number: db "Enter first number:", 0Dh,0Ah, '$'

enter\_second\_number: db "Enter second number:", 0Dh,0Ah, '$'

xor\_str: db "Xor: $"

or\_str: db "Or: $"

and\_str: db "And: $"

new\_line: db 0Dh, 0Ah, '$'

end start

3. Результат выполнения программы

