Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Отчет

Лабораторная работа №5

Выполнил студент гр. 150505: Горбачевский К.В

Проверил: Туровец Н.О.

Минск 2022

**Тема работы:** Работа с файлами.

**Цель работы:** Ознакомиться с основными операциями обработки файлов, получить понятие о работе с параметрами командной строки.

**Вариант лабораторной работы:** Подсчитать число не пустых строк в файле.

**Код программы**

data segment

errorMessage db "Error!$"

startCounter db "Counted not-emty strings and print all text...$"

enterFilePath db "Enter path to file: $"

answer db "Count of not-empty strings is - $"

f\_name db 'text.txt', 0

;f\_name db 126,?,126 dup("$"), 0

buffer db 255,?, 255 dup('$')

data ends

code segment

OutInt proc

mov ax, bp ; ax == answer

xor cx, cx

mov bx, 0ah ; div number

delenie:

xor dx,dx

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

push dx ; save ax % 10

inc cx ; inc count of numbers

test ax, ax ; wile ax !== 0

jnz delenie ; call oi2

mov ah, 02h ; printing

printing:

pop dx

add dl, '0' ; number to symbol

int 21h

loop printing

ret

OutInt endp

close\_file:

mov ah,3Eh

int 21h

ret

error:

mov ah,09h

mov dl, offset errorMessage

int 21h

call close\_file

jmp exit

ret

newLine:

push ax

mov ah, 02h

mov dl, 0ah

int 21h

mov dl, 0dh

int 21h

pop ax

ret

isEmptyString proc

push si

inc si

cmp [buffer+si], 0AH

je isNext13

jne back

isNext13:

inc si

cmp [buffer+si], 0Dh

je back2:

jne back

back:

pop si

ret

back2:

pop si

dec bp

ret

isEmptyString endp

start:

mov ax, data

mov ds, ax

mov dx, offset 81h ; PSP block

mov al,2

mov dx, offset f\_name

mov ah,3dh

int 21h ; open file

jc error

mov bx,ax

mov di,01

mov bp, 1 ; not empty str counter

mov ah, 09h

mov dl, offset startCounter

int 21h

call newLine

call newLine

read\_data:

mov cx, 255 ; bites for read

mov dx, offset buffer ;

mov ah,3Fh ;

int 21h ; go read bites

jc error ; if error

mov cx,ax ; count of reading bites

mov si, 0 ; bite index

printBite:

cmp [buffer+si], '$' ; if str ended

je exit ; exit

push ax ; save ax

mov ah, 02h

mov dl, [buffer+si]

int 21h

pop ax ; get ax

cmp dl, 0Dh

je incCounter

jne nextSymbol

incCounter:

call isEmptyString ; is next str empty?

inc bp

nextSymbol: inc si

loop printBite

jmp read\_data

exit:

call newLine

call newLine

mov ah, 09h

mov dl, offset answer

int 21h

call OutInt

int 20h

code ends

end start

**Вывод в консоли после компиляции**

