Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра Информатики

Дисциплина «Конструирование программ»

**ОТЧЕТ**

к лабораторной работе №6

на тему:

**«**ЗАГРУЗКА И ВЫПОЛНЕНИЕ ПРОГРАММ. РАБОТА С ПАМЯТЬЮ**»**

БГУИР 6-05-0612-02 85

|  |
| --- |
| Выполнил студент группы 353503  МАРТИНОВИЧ Андрей Александрович |
|  |
| (дата, подпись студента) |
| Проверил ассистент каф. Информатики  РОМАНЮК Максим Валерьевич |
|  |
| (дата, подпись преподавателя) |

Минск 2024

# 1 Индивидуальное задание

**Задание 1. Вариант 5.** Написать программу, последовательно запускающую программы, которые расположены в заданном каталоге.

# 2 Выполнение работы

Листинг кода 1 – Текст программы задания 1

.model small

.data

msg1 db 13, 10, "Trying execute program 'test.exe'.$"

msg2 db 13, 10, "Program start error.$"

msg3 db 13, 10, "First program has been close. Press any key.$"

program db "TEST.exe", 0

fileParameters db 128 dup("$")

bufferFile db 1

counter db 1

buffer db 128 dup("$")

parameters db 128 dup("$")

endl db 10, 13, "$"

notFoundFile db 10, 13, "File not found!$"

errorNumberParameters db 13, 10, "Invalid number of parameters.$"

opened db 13, 10, "File has been opened.$"

closed db 13, 10, "File has been closed.$"

EPB dw 0

cmd\_off dw offset parameters

cmd\_seg dw ?

fcb1 dd ?

fcb2 dd ?

EPB\_len dw $ - EPB

dsize = $ - msg1

.stack 100h

.code

;-------------------

start:

call Main

;-------------------

Cout macro str

mov ax, 0900h

lea dx, str

int 21h

endm Cout

;-------------------

CopyParameters proc

mov si, 2

xor di, di

mov fileParameters[1], 0

ForCheck:

cmp buffer[si], 20h ;spase

je Next

cmp buffer[si], 09h

je Next

cmp buffer[si], 24h ;$

je ExitCopyParameters

cmp buffer[si], 0dh

je ExitCopyParameters

jmp ForCopyParameters

Next:

cmp di, 0

jne ExitCopyParameters

inc si

jmp ForCheck

ForCopyParameters:

mov dl, buffer[si]

mov fileParameters[di], dl

inc di

inc si

jmp ForCheck

ExitCopyParameters:

inc di

mov fileParameters[di], 0

ret

endp CopyParameters

;------------------------

OpenFile proc

mov ax, 3d00h

mov dx, offset fileParameters

int 21h

jc ErrorOpenFile

mov bx, ax

Cout opened

call ReadFile

call CloseFile

jmp OK

ErrorOpenFile:

Cout notFoundFile

jmp Exit

OK:

ret

endp OpenFile

;------------------------

ReadFile proc

mov si, 1

mov counter, 1

mov parameters[si], 20h

inc si

Reading:

mov ax, 3f00h

mov dx, offset bufferFile

mov cx, 1

int 21h

cmp ax, 0

je ReadingEnd

mov al, bufferFile

cmp al, 0dh

je NextArg

cmp al, 0ah

je Reading

mov parameters[si], al

inc si

inc counter

jmp Reading

NextArg:

mov parameters[si], ' '

inc si

inc counter

jmp Reading

ReadingEnd:

mov dl, counter

mov parameters[0], dl

ret

endp ReadFile

;------------------------

CloseFile proc

mov ax, 3e00h

int 21h

Cout closed

ret

endp CloseFile

;------------------------

CommandLine proc ;iz psp bloca zanosim parametri comandnoi stroci v buffer

xor cx, cx

xor di, di

mov si, 80h

commandLineInput:

mov al, es:[si] ;dlinna cpmandnoi stroci

inc si

cmp al, 0

je commandLineEnd

mov buffer[di], al

inc di

jmp commandLineInput

commandLineEnd:

ret

endp CommandLine

;-------------------------

StartAnotherProgram proc

mov ax, 4a00h ;izmenit razmer bloca pamiaty

mov bx, ((csize/16) + 17) + ((dsize/16) + 17) + 1 ;novii razmer

int 21h

Cout msg1

Cout msg3

;mov ax, 0100h

;int 21h

mov ax, @data

mov es, ax

mov ax, 4B00h ;zagruzka i vipolnenie programi

lea dx, program

lea bx, EPB ;adress bloca parametrov

int 21h

jb ErrorStartProgram

jmp Exit

ErrorStartProgram:

Cout msg2

ret

endp StartAnotherProgram

;-------------------------

Main proc

mov ax, @data

mov ds, ax

mov cmd\_seg, ax

call CommandLine

call CopyParameters

cmp fileParameters[2], 24h ;$

je Error

Cout fileParameters

Cout endl

call OpenFile

Cout endl

Cout endl

Cout parameters[1]

Cout endl

call StartAnotherProgram

jmp Exit

Error:

Cout errorNumberParameters

Exit:

mov ax, 4c00h

int 21h

endp Main

csize = $ - start

end start

# Вывод

В ходе выполнения лабораторной работы с программным обеспечением, предназначенным для сборки, отладки и запуска программ на языке ассемблера ознакомились с загрузкой и выполнением программ, рассмотрена работа с памятью.