

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 №7

на тему:

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

БГУИР 1-40 04 01

Выполнил студент группы 253505  
БЕКАРЕВ Станислав Сергеевич

---

(дата, подпись студента)

Проверил ассистент кафедры  
информатики  
РОМАНЮК Максим Валерьевич

---

(дата, подпись преподавателя)

Минск 2023

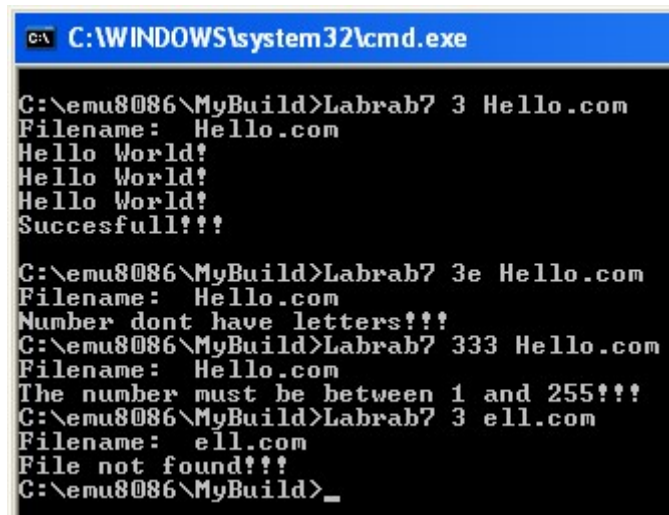
**Цель работы.** Ознакомиться с загрузкой и выполнением программ, рассмотреть работу с памятью.

В рамках выполнения лабораторной работы должно быть выполнено следующее задание:

Написать программу, запускающую другую программу N раз (N – число в диапазоне [1, 255]). Имя запускаемой программы передается в командной строке.

**Ход работы:** Для выполнения задания был написан программный код на языке Assembler, представленный в листинге №1.

Данный код запускает файл с введенным названием n раз. При выполнении работы требовалось также перевести 10-ное число из консоли записанное символами в 16-ное. Работа программы с обработкой ошибок ввода показана на рисунке 1.



```
C:\WINDOWS\system32\cmd.exe
C:\emu8086\MyBuild>Labrab7 3 Hello.com
Filename: Hello.com
Hello World!
Hello World!
Hello World!
Succesfull!!!

C:\emu8086\MyBuild>Labrab7 3e Hello.com
Filename: Hello.com
Number dont have letters!!!

C:\emu8086\MyBuild>Labrab7 333 Hello.com
Filename: Hello.com
The number must be between 1 and 255!!!

C:\emu8086\MyBuild>Labrab7 3 ell.com
Filename: ell.com
File not found!!!

C:\emu8086\MyBuild>_
```

Рисунок 1 – Работа программы.

#### Листнинг №1 – Исходный код задания

```
org 100h
start:
mov cx, 0h
mov cl, [80h]
mov si, 81h
mov di, offset amount

mov check, 0
lop1:
movsb
cmp [si], ' '
jne cont1
cmp check, 1
jne cont2
mov ah, 09h
mov dx, offset error_much_params
int 21h
```

```

mov ah, 4Ch
int 21h
cont2:
mov check, 1
mov di, offset filename
cont1:
loop lop1
inc di
mov [di], '$'
mov ah, 09h
mov dx, offset mess_filename
int 21h
mov dx, offset filename
int 21h
mov dx, offset enter
int 21h

mov sp,program_length + 100h + 200h

mov ah,4Ah
stack_shift = program_length + 100h + 200h
mov bx,stack_shift shr 4 + 1
int 21h

mov ax,cs
mov word ptr EPB+4,ax
mov word ptr EPB+8,ax
mov word ptr EPB+0Ch,ax

call convert_to_hex
mov cx, ax
cmp cx, 0
jne cont5
mov ah, 09h
mov dx, offset error_size_amount
int 21h
mov ah, 4Ch
int 21h
cont5:
lop:
mov ax,4B00h
mov dx,offset filename + 1
mov bx, offset EPB
int 21h
jc error
loop lop

mov ah, 09h
mov dx, offset mess_succ
int 21h

mov ah, 4Ch
int 21h

error:
cmp al, 02h
jne step1

```

```

mov ah, 09h
mov dx, offset error1
int 21h
mov ah, 4Ch
int 21h
step1:
cmp al, 05h
jne step2
mov ah, 09h
mov dx, offset error2
int 21h
mov ah, 4Ch
int 21h
step2:
cmp al, 08h
jne step3
mov ah, 09h
mov dx, offset error3
int 21h
mov ah, 4Ch
int 21h
step3:
cmp al, 0Ah
jne step4
mov ah, 09h
mov dx, offset error4
int 21h
mov ah, 4Ch
int 21h
step4:
cmp al, 0Bh
jne step5
mov ah, 09h
mov dx, offset error5
int 21h
mov ah, 4Ch
int 21h
step5:
mov ah, 09h
mov dx, offset last_error
int 21h
mov ah, 4Ch
int 21h

proc convert_to_hex
    mov ten, 0Ah
    mov ax, 0
    mov di, offset amount + 1
    lop2:
    cmp [di], 30h
    jl err
    cmp [di], 39h
    jg err
    imul ten
    mov bl, [di]
    sub bl, 30h
    add al, bl

```

```

        jc err2
        cmp ax, 00FFh
        jg err2
        inc di
        cmp [di], '$'
        jne lop2
        jmp endl
err:
        mov ah, 09h
        mov dx, offset error_write_amount
        int 21h
        mov ah, 4Ch
        int 21h
err2:
        mov ah, 09h
        mov dx, offset error_size_amount
        int 21h
        mov ah, 4Ch
        int 21h
endl:
        ret
endp convert_to_hex

EPB dw 0000
dw offset commandline,0
dw 005Ch,0,006Ch,0
commandline db 125
db " "
command_text db 124 dup (?)
program_length equ $-start
enter db 0Ah, 0Dh, '$'
filename db 20 DUP(0)
amount db 5 DUP('$')
mess_filename db "Filename: $"
mess_succ db "Succesfull!!!", 0Ah, 0Dh, '$'
error_much_params db "To much params!!!$"
error_size_amount db "The number must be between 1 and 255!!!$"
error_write_amount db "Number dont have letters!!!$"
error1 db "File not found!!!$"
error2 db "File acces denied!!!$"
error3 db "Not enough memory!!!$"
error4 db "Wrong weapon!!!$"
error5 db "Wrong format!!!$"
last_error db "Something wrong!!!$"
check db 0
ten dw 0Ah
end start

```

**Выводы:** Была написана программа, которая запускает введенную программу n раз. Были изучены основные способы загрузки и выполнения программ, рассмотрена работа с памятью.