

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

Снежко Максим Андреевич

(дата, подпись студента)

Проверил ассистент кафедры
информатики

Романюк Максим Валерьевич

(дата, подпись преподавателя)

Минск 2023

Цель работы: Задание 1. Вариант 1. Написать программу, запускающую другую программу N раз (N – число в диапазоне [1, 255]). Имя запускаемой программы задается константой.

Ход работы: На рисунке 1 представлена передача количества запуска в командной строке. На рисунке 2 представлен вывод и предложение ввода запускаемой программы.

Листинг 1 – Исходный код программы задания 1

```
.model tiny
.code
org 100h
begin:
jmp start
ClearConsole proc
    mov ah, 06h
    mov al, 0
    mov bh, 07h
    mov ch, 0
    mov cl, 0
    mov dh, 24
    mov dl, 79
    int 10h
    mov ah, 02h
    mov bh, 0
    mov dh, 0
    mov dl, 0
    int 10h
    ret
ClearConsole endp
getProgramParams proc
    push ax
    push bx
    push cx
    push dx
    mov error, 0
    xor cx, cx
    mov cl, es:[80h]
    cmp cl, 0
    je invalidCommandLine
    mov di, 82h
    lea si, commandText
getSymbols:
    mov al, es:[di]
    cmp al, 0Dh
    je parametersEnded
```

```

        mov [si], al
        inc di
        inc si
        jmp getSymbols
invalidCommandLine:
        mov error, 1
        jmp getParamsExit
parametersEnded:
        inc si
        mov byte ptr [si], 0
getParamsExit:
        pop dx
        pop cx
        pop bx
        pop ax
        ret
getProgramParams endp
atoi proc
        push bx
        push cx
        push dx
        push si
        xor bx, bx
        xor dx, dx
        mov error, 0
convert:
        xor ax, ax
        lodsb
        cmp al, 0
        je exit
        cmp al, 57
        ja exit
        cmp al, 48
        jb exit
        cmp al, 21h
        je exit
        cmp al, 72h
        je exit
        cmp al, '-'
        je invalidInput
        cmp al, '9'
        jnbe invalidInput
        cmp al, '0'
        jb invalidInput
        sub ax, '0'

```

```

        shl dx, 1
        add ax, dx
        shl dx, 2
        add dx, ax
        cmp dx, MAX_NUMBER
        jg invalidInput
        cmp dx, 0
        je invalidInput
    jmp convert
        jmp exit
invalidInput:
        mov error, 1
exit:
        mov ax, dx
        mov N_times, al
        pop si
        pop dx
        pop cx
        pop bx
        ret
atoi endp
start:
        call ClearConsole
        call getProgramParams
        cmp error, 1
        je invalidCommandLineArgs
        lea si, commandText
        call atoi
        cmp error, 1
        jne loadProgram
        mov ax, cs
        mov ds, ax
        mov ah, 09h
        lea dx, invalidNumberMessage
        int 21h
        jmp exitStart
loadProgram:
        mov ax, 4A00h
        mov bx, ((CodeSize / 16) + 17) + ((CodeSize / 16) +
17) + 1
        int 21h
        jc errResize
        xor cx, cx
        mov cl, N_times
runProgram:

```

```

    mov ax, @data
    mov es, ax
    mov ax, cs
    mov ds, ax
    mov ah, 4Bh
    mov al, 00h
    lea dx, programPath
    lea bx, EPB
    int 21h
    mov ax, cs
    mov ds, ax
    mov ah, 09h
    lea dx, newLine
    int 21h
    mov ax, cs
    mov ds, ax
    mov ah, 09h
    lea dx, repeat_mes
    int 21h
    mov ah, 0
    int 16h
    push cx
    call ClearConsole
    pop cx
    jc errorDuringLoadingProgram
    loop runProgram
    jmp exitStart
errorDuringLoadingProgram:
    mov ax, cs
    mov ds, ax
    mov ah, 09h
    lea dx, errorWhileExec
    int 21h
    jmp exitStart
invalidCommandLineArgs:
    mov ax, cs
    mov ds, ax
    mov ah, 09h
    lea dx, invalidCmdArgs
    int 21h
    jmp exitStart
errResize:
    mov ax, cs
    mov ds, ax
    mov ah, 09h

```

```

        lea dx, rsErr
        int 21h
        jmp exitStart
exitStart:
        mov ah, 4ch
        int 21h
repeat_mes db 'Press any key to go to next iteration',
'$'
N_times      db      0
newLine      db      10, 13, '$'
end_prograb db 'Progr end$'
invalidNumberMessage db      'Invalid input: number
should contain only digits in the following range [1,
255]', 10, 13, '$'
invalidCmdArgs      db      'Invalid command line
input. Try again.', 10, 13, '$'
errorWhileExec      db      'Error has happened
during loading/running program.', 10, 13, 'Check your
program path and try again', '$'
rsErr db "An error occurred while resizing", 10, 13,
'$'
error          db      0
programPath    db      "LAB5.EXE", 0
EPB          dw      0000
             dw      offset commandLine, 0
             dw      005Ch, 0, 006Ch, 0
commandLine    db      0
commandText    db      125 dup(0)
MAX_NUMBER     equ      255
DataSize = $ - EPB
CodeSize  =      $ - start
end begin

```



Рисунок 1 – передача количество раз в командную строку

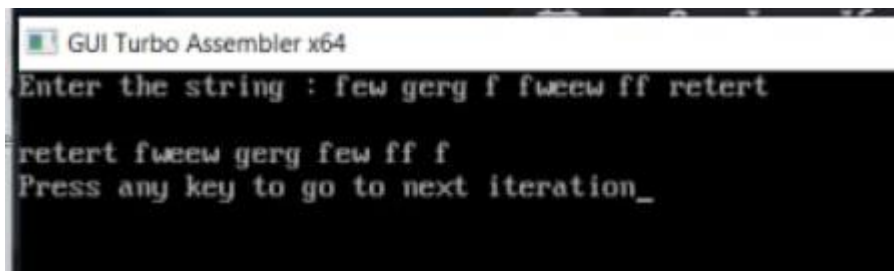


Рисунок 2 – вывод и предложение ввода запускаемой программы

Вывод: выполнено задание написать программу, запускающую другую программу N раз (N – число в диапазоне [1, 255]). Имя запускаемой программы задается константой. Таким образом были изучение основные команды (прерывания) работы с загрузкой и выполнение программ, а также работа с памятью.