

МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

**ЛАБОРАТОРНАЯ РАБОТА №8**

**ОТЧЁТ**

студента 2 курса 251 группы  
направления 09.03.04 — Программная инженерия  
факультета КНиИТ  
Григорьева Даниила Евгеньевича

Проверено:

Старший преподаватель

\_\_\_\_\_

Е. М. Черноусова

## **СОДЕРЖАНИЕ**

1 Текст программы на языке ассемблера с комментариями .....	3
2 Скриншот запуска программы .....	9

## 1 Текст программы на языке ассемблера с комментариями

```
extrn GetStdHandle:proc, WriteConsoleW:proc, ReadConsoleW:proc,
lstrlenW:proc, wprintfW:proc, ExitProcess:proc, ReadConsoleA:proc
STD_OUTPUT_HANDLE=-11
STD_INPUT_HANDLE=-10
NULL=0
MB_OK=0
dwMAX=2147483647          ; INT32_MAX (0x7FFFFFFF)
dwMIN=-2147483648         ; INT32_MIN (0x80000000)
wMAX=32767
wMIN=-32768
BUF_SIZE=1024
.data
    ; a =
    strA      db 61h, 00h, 20h, 00h, 3dh, 00h, 20h, 00h, 00h, 00h
    strALen   dw 4
    ; b =
    strB      db 62h, 00h, 20h, 00h, 3dh, 00h, 20h, 00h, 00h, 00h
    strBlen   dw 4
    ; 0x5D + %d - %d = %d
    strResult db 30h, 00h, 78h, 00h, 35h, 00h, 44h, 00h, 20h, 00h,
2bh, 00h, 20h, 00h, 25h, 00h, 64h, 00h, 20h, 00h, 2dh, 00h, 20h,
00h, 25h, 00h, 64h, 00h, 20h, 00h
                db 3dh, 00h, 20h, 00h, 25h, 00h, 64h, 00h, 00h, 00h

    ; Ошибка ввода числа
    erStoi    db 1eh, 04h, 48h, 04h, 38h, 04h, 31h, 04h, 3ah, 04h,
30h, 04h, 20h, 00h, 32h, 04h, 32h, 04h, 3eh, 04h, 34h, 04h, 30h,
04h, 20h, 00h, 47h, 04h, 38h, 04h
                db 41h, 04h, 3bh, 04h, 30h, 04h, 0ah, 00h
                db 00h, 00h
    erStoiLen dw 20

    varA      dw ?
    varB      dw ?
    varResult dq ?
    hCin      dq ?
    hCout     dq ?
    buffer    dw 512 dup (?)
    bufSize   dq 1024
    bufLen     dq 0
    SPACE db ' ', 0
    number    dq ?
; 5Dh + A - B
```

```

.code
initHandlers proc
    sub rsp, 8
    mov rcx, STD_INPUT_HANDLE
    call GetStdHandle
    mov [hCin], rax
    mov rcx, STD_OUTPUT_HANDLE
    call GetStdHandle
    mov [hCout], rax
    add rsp, 8
    ret
initHandlers endp
cout proc
    sub rsp, 28h
    mov rdx, rcx
    call lstrlenW
    mov r8, rax
    mov rcx, hCout
    xor r9, r9
    call WriteConsoleW
    add rsp, 28h
    ret
cout endp
cin proc
    sub rsp, 28h
    lea rax, buffer
    xor dl, dl
    mov rcx, BUF_SIZE
clear_buffer:
    mov [rax + rcx], dl
    loop clear_buffer
    mov rcx, hCin
    lea rdx, buffer
    mov r8, 1024
    lea r9, bufSize
    call ReadConsoleW
    lea rcx, buffer
    call lstrlenW
    sub rax, 2
    mov bufLen, rax
    add rsp, 28h
    ret
cin endp

```

```

stoi proc
local num:word, sign:byte, i:byte, _size:dword, digit:word
    push rbp
    mov num,0
    mov sign,1
    mov i,0
    skipLeadingSpace:
        mov rax,bufLen
        cmp i,al
    jae checkForSign
        lea rax,buffer
        movzx rcx,i
        mov ax, word ptr [rax+rcx*2]
        mov bx, 20h
        cmp ax, bx
    jne checkForSign
        inc cl
        mov i,cl
    jmp skipLeadingSpace
    checkForSign:
        movsxd rax,dword ptr [bufLen]
        cmp i,al
    jae convert_to_int
        lea rax, buffer
        movzx rcx,i
        movzx dx,byte ptr [rax+rcx*2]
        mov bx,2Dh
        cmp dx,bx
    je change_sign
        mov bx,2Bh
        cmp dx,bx
    jne convert_to_int
    change_sign:
        lea rax, buffer
        movzx rcx,i
        mov dx,word ptr [rax+rcx*2]
        mov bx,2Dh
        cmp dx,bx
    jne its_plus
        mov dl,-1
        jmp its_minus
    its_plus:
        mov dl,1
    its_minus:

```

```

        mov sign,dl
        mov al,i
        inc al
        mov i,al
convert_to_int:
    movsxd rax,dword ptr [bufLen]
    cmp i,al
    jae apply_sign_and_return
        lea rax,buffer
        movzx rcx,i
        mov ax,word ptr [rax+rcx*2]
        test ax,ax
    je apply_sign_and_return
    movzx rcx,i
    mov ax,word ptr [rax+rcx*2]
    mov bx, 20h
    cmp ax,bx
    jne check_is_digit
        jmp convert_to_int
check_is_digit:
    lea rax,buffer
    movzx rcx,i
    mov ax,word ptr [rax+rcx*2]
    mov dx, 30h
    cmp ax,dx
    jl parse_error
    mov dx, 39h
    cmp ax,dx
    jg parse_error
char_to_dword:
    lea rax,buffer
    movzx rcx,i
    mov ax,word ptr [rax+rcx*2]
    sub ax,30h
    mov digit,ax
    mov bx, wMAX / 10
    cmp num,bx
    jg parse_error
    cmp num,bx
    jne build_number
        mov bx, 7
        cmp digit,bx
        jg parse_error
build_number:

```

```

        mov ax, num
        mov bx, 0Ah
        imul bx
        mov bx, digit
        add ax, bx
        mov num, ax
        mov al, i
        inc al
        mov i, al
        jmp convert_to_int
apply_sign_and_return:
        mov ax, num
        movsx bx, sign
        imul ax, bx
        ret
parse_error:
        lea rcx, erStoi
        call cout
        xor rax, rax
        ret
    ret
stoi endp

input proc
    sub rsp, 8
    lea rcx, strA
    call cout
    call cin
    call stoi
    mov varA, ax

    lea rcx, strB
    call cout
    call cin
    call stoi
    mov varB, ax
    add rsp, 8
    ret
input endp
calc proc
    sub rsp, 8
    mov rax, 5Dh
    movsx rbx, varA
    movsx rcx, varB

```

```

        add rax, rbx
        sub rax, rcx
        mov varResult, rax
        add rsp, 8
        ret
calc endp
output proc
    sub rsp, 30h
    mov rbx, varResult
    mov [rsp + 20h], rbx
    lea rcx, buffer
    lea rdx, strResult
    movsx r8, varA
    movsx r9, varB
    call wsprintfW

    lea rcx, buffer
    call cout
    add rsp, 30h
    ret
output endp
mainCRTStartup proc
    push rbp
    call initHandlers

    call input
    call calc
    call output

    xor rcx, rcx
    call ExitProcess
    pop rbp
    ret
mainCRTStartup endp
end

```



## 2 Скриншот запуска программы

```
Z:\data\docs\sgu\code>lab8
a = 10
b = 15
0x5D + 10 - 15 = 88
Z:\data\docs\sgu\code>lab8
a = -100
b = 200
0x5D + -100 - 200 = -207
Z:\data\docs\sgu\code>lab8
a = 823985234985
Ошибка ввода числа
b = 10
0x5D + 0 - 10 = 83
Z:\data\docs\sgu\code>lab8
a = 123
b = aboba
Ошибка ввода числа
0x5D + 123 - 0 = 216
Z:\data\docs\sgu\code>lab8
a = -10
b = -10
0x5D + -10 - -10 = 93
Z:\data\docs\sgu\code>
```