**МИНОБРНАУКИ РОССИИ**

**ФЕДЕРАЛЬАНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**ИНЖЕНЕРНО-ТЕХНОЛОГИЧЕСКАЯ АКАДЕМИЯ**

**Институт компьютерных технологий и информационной безопасности Кафедра математического обеспечения и применения ЭВМ**

**Отчёт по лабораторной работе №6**

по курсу «Машинно-ориентированное программирование»

на тему: «Подпрограммы»

Выполнил: студент гр. КТбо2-7

Вахрушев Д.Е.

Проверил: доцент МОП ЭВМ, к.т.н.

Скороход С.В.

Таганрог 2022

# Вариант №2

**Постановка задачи**

Разработать подпрограмму, которая подсчитывает, сколько раз заданный символ встречается в строке. Разработать программу, которая вводит с клавиатуры строку, вводит число N и выдает список символов, которые встречаются в строке не менее чем N раз.

Необходимо:

1. Разработать программу, реализующую указанную функцию.
2. Исполнить программу с несколькими наборами исходных данных, проверить правильность результатов.

# Алгоритм решения

# Str\_promt: Получение строки

# Если строка нулефой длины

# Повторный ввод

# Ввод числа

# Если число нулевой длины или некорректно

# Повторный ввод числа

# Преобразование числа из строки в байт num

# Цикл от 255 до 0 по CX:

# Получить количество вхождений CX

# Если больше num

# Вывод символа

# Листинг программ

model SMALL

dataseg

str\_promt\_str db 0Dh, 0Ah, 'Enter string: $'

str\_max\_len db 30

str\_len db ?

str\_str db 30 dup(?)

num\_error\_str db 0Dh, 0Ah, 'Error! $'

num\_promt\_str db 0Dh, 0Ah, 'Enter num: $'

num\_max\_len db 4

num\_len db ?

num\_str db 4 dup(?)

num db ?

ten db 10

out\_str db 0Dh, 0Ah, '" " $'

codeseg

startupcode

mov AX, @DATA

mov ES, AX

mov DS, AX

mov BX, 0

mov DX, 0

str\_promt:

lea DX, [str\_promt\_str]

mov AH, 09h

int 21h

mov AH, 0Ah

lea DX, [str\_max\_len]

int 21h

cmp [str\_len], 0

jz str\_promt

jmp get\_num

error\_num:

lea DX, [num\_error\_str]

mov AH, 09h

int 21h

get\_num:

lea DX, [num\_promt\_str]

mov AH, 09h

int 21h

mov AH, 0Ah

lea DX, [num\_max\_len]

int 21h

cmp [num\_len], 0

jz error\_num

lea SI, [num\_str]

mov [num], 0

get\_digit:

lodsb

cmp AL, 0Dh

jz end\_get\_num

sub AL, '0'

cmp AL, 10

jnb error\_num

mov CL, AL

mov AL, num

mul ten

cmp AH, 0

jnz error\_num

add AL, CL

jo error\_num

mov num, AL

jmp get\_digit

end\_get\_num:

cmp num, 0

jz error\_num

lea DI, [str\_str]

mov DH, 0

mov DL, [str\_len]

mov BH, 0

mov CX, 255

count\_s:

mov BL, CL

call my\_prnt\_func

cmp AH, num

jnae skip\_print

mov [out\_str+3], AL

push DX

lea DX, [out\_str]

mov AH, 09h

int 21h

pop DX

skip\_print:

loop count\_s

exitcode 0

my\_prnt\_func proc PASCAL

uses DI,CX

mov AH, 255

mov AL, BL

mov CX, DX

inc CX

begin:

inc AH

repnz scasb

jcxz $+4

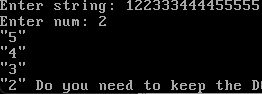
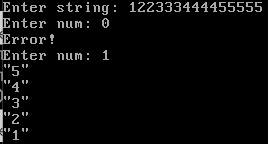
jmp begin

ret

my\_prnt\_func endp

end

# Результаты тестирования

1. Пример работы программы  
   
2. Пример работы программы   
   
3. Пример работы программы  
   
4. Пример работы программы  
   

# Вывод

При выполнении лабораторной работы я получил практические навыки работы с функциями и стеком.