

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Департамент программной инженерии

**КОНСОЛЬНОЕ ПРИЛОЖЕНИЕ, ВЫЧИСЛЯЮЩЕЕ ЧИСЛО ВХОЖДЕНИЙ
СИМВОЛОВ В ЗАДАННОЙ ASCII-СТРОКЕ**

Пояснительная записка

Исполнитель:
Студентка группы БПИ199
_____/Точилина П. В./
«1» ноября 2020 г.

Содержание

1.Текст задания	2
2. Применяемые расчетные методы	3
2.1 Теория решения задания	3
2.2 Ограничения на ввод.....	3
2.3 Дополнительные функционалы программы.....	3
3. Тестирование программы.....	4
3.1 Корректные данные.....	4
3.2 Некорректные данные	5
4. Список источников.	6
<i>Приложение 1</i>	<i>7</i>
<i>Приложение 2</i>	<i>11</i>

1.Текст задания

Разработать программу, вычисляющую число вхождений символов в заданной ASCII-строке.

2. Применяемые расчетные методы

2.1 Теория решения задания

Для вычисления числа вхождений каждого символа создаётся массив, индекс элемента в котором означает ASCII-код символа, а значение элемента – число вхождений данного символа. После чего для каждого символа в строке инкрементируется значение соответствующего элемента массива.

2.2 Ограничения на ввод

Входная строка должна содержать не более 1000 символов.

2.3 Дополнительные функционалы программы

При выводе программа итерируется по массиву и выводит только ненулевые элементы для удобства пользователя.

3. Тестирование программы

3.1 Корректные данные

Проверим строку на английских буквах.

```
Input a string: abcdefghijklmnopqrstuvwxyz
Your string: abcdefghijklmnopqrstuvwxyz

"a" symbol was repeated 1 times
"b" symbol was repeated 1 times
"c" symbol was repeated 1 times
"d" symbol was repeated 1 times
"e" symbol was repeated 1 times
"f" symbol was repeated 1 times
"g" symbol was repeated 1 times
"h" symbol was repeated 1 times
"i" symbol was repeated 1 times
"j" symbol was repeated 1 times
"k" symbol was repeated 1 times
"l" symbol was repeated 1 times
"m" symbol was repeated 1 times
"n" symbol was repeated 1 times
"o" symbol was repeated 1 times
"p" symbol was repeated 1 times
"q" symbol was repeated 1 times
"r" symbol was repeated 1 times
"s" symbol was repeated 1 times
"t" symbol was repeated 1 times
"u" symbol was repeated 1 times
"v" symbol was repeated 1 times
"w" symbol was repeated 1 times
"x" symbol was repeated 1 times
"y" symbol was repeated 1 times
"z" symbol was repeated 1 times
Other symbols did not appear in the string
```

Рисунок 1. Тест 1

Проверим на тексте с пробелами

```
Input a string: мама мыла раму
Your string: мама мыла раму

" " symbol was repeated 2 times
"a" symbol was repeated 4 times
"л" symbol was repeated 1 times
"м" symbol was repeated 4 times
"р" symbol was repeated 1 times
"у" symbol was repeated 1 times
"ы" symbol was repeated 1 times
Other symbols did not appear in the string
```

Рисунок 2. Тест 2

Введём 100 одинаковых символов

```
Input a string: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
Your string: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
"a" symbol was repeated 100 times
Other symbols did not appear in the string
```

Рисунок 3. Тест 3

3.2 Некорректные данные

Ввод более 1000 символов. Все лишние символы обрежутся, программа отработает на оставшихся символах. Для проверки сгенерировано 1000 буквенных символов и добавлено сверх числовыми значениями, которые обрезались.

```
Input a string: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim. Aliquam lorem ante, dapibus in, viverra quis, feugiat a, tellus. Phasellus viverra nulla ut metus varius laoreet. Quisque rutrum. Aenean imperdiet. Etiam ultricies nisi vel augue. Curabitur ullamcorper ultricies nisi. Nam eget dui. Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet adipiscing sem neque sed ipsum. N46574586996574362545678979654632467658

Your string: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim. Aliquam lorem ante, dapibus in, viverra quis, feugiat a, tellus. Phasellus viverra nulla ut metus varius laoreet. Quisque rutrum. Aenean imperdiet. Etiam ultricies nisi vel augue. Curabitur ullamcorper ultricies nisi. Nam eget dui. Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet adipiscing sem neque sed ipsum. N

" " symbol was repeated 143 times
" " symbol was repeated 25 times
"." symbol was repeated 23 times
"A" symbol was repeated 6 times
"C" symbol was repeated 3 times
"D" symbol was repeated 2 times
"E" symbol was repeated 2 times
"I" symbol was repeated 2 times
"L" symbol was repeated 1 times
"M" symbol was repeated 1 times
"N" symbol was repeated 4 times
"P" symbol was repeated 1 times
"Q" symbol was repeated 1 times
"V" symbol was repeated 1 times
"a" symbol was repeated 58 times
"b" symbol was repeated 5 times
"c" symbol was repeated 27 times
"d" symbol was repeated 20 times
"e" symbol was repeated 113 times
"f" symbol was repeated 6 times
"g" symbol was repeated 13 times
"h" symbol was repeated 4 times
"i" symbol was repeated 76 times
"j" symbol was repeated 3 times
"l" symbol was repeated 48 times
"m" symbol was repeated 44 times
"n" symbol was repeated 52 times
"o" symbol was repeated 31 times
"p" symbol was repeated 22 times
"q" symbol was repeated 13 times
"r" symbol was repeated 40 times
"s" symbol was repeated 62 times
"t" symbol was repeated 62 times
"u" symbol was repeated 73 times
"v" symbol was repeated 13 times
Other symbols did not appear in the string
-
```

Рисунок 4. Тест 4

4. Список источников.

1) Требования к оформлению [Электронный ресурс].

Режим доступа: <http://softcraft.ru/edu/comparch/tasks/mp01/>, свободный.

2) Документация FASM [Электронный ресурс].

Режим доступа: <http://flatassembler.narod.ru/fasm.htm>, свободный.

Основной файл программы

```
;Точилина Полина
;БПИ199
;Вариант 25
;Разработать программу, вычисляющую число вхождений символов в заданной
ASCII-строке
;Дополнительное ограничение - входная строка не более 1000 символов
format PE console

entry start

include 'win32a.inc'
include 'ASCII_counter_macro.inc'

section '.data' data readable writeable

    ;форматирование вывода числа
    formatNum db '%d', 0
    ;форматирование вывода строки
    formatStr db '%s', 0
    ;форматирование ввода строки с пробелами с ограничением
    formatInpStr db '%1000[^',10,']c', 0

    ;строка вывода каждого элемента
    outputElementString db '"%c" symbol was repeated %d times', 10,
13, 0

    ;запрос на ввод строки
    stringStr db 'Input a string: ', 0
    ;вывод информации о введённой строке
    yourStringStr db 'Your string: ', 0
    ;вывод информации о непосчитанных символах
    otherSymb db 'Other symbols did not appear in the string', 10,
13, 0

    ;переход на новую строку
```



```
endlStr db 10, 13, 0
```

```
;для хранения введенной строки
```

```
inputStr rb 1000;
```

```
;для хранения длины введенной строки
```

```
strLen dd ?
```

```
;кол-во ASCII-символов
```

```
N dd 256
```

```
;для хранения массива символов
```

```
A dd ?
```

```
;для сохранения стека
```

```
tmpStack dd ?
```

```
;итератор для массива
```

```
i dd 0
```

```
NULL = 0
```

```
section '.code' code readable executable
```

```
start:
```

```
; ввод строки
```

```
cinvoke printf, formatStr, stringStr
```

```
cinvoke scanf, formatInpStr, inputStr
```

```
cinvoke printf, formatStr, endlStr
```

```
; вывод введенной строки
```

```
cinvoke printf, formatStr, yourStringStr
```

```
cinvoke printf, formatStr, inputStr
```

```
cinvoke printf, formatStr, endlStr
```

```
cinvoke printf, formatStr, endlStr
```

```
; считаем длину строки для итерации по ней
```

```
push inputStr
```

```
call strlen
```

```

        mov [strLen], eax

        ; выделяем память для массива кодов ASCII
        invoke VirtualAlloc, NULL, [N],
MEM_COMMIT+MEM_RESERVE, PAGE_READWRITE
        mov [A], eax

        ; считаем число вхождений каждого символа
        call countAscii

        ; вывод массива
        call outputArray

        cinvoke printf, otherSymb

        call [getch]

        push NULL
        call [ExitProcess]

; макросы
count_ASCII_macro
out_macro
strLen_macro

section '.idata' import data readable

library kernel, 'kernel32.dll',\
        msvcrt, 'msvcrt.dll'

import kernel,\
        ExitProcess, 'ExitProcess',\
        VirtualAlloc, 'VirtualAlloc',\
        VirtualFree, 'VirtualFree'

```

```
import msvcrt,\n    printf, 'printf',\n    getch, '_getch',\n    scanf, 'scanf'
```

Макросы

```

macro count_ASCII_macro {
;-----
-----
; Считает число вхождений каждого символа в строке
countAscii:
    mov [tmpStack], esp
    mov ecx, [strLen]
    mov esi, inputStr
_loop:
        ; чтобы сохранить значение счетчика
        push ecx
        ; обновляем ссылку на первый элемент массива
        mov edi, [A]

        ; получаем ASCII код символа
        mov eax, 0
        lodsb

        ; инкрементируем элемент с индексом ASCII кода
символа

        mov edx, eax
        imul edx, dword 4
        add edi, edx
        mov eax, [edi]
        inc eax
        mov [edi], eax

        ;; обновляем счётчик и зацкливаем
        pop ecx
        LOOP _loop

    mov esp, [tmpStack]
    ret
}

```

```

macro out_macro {
;-----
-----
; Выводит символы, повторенные как минимум один раз
outputArray:
    ;сохраняем стэк для адреса возврата
    mov [tmpStack], esp
    mov ecx, 0
    mov [i], ecx

    mov ecx, [N]

lp:
    push ecx
    ;вычисляем адрес новой ячейки
    mov ecx, [i]
    imul ecx, dword 4
    mov eax, [A]
    add eax, ecx

    ;выводим значение, если оно не 0
    mov ecx, 0
    cmp [eax], ecx
    jne outputElement
    contin:

    ;зацикливаем
    inc[i]
    pop ecx
    LOOP lp

    mov esp, [tmpStack]
    ret

outputElement:

```

```

        cinvoke printf, outputElementString, [i], [eax]
        jmp contin
    }

```

```

macro strlen_macro {

```

```

;-----
-----

```

```

; Считает длину строки

```

```

    strlen:

```

```

        mov     edi, [esp+4]

```

```

        mov     ecx, -1

```

```

        xor     al, al

```

```

        ; с начала в конец

```

```

        cld

```

```

        ; while(msg[edi] != al) {edi++; ecx--;}

```

```

        repne   scasb

```

```

        neg     ecx

```

```

        sub     ecx, 2

```

```

        mov     eax, ecx

```

```

        ret

```

```

    }

```