*Федеральное государственное бюджетное образовательное учреждение* *высшего образования*

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана***  ***(национальный исследовательский университет)»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

**Отчет**

**по лабораторной работе № 5**

**Дисциплина:** Машинно-зависимые языки программирования

**Название лабораторной работы:**

Программирование с использованием разноязыковых модулей

Студент гр. ИУ6-42  **\_\_\_\_\_\_25.04.2018 Бурлаков Александр Сергеевич**

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О. Фамилия)

Москва, 2018

**ЗАДАНИЕ**

Дан текст не более 255 символов. Определить частоту повторения каждого символа.

**ОСНОВНАЯ ЧАСТЬ**

1. **Схема алгоритма**

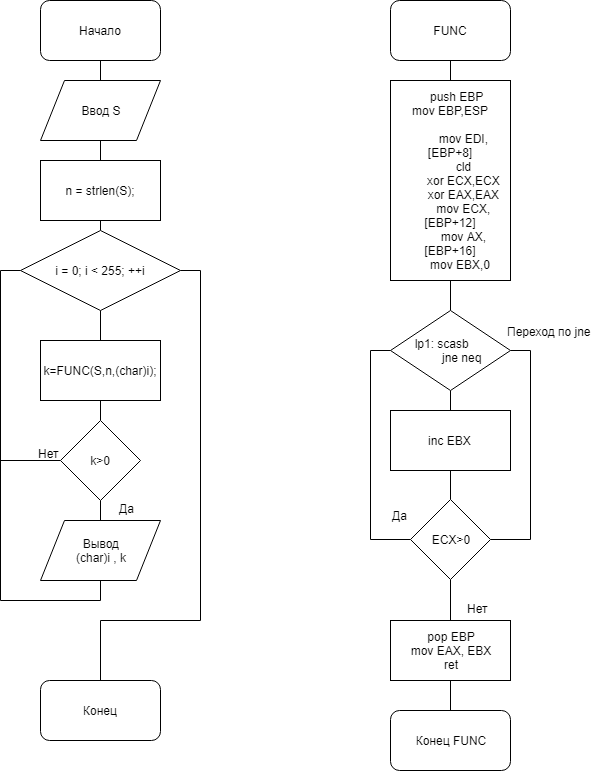


Рисунок 1 – схема алгоритма разрабатываемой программы

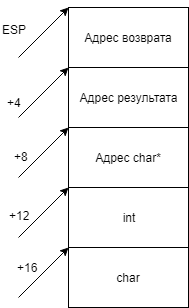


Рисунок 2 – стек в момент передачи управления

1. **Код разработанной программы**

**Код программы на C++:**

#include <iostream>

#include <cstring>

using namespace std;

extern "C" int FUNC(char \*, int, char); // Возвращает кол-во

int main() {

int k, n;

system("chcp 1251");

system("cls");

char S[255];

gets(S);

n = strlen(S);

for (int i = 0; i < 255; ++i){

k = FUNC(S, n, (char)i);

if (k > 0){

cout << (char)i << " " << k << endl;

}

}

return 0;

}

**Код программы на Assembler:**

.586

.MODEL FLAT, C

.CODE

FUNC PROC

push EBP

mov EBP,ESP

mov EDI,[EBP+8]

cld

xor ECX,ECX

xor EAX,EAX

mov ECX,[EBP+12]

mov AX,[EBP+16]

mov EBX,0

lp1: scasb

jne neq

inc EBX

neq:

loop lp1

pop EBP

mov EAX, EBX

ret

FUNC ENDP

END

1. **Тестирование программы**

Таблица 1 – результаты тестирования программы

|  |  |  |
| --- | --- | --- |
| Входные данные | Ожидаемые результаты | Полученные результаты |
| 1234 | 1-1, 2-1, 3-1, 4-1 | 1 1  2 1  3 1  4 1 |
| gasdg | g-2, a-1, s-1, d-1 | a 1  d 1  g 2  s 1 |
| patapoN | p-2, a-2, t-1, o-1, N-1 | N 1  a 2  o 1  p 2  t 1 |

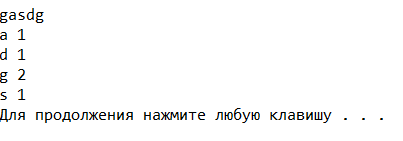
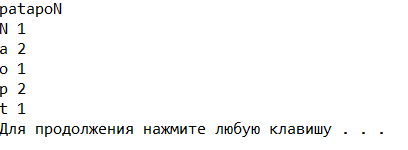


Рисунок 3 – Результаты тестирования программы

**ВЫВОД**

1. Разработан алгоритм, позволяющий решить данное задание.
2. Написан код программы по разработанному алгоритму.
3. Произведено тестирование программы, тестирование показало корректность работы программы.