|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«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)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.03 Прикладная информатика**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 6 |

**Название:**

Динамические структуры данных. Списки

**Дисциплина:** Объектно-ориентированное программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-25 Б |  |  |  |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  | 20.03.2023 |  |
| Преподаватель |  |  |  | Scan0023Scan0026 |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

20.03.2023

Москва, 2023

**Вариант 8**

**Задание:**

Дана символьная строка, состоящая из целых чисел в символьном представлении, разделенных пробелом. Составить программу, которая вводит строку, организует из чисел-слов однонаправленный список, упорядочивая его по длине слова. На его основе строит новый список, преобразуя числа из символьного представления в целое. Распечатать списки, определить и вывести на печать сумму цифр каждого элемента списка.

**Код программы**

#include <iostream>

#include <string.h>

#include <stdlib.h>

using namespace std;

const int numb = 30;

//сортировка списк

char\* sort(char \*str) {

char\* ptr, \*token, \*nexttoken, \*a[numb], \*c;

int count = 0;

ptr = &str[0];

token = strtok\_s(str, " ", &nexttoken);

while (token != nullptr) {

a[count] = new char[numb];

strcpy\_s(a[count], numb, token);

count++;

token = strtok\_s(nullptr, " ", &nexttoken);

}

strcpy\_s(str, sizeof(str), "");

for (int i = 0; i < count-1; i++) {

for (int j = count - 2; j >= 0; j--)

if (strlen(a[j]) > strlen(a[j + 1])) {

c = a[j];

a[j] = a[j + 1];

a[j + 1] = c;

}

}

for (int i = 0; i <= count - 1; i++) {

strcat\_s(str, numb, a[i]);

strcat\_s(str, numb, " ");

delete[]a[i];

}

return str;

}

struct zap { char numb[30]; zap\* p; }; struct zap2 { unsigned int numb; zap2\* p; };

int main() {

zap \* r1, \*v1, \* q /\*первый список\*/;

zap2 \*f, \* r2, \*v2/\*второй список\*/;

char s[numb], \* ptr1, \* ptr2, n, \*token, \*nexttoken;

int k, summ;

cout << "Input string\n";

gets\_s(s);

sort(s);

token = strtok\_s(s, " ", &nexttoken);

q = new zap;

f = new zap2;

strcpy\_s(q->numb, token);

f->numb = atoi(q->numb);

f->p = nullptr;

q->p = nullptr;

v1 = q; v2 = f;

token = strtok\_s(nullptr, " ", &nexttoken);

//формирование списков

while (token != nullptr) {

r1 = new zap;

r2 = new zap2;

r1->p = nullptr;

r2->p = nullptr;

strcpy\_s(r1->numb, token);

r2->numb = atoi(r1->numb);

v1->p = r1;

v2->p = r2;

v1 = r1;

v2 = r2;

token = strtok\_s(nullptr, " ", &nexttoken);

}

r2 = f;

r1 = q;

//вывод списков

cout << "\nList1\n";

if (q == nullptr) cout << "Not found\n";

else do {

cout << q->numb << ' ' << endl;

q = q->p;

} while (q != nullptr);

cout << "\nList2\n";

if (f == nullptr) cout << "Not found\n";

else {

do {

cout << r2->numb << ' ' << endl;

r2 = r2->p;

} while (r2 != nullptr);

cout << "\nThe sums of numbers\n";

do {

summ = 0;

k = 0;

\_itoa\_s(f->numb, s, 10);

while (k != strlen(s)) {

n = s[k];

summ += atoi(&n);

k+=1;

}

cout << summ << endl;

f = f->p;

} while (f != nullptr);

}

return 0;

}

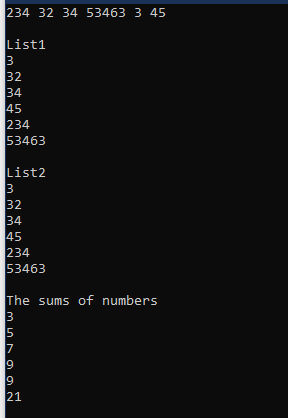
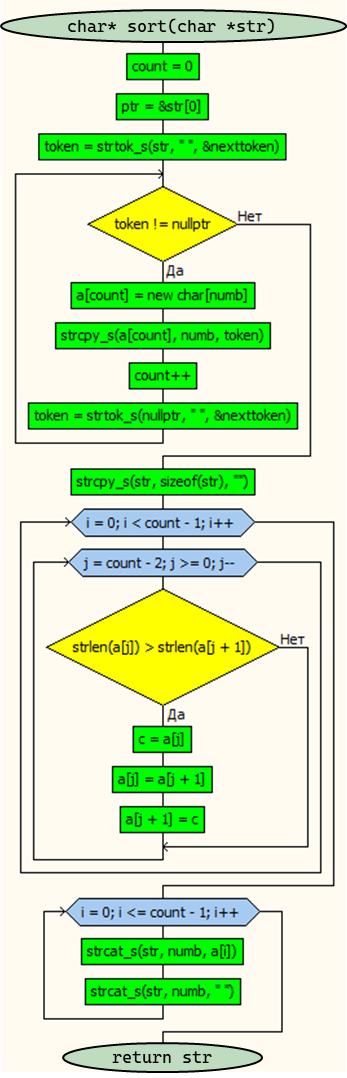


Рисунок 1 – пример работающей версии программы



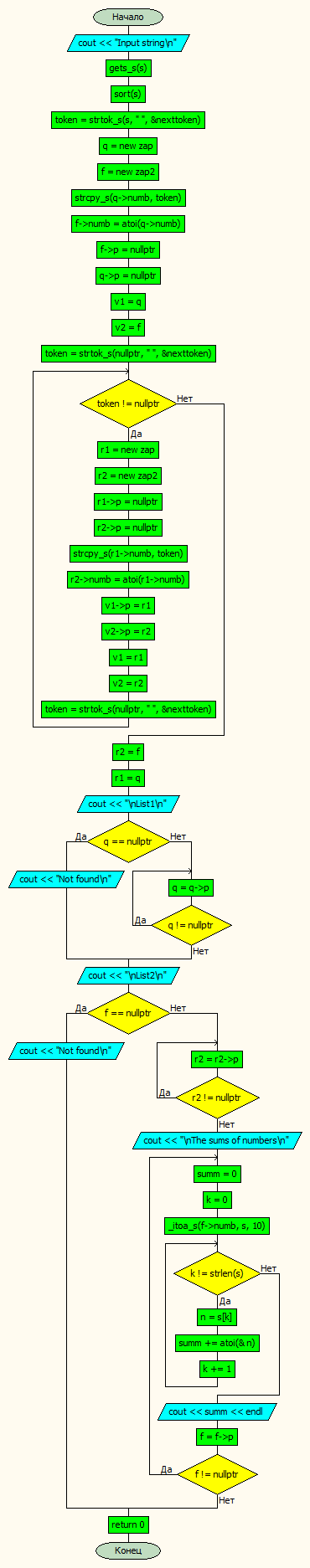
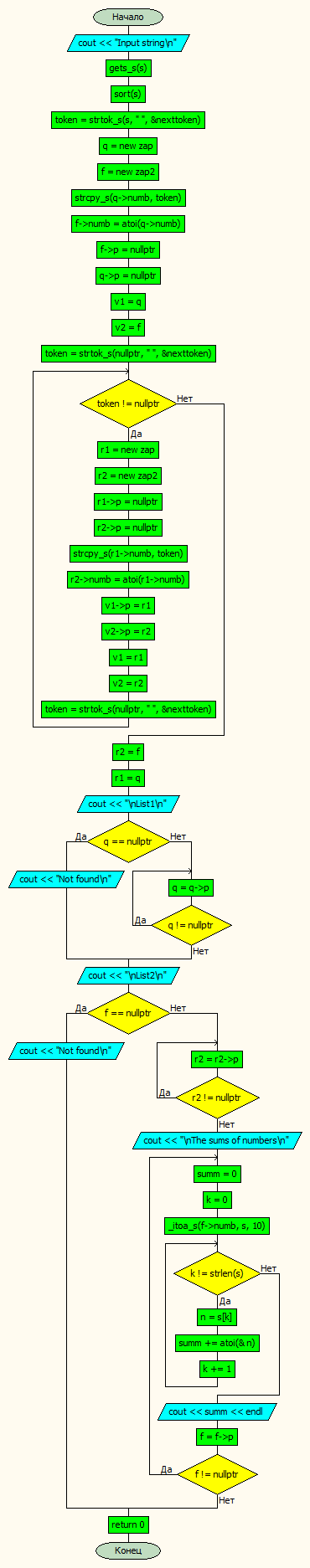


Рисунок 2 – блок-схема

**Вывод**: мы научились работать со списками на языке C++