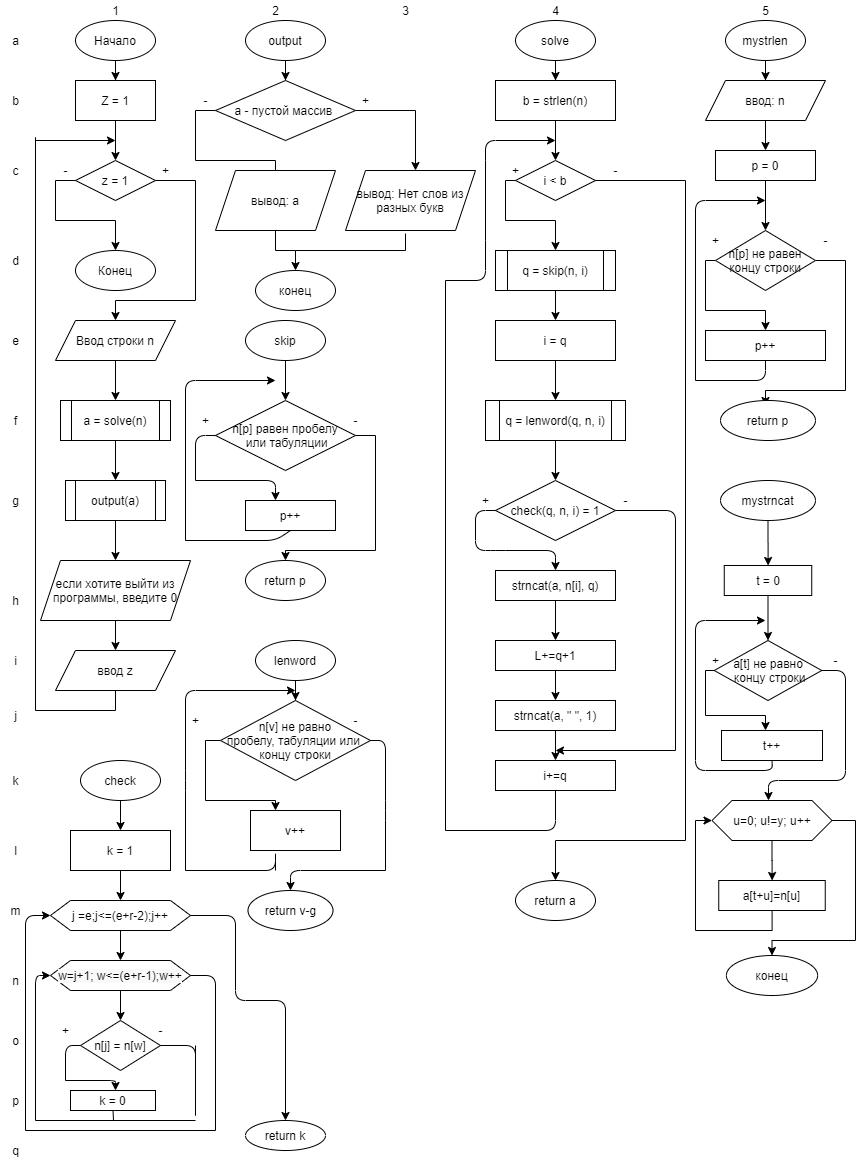
Самородов Роман Б20-513

3.44. Найти в строке слова, состоящие из разных букв.

**1. Блок-схема алгоритма работы основной программы и функций по обработке строк.**



**2. Исходные коды всех программ.**

lab3\_1 (С функциями из библиотеки)

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <time.h>

int skip (char\* n, int p) //пропускаем пробелы и табуляции до буквы

{

while (n[p] == ' ' || n[p] == '\t' )

{

p++;

}

return (p);

}

int lenword (int v, char\* n, int g) //длина слова

{

while (n[v] != ' ' && n[v] != '\t' && n[v] != '\0')

{

v++;

}

v = v - g;

return v;

}

int check (char\* n, int e, int r)

{

int k = 1;

for(int j = e; j<= (e+r-2); j++)//цикл рассматривает все буквы слова

{

for(int w = j+1; w <= (e+r-1); w++)

{

if (n[j] == n[w])

{

k = 0;

}

}

}

return (k);

}

char \*solve (char \*n)

{

int q, i = 0;

char \*a=calloc(strlen(n)+1, sizeof(char)); // a - строка-результат

int L = 0; // L - длина строки-результата

int b = strlen(n);// длина введенной строки

while (i < b) //проходим строчку по каждому символу

{

q = skip (n, i); //пропускаем пробелы и табуляции до буквы

i = q;

q = lenword(q, n, i); // длина слова

if (check(n, i, q) == 1)// если в слове все буквы разные, то оно добавляется в строку-результат

{

strncat (a, &n[i], q);

L+= q + 1; // увеличиваем длину строки-результата на количество букв в слове + 1 (пробел)

strncat(a, " ", 1);

}

i += q;// следующий символ после конца слова

}

a = realloc(a, L + 1);

a[L] = '\0';

return a;

}

void output (char\* a)

{

if(a[0] == '\0')//если строка-результат ненулевая- выводим ее, иначе выводим соответствующий отчет

{

printf("Нет слов из разных букв\n");

}

else {

printf("%s\n", a);

}

}

int main()

{

int z = 1;

while (z)

{

char n[203];

char \*a = NULL;

clock\_t begin = clock();//начало отсчета времяни работы программы со строкой

scanf("%200[^\n]s" , n);

a = solve(n);

output (a); //вывод результата

free(a);

a = NULL;

clock\_t end = clock();

double time\_spent = (double)(end - begin);

printf("время работы программы: %f мс\n", time\_spent);// время работы программы со строкой

printf("Если хотите выйти из программы введите 0\n");

scanf("%d", &z);

}

return 0;

}

Lab3\_2 (со своими реализациями функций)

#pragma once

int mystrlen(char\* n)

{

int p = 0;

while (n[p])

{

++p;

}

return p;

}

void mystrncat(char\* a, char\* n, int y)

{

int t = 0;

while (a[t] != '\0')

{

t++;//длина строки a до дополнения

}

for(int u = 0; u != y; u++)

{

a[t+u] = n[u];

}

}

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include "mystring.h"

int skip (char\* n, int p) //пропускаем пробелы и табуляции до буквы

{

while (n[p] == ' ' || n[p] == '\t' )

{

p++;

}

return (p);

}

int lenword (int v, char\* n, int g) //длина слова

{

while (n[v] != ' ' && n[v] != '\t' && n[v] != '\0')

{

v++;

}

v = v - g;

return v;

}

int check (char\* n, int e, int r)

{

int k = 1;

for(int j = e; j<= (e+r-2); j++)//цикл рассматривает все буквы слова

{

for(int w = j+1; w <= (e+r-1); w++)

{

if (n[j] == n[w])

{

k = 0;

}

}

}

return (k);

}

char \*solve (char \*n)

{

int q, i = 0;

char \*a=calloc(mystrlen(n)+1, sizeof(char)); // a - строка-результат

int L = 0; // L - длина строки-результата

int b = mystrlen(n);// длина введенной строки

while (i < b) //проходим строчку по каждому символу

{

q = skip (n, i); //пропускаем пробелы и табуляции до буквы

i = q;

q = lenword(q, n, i); // длина слова

if (check(n, i, q) == 1)// если в слове все буквы разные, то оно добавляется в строку-результат

{

mystrncat (a, &n[i], q);

L+= q + 1; // увеличиваем длину строки-результата на количество букв в слове + 1 (пробел)

mystrncat(a, " ", 1);

}

i += q;// следующий символ после конца слова

}

a = realloc(a, L + 1);

a[L] = '\0';

return a;

}

void output (char\* a)

{

if(a[0] == '\0')//если строка-результат ненулевая- выводим ее, иначе выводим соответствующий отчет

{

printf("Нет слов из разных букв\n");

}

else {

printf("%s\n", a);

}

}

int main()

{

int z = 1;

while (z)

{

char n[203];

char \*a = NULL;

clock\_t begin = clock();//начало отсчета времяни работы программы со строкой

scanf("%200[^\n]s" , n);

a = solve(n);

output (a); //вывод результата

free(a);

a = NULL;

clock\_t end = clock();

double time\_spent = (double)(end - begin);

printf("время работы программы: %f мс\n", time\_spent);// время работы программы со строкой

printf("Если хотите выйти из программы введите 0\n");

scanf("%d", &z);

}

return 0;

}

**3. Тестовые наборы для программ п. 1 и п. 2.**

1) входная строка: i am gonna play football

выходная строка: i am play

2) входная строка: pan pineapple apple pan

выходная строка: pan pan

3) входная строка: i really want to sleep

выходная строка: i want to

4) входная строка: ananas apple pineapple

выходная строка: Нет слов из разных букв

5) входная строка: ertyu ghjk ghjk bnnms ghjk vvvfr ggghjks bbbnn ert ghj

выходная строка: ertyu ghjk ghjk ghjk ert ghj

**4. Сравнительный анализ времени, потраченного на решение задачи**

**программами п. 1 и п. 2.**

|  |  |  |
| --- | --- | --- |
| Номер теста | Lab3\_1 | Lab3\_2 |
| 1 | 26 мс | 27 мс |
| 2 | 26 мс | 27 мс |
| 3 | 25 мс | 27 мс |
| 4 | 27 мс | 28 мс |
| 5 | 31 мс | 36 мс |

**5. Valgrind**

