

ГУАП

КАФЕДРА № 53

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ _____
ПРЕПОДАВАТЕЛЬ

старший преподаватель
должность, уч. степень, звание

подпись, дата

Ушаков В.А.
инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №5

Анализ строк, хеширование

Вариант 3

по курсу: ОСНОВЫ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ(А)

СТУДЕНТ ГР. № 5138

подпись, дата

Воробьев В.А.
инициалы, фамилия

Санкт-Петербург 2022

Задание: реализовать программную функцию на языке C/C++, выполняющую поставленную задачу. Глобальные параметры использовать запрещено; допустимо использование дополнительных функций.

3	Совпадение хеша Задано основание хеша P и модуль M . Вывести строки, полученные удалением любого количества символов исходной строки S размерностью N , для которых вычисленный хеш будет одинаков. С указанием значения хеша. Если таких строк нет, то вывести false.	$P : 3$ $M : 10$ $S : \text{"aabc"}$	$4 : [\text{"abc"},$ $\text{"abc"}, \text{"aa"}];$ $1 : [\text{"aac"},$ $\text{"bc"}];$ $7 : [\text{"ab"},$ $\text{"ab"}]$
---	--	--	---

Рисунок 1 - вариант задания

Выполнение задания:

```
1  #include <iostream>
2  #include <vector>
3  #include <Windows.h>
4  #include <set>
5  #include <string>
6  #include <map>
7  #include <math.h>
8
9  using namespace std;
10
11 int hashsum(string input, int P, int M) {
12     int sum = 0;
13     for (int i = 0; i <= input.size(); i++) {
14         sum += input[i] * pow(P, i);
15     }
16     return sum % M;
17 }
18
19 string cutString(string input, set<int> indexesLeft) {
20     string res = "";
21     for (int i = 0; i < input.size(); i++) {
22         if (indexesLeft.find(i) != indexesLeft.end()) {
23             res += input[i];
24         }
25     }
26     return res;
27 }
28
29 // Сочетания без повторов.
30 vector<set<int>> sample(int length) {
31     vector<set<int>> res = vector<set<int>>();
32     res.push_back(set<int>{-1});
33     for (int i = 0; i < length; i++) {
34         long startSize = res.size();
35         for (int j = 0; j < startSize; j++) {
36             set<int> newSet(res[j]);
37             newSet.insert(i);
38             res.push_back(newSet);
39         }
40     }
41     return res;
42 }
43
44
45 int main()
46 {
47     setlocale(LC_ALL, "");
48     SetConsoleCP(1251);
49     SetConsoleOutputCP(1251);
50     string b;
51     int P, M;
52     cout << "S: ";
53     cin >> b;
54     cout << "P: ";
55     cin >> P;
56     cout << "M: ";
57     cin >> M;
```

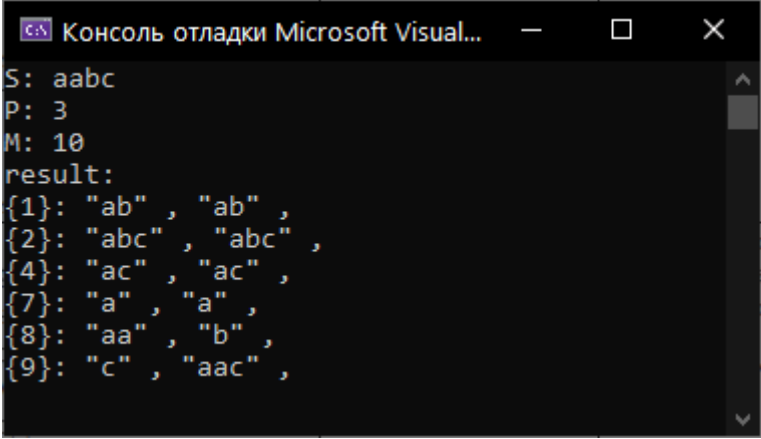
Рисунок 2.1 – код

```

57 cin >> M;
58 vector<set<int>> samples = sample(b.size());
59 map<int, vector<string>> dict = map<int, vector<string>>();
60
61 /* ... */
62
63 //Добавляем в словарь обрезанные строки по их хеш-сумме.
64 for (int i = 1; i < samples.size(); i++) {
65     string cuttedStr = cutString(b, samples[i]);
66     dict[hashsum(cuttedStr, P, M)].push_back(cuttedStr);
67 }
68
69 map<int, vector<string>> ::iterator ii;
70 cout << "result: \n";
71 bool twoHashes = false; // Было минимум два одинаковых хеша.
72 for (ii = dict.begin(); ii != dict.end(); ii++) {
73     if ((ii->second).size() < 2) continue;
74     twoHashes = true;
75     cout << "{" << ii->first << ": "; // отображаем хеш.
76     vector<string> strings = ii->second;
77     for (int i = 0; i < strings.size(); i++) {
78         cout << "\"" << strings[i] << "\" , "; // отображаем строку.
79     }
80     cout << "\n";
81 }
82 if (!twoHashes) {
83     cout << "false";
84 }
85 }

```

Рисунок 2.2 – код



Консоль отладки Microsoft Visual...

```

S: aabc
P: 3
M: 10
result:
{1}: "ab" , "ab" ,
{2}: "abc" , "abc" ,
{4}: "ac" , "ac" ,
{7}: "a" , "a" ,
{8}: "aa" , "b" ,
{9}: "c" , "aac" ,

```

Рисунок 3 - ввод и вывод

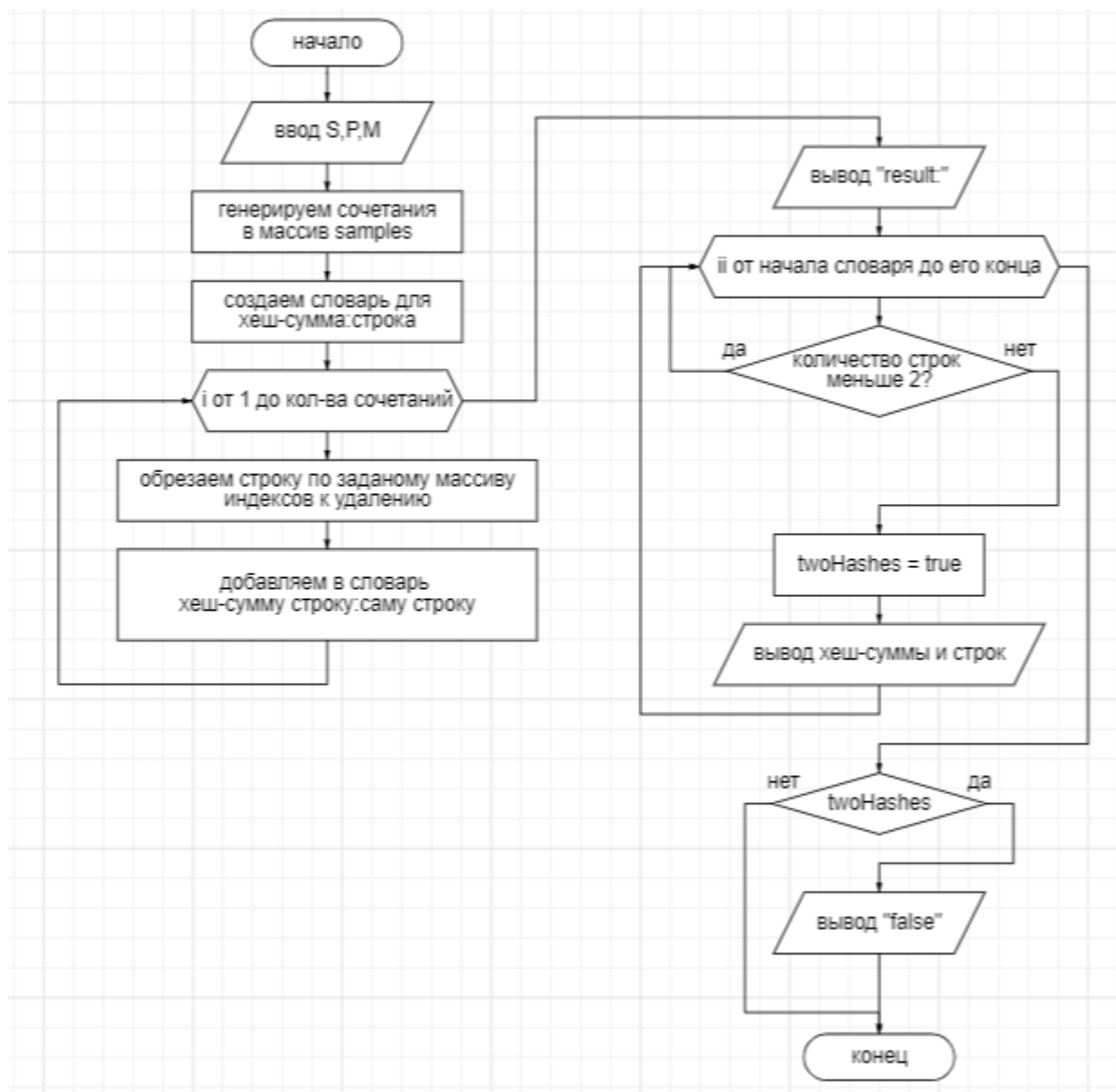


Рисунок 4 - Блок-схема

Вывод: в ходе выполнения работы создал простейший алгоритм хеширования, а на основе его алгоритма анализа строки. Понял принцип хеширования объектов и методы работы с хешем объекта.