



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»
(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова)

Факультет

О

Естественнонаучный

шифр

наименование

Кафедра

О7

Информационные системы и программная инженерия

шифр

наименование

Дисциплина

Информационные технологии и программирование

ПРАКТИЧЕСКАЯ РАБОТА №4

STL

Вариант №6

Выполнил студент группы

О722Б

Вяткин Н.А.

Фамилия И.О.

ПРЕПОДАВАТЕЛЬ

Гладевич А.А.

Фамилия И.О.

Подпись

« 13 »

апреля

2023 г.

САНКТ-ПЕТЕРБУРГ

2023

Постановка задачи

Дан файл с английскими словами, разделёнными знаком '-'. Используя (`std::getline` или `regex`) составить множество (`std::set`) из уникальных слов в файле. Слова хранить в `std::string`. Также составить словарь (`std::map`) по следующему принципу: Ключ - буква, значение - количество слов, содержащих эту букву в файле.

Сгенерировать список из 100 структур со следующими полями: Название - случайное слово из множества Основной параметр - используя (`std::advance`) получить значения трёх случайных элементов из словаря и перемножить их Вторичный параметр - квадрат значения из словаря, связанного с названием(в случае буквы брать первую букву названия). Используя функции из библиотеки (`algorithms`) Отсортировать список по убыванию названия Используя (`std::copy_if`), создать новый список, содержащий только: Элементы с нечётным числом букв в названии Сгенерировать на основе полученного отфильтрованного списка файл формата `json`.

Добавить возможность загрузки файла формата `json`, создания на основе данных в нём списка, перетасовки (`shuffle`) его и отображения полученного результата.

Текст программы

Файл main.cpp:

```
#include <iostream>
#include <fstream>
#include <string>
#include <locale.h>
#include <iterator>
#include <map>
#include <list>
#include <set>
#include <time.h>
#include <algorithm>
#include <random>
#include "dep\single_include\nlohmann\json.hpp"

using namespace std;
using json = nlohmann::json;

struct node
{
    string name;
    int standart;
    int secondary;
};

void to_json(json &j, const node &t)
{
    j = json>{"name", t.name}, {"standart", t.standart}, {"secondary", t.secondary}};
}

void from_json(json const &j, node &t)
{
    j.at("name").get_to(t.name);
    j.at("standart").get_to(t.standart);
    j.at("secondary").get_to(t.secondary);
}

void tokenize(string const &, const char, set<string> &);
void createDictionary(string const &, const char, map<char, int> &);

int GetRand(map<char, int> &);

int main()
{
    setlocale(LC_ALL, "Rus");
    srand(time(0));
    string s;
    list<node> spisok, newSpisok, spisokFile;
    set<string> words;
    map<char, int> dictionary;
    string str;
    node info, x, y, z;
    const char razdel = '-';
    int i, j, pr, n;
    json js{}, jf;
```

```

ifstream in("test.txt");
if (in.is_open())
{
    getline(in, str);
}
else
{
    cout << "Файл не найден" << endl;
}
in.close();
cout << "Множество" << endl;
tokenize(str, razdel, words); // создаем множество из слов
файла
for (auto i : words)
    cout << i << endl;

system("pause");
cout << "Словарь" << endl;
createDictionary(str, razdel, dictionary); // создаем словарь
буква -количество в файле
for (const auto &elem : dictionary)
{
    cout << elem.first << " " << elem.second << endl;
}

system("pause"); // создаем список из данных множества и
словаря
cout << "Список" << endl;
map<char, int>::iterator item = dictionary.begin();
for (auto i : words)
{
    info.name = i; // name
    for (j = 0, pr = 1; j < 3; j++)
    {
        pr *= GetRand(dictionary);
    }
    info.standart = pr; // standart
    item = dictionary.find(i[0]);
    info.secondary = item->second * item->second; // secondary
    spisok.push_back(info);
}
for (auto i : spisok) // выводим полученный список
    cout << "Название: " << i.name << "; Основной параметр: "
<< i.standart << "; Вторичный параметр: " << i.secondary << endl;
system("pause");

// сортируем список по убыванию (Z to A)
cout << "Список по убыванию названия" << endl;
n = spisok.size();
node ar[n];
j = 0;
for (auto i : spisok)
    ar[j++] = i;
sort(ar, ar + n, [](const node &x, const node &y)
    { return x.name[0] > y.name[0]; });
spisok.clear();
for (j = 0; j < n; j++)
    spisok.push_back(ar[j]);

```

```

        /*spisok.sort([](const node &x, const node &y)
                    { return x.name[0] > y.name[0]; });*/
        for (auto i : spisok) // выводим список отсортированный по
убыванию
            cout << "Название: " << i.name << "; Основной параметр: "
<< i.standart << "; Вторичный параметр: " << i.secondory << endl;
            system("pause");

        // создаем новый список только со словами из нечетного
количества букв
        cout << "Список из слов состоящих из нечетного количества букв"
<< endl;
        copy_if(spisok.begin(), spisok.end(),
back_inserter(newSpisok), [](const node &x)
                    { return x.name.length() % 2 != 0; });
        for (auto i : newSpisok) // выводим список отсортированный по
убыванию и из слов из нечетного количества букв
            cout << "Название: " << i.name << "; Основной параметр: "
<< i.standart << "; Вторичный параметр: " << i.secondory << endl;
            system("pause");

        // записываем полученный список в json файл
        cout << "Запись в файл json" << endl;
        ofstream NewFile("file.json");
        for (auto i : newSpisok)
        {
            js.push_back(i);
        }
        NewFile << js;
        NewFile.close();
        system("pause");

        // читаем список из json файла
        cout << "Читаем из json в список" << endl;
        fstream file("file.json");
        file >> jf;
        file.close();
        for (auto i : jf)
        {
            spisokFile.push_back(i);
        }

        system("pause");
        cout << "Список из файла" << endl;
        for (auto i : spisokFile) // выводим список из файла
            cout << "Название: " << i.name << "; Основной параметр: "
<< i.standart << "; Вторичный параметр: " << i.secondory << endl;
            system("pause");

        // перетасовка списка
        cout << "Ператсовка списка" << endl;
        n = spisokFile.size();
        node arr[n];
        j = 0;
        for (auto i : spisokFile)
            arr[j++] = i;
        unsigned seed = std::chrono::system_clock::now()

```

```

        .time_since_epoch()
        .count();
shuffle(arr, arr + n, default_random_engine(seed));
spisokFile.clear();
for (j = 0; j < n; j++)
    spisokFile.push_back(arr[j]);

for (auto i : spisokFile) // выводим перетасованный список
    cout
        << "Название: " << i.name << "; Основной параметр: "
<< i.standart << "; Вторичный параметр: " << i.secondory << endl;
    return 0;
}

int CountSymbol(string s, char symbol)
{
    int i, n = s.length(), count = 0;
    for (i = 0; i < n; i++)
        if (s[i] == symbol)
            count++;
    return count;
}

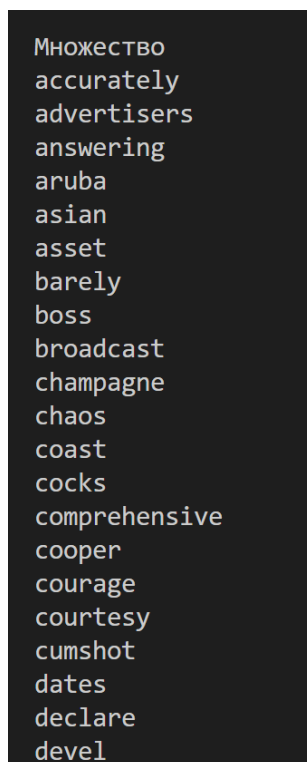
void tokenize(string const &str, const char delim, set<string>
&out)
{
    // строим поток из строки
    stringstream ss(str);
    string s;
    while (getline(ss, s, delim))
        out.insert(s);
}

void createDictionary(string const &str, const char delim,
map<char, int> &out)
{
    stringstream ss(str);
    string word, s;
    char symbol;
    while (getline(ss, word, delim))
        s.append(word);
    int i;
    for (i = 97; i < 123; i++)
    {
        symbol = static_cast<char>(i);
        out.insert(pair<char, int>(symbol, CountSymbol(s,
symbol)));
    }
}

int GetRand(map<char, int> &dictionary)
{
    int j, pr;
    map<char, int>::iterator item = dictionary.begin();
    int random = rand() % dictionary.size();
    std::advance(item, random);
    return item->second;
}

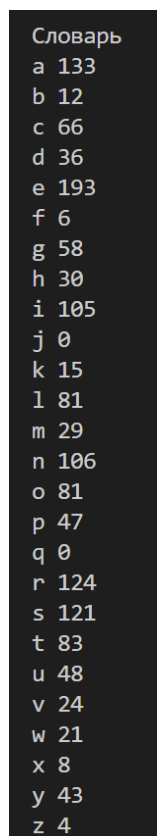
```

Скриншоты



Множество
accurately
advertisers
answering
aruba
asian
asset
barely
boss
broadcast
champagne
chaos
coast
cocks
comprehensive
cooper
courage
courtesy
cumshot
dates
declare
devel

Рисунок 1 – Множество слов из файла



Словарь
a 133
b 12
c 66
d 36
e 193
f 6
g 58
h 30
i 105
j 0
k 15
l 81
m 29
n 106
o 81
p 47
q 0
r 124
s 121
t 83
u 48
v 24
w 21
x 8
y 43
z 4

Рисунок 2 – Словарь из букв и количества их использования в файле

```

Список
Название: accurately; Основной параметр: 2895772; Вторичный параметр: 17689
Название: advertisers; Основной параметр: 0; Вторичный параметр: 17689
Название: answering; Основной параметр: 0; Вторичный параметр: 17689
Название: aruba; Основной параметр: 70470; Вторичный параметр: 17689
Название: asian; Основной параметр: 1388056; Вторичный параметр: 17689
Название: asset; Основной параметр: 34020; Вторичный параметр: 17689
Название: barely; Основной параметр: 353220; Вторичный параметр: 144
Название: boss; Основной параметр: 9024; Вторичный параметр: 144
Название: broadcast; Основной параметр: 1938299; Вторичный параметр: 144
Название: champagne; Основной параметр: 5040; Вторичный параметр: 4356
Название: chaos; Основной параметр: 0; Вторичный параметр: 4356
Название: coast; Основной параметр: 46440; Вторичный параметр: 4356
Название: cocks; Основной параметр: 0; Вторичный параметр: 4356
Название: comprehensive; Основной параметр: 0; Вторичный параметр: 4356
Название: cooper; Основной параметр: 250604; Вторичный параметр: 4356
Название: courage; Основной параметр: 473184; Вторичный параметр: 4356
Название: courtesy; Основной параметр: 0; Вторичный параметр: 4356
Название: cumshot; Основной параметр: 0; Вторичный параметр: 4356
Название: dates; Основной параметр: 180048; Вторичный параметр: 1296
Название: declare; Основной параметр: 5040; Вторичный параметр: 1296
Название: devel; Основной параметр: 28188; Вторичный параметр: 1296
Название: deviation; Основной параметр: 0; Вторичный параметр: 1296
Название: dust; Основной параметр: 20352; Вторичный параметр: 1296
Название: encouraged; Основной параметр: 0; Вторичный параметр: 37249
Название: entry; Основной параметр: 191664; Вторичный параметр: 37249
Название: exec; Основной параметр: 155034; Вторичный параметр: 37249

```

Рисунок 3 – Вывод составленного списка

```

Список по убыванию названия
Название: younger; Основной параметр: 933394; Вторичный параметр: 1849
Название: xnxx; Основной параметр: 11352; Вторичный параметр: 64
Название: with; Основной параметр: 0; Вторичный параметр: 441
Название: welding; Основной параметр: 3870; Вторичный параметр: 441
Название: weather; Основной параметр: 167184; Вторичный параметр: 441
Название: washer; Основной параметр: 0; Вторичный параметр: 441
Название: villages; Основной параметр: 0; Вторичный параметр: 576
Название: variety; Основной параметр: 58320; Вторичный параметр: 576
Название: twist; Основной параметр: 106848; Вторичный параметр: 6889
Название: taking; Основной параметр: 46284; Вторичный параметр: 6889
Название: tissue; Основной параметр: 17856; Вторичный параметр: 6889
Название: towards; Основной параметр: 0; Вторичный параметр: 6889
Название: strings; Основной параметр: 133920; Вторичный параметр: 14641
Название: saints; Основной параметр: 14094; Вторичный параметр: 14641
Название: satisfy; Основной параметр: 13920; Вторичный параметр: 14641
Название: scripts; Основной параметр: 333900; Вторичный параметр: 14641
Название: seal; Основной параметр: 0; Вторичный параметр: 14641
Название: sexual; Основной параметр: 5808; Вторичный параметр: 14641
Название: shelf; Основной параметр: 105270; Вторичный параметр: 14641
Название: sizes; Основной параметр: 62532; Вторичный параметр: 14641
Название: started; Основной параметр: 0; Вторичный параметр: 14641
Название: stephen; Основной параметр: 0; Вторичный параметр: 14641
Название: swim; Основной параметр: 22704; Вторичный параметр: 14641
Название: remember; Основной параметр: 729540; Вторичный параметр: 15376
Название: reaction; Основной параметр: 0; Вторичный параметр: 15376

```

Рисунок 4 – Вывод списка отсортированного по убыванию названия


```

Список из слов состоящих из нечетного количества букв
Название: younger; Основной параметр: 933394; Вторичный параметр: 1849
Название: welding; Основной параметр: 3870; Вторичный параметр: 441
Название: weather; Основной параметр: 167184; Вторичный параметр: 441
Название: variety; Основной параметр: 58320; Вторичный параметр: 576
Название: twist; Основной параметр: 106848; Вторичный параметр: 6889
Название: towards; Основной параметр: 0; Вторичный параметр: 6889
Название: strings; Основной параметр: 133920; Вторичный параметр: 14641
Название: satisfy; Основной параметр: 13920; Вторичный параметр: 14641
Название: scripts; Основной параметр: 333900; Вторичный параметр: 14641
Название: shelf; Основной параметр: 105270; Вторичный параметр: 14641
Название: sizes; Основной параметр: 62532; Вторичный параметр: 14641
Название: started; Основной параметр: 0; Вторичный параметр: 14641
Название: stephen; Основной параметр: 0; Вторичный параметр: 14641
Название: receptors; Основной параметр: 0; Вторичный параметр: 15376
Название: recycling; Основной параметр: 18232; Вторичный параметр: 15376
Название: religious; Основной параметр: 21576; Вторичный параметр: 15376
Название: ruled; Основной параметр: 6948; Вторичный параметр: 15376
Название: regular; Основной параметр: 35856; Вторичный параметр: 15376
Название: peers; Основной параметр: 0; Вторичный параметр: 2209
Название: preceding; Основной параметр: 35856; Вторичный параметр: 2209
Название: precipitation; Основной параметр: 87480; Вторичный параметр: 2209
Название: premium; Основной параметр: 12672; Вторичный параметр: 2209
Название: prominent; Основной параметр: 0; Вторичный параметр: 2209
Название: onion; Основной параметр: 139392; Вторичный параметр: 6561
Название: notices; Основной параметр: 32712; Вторичный параметр: 11236

```

Рисунок 5 – Список, созданный из слов, состоящих из нечетного количества букв

```

Запись в файл json
Для продолжения нажмите любую клавишу . . .
Читаем из json в список
Для продолжения нажмите любую клавишу . . .
Список из файла
Название: younger; Основной параметр: 933394; Вторичный параметр: 1849
Название: welding; Основной параметр: 3870; Вторичный параметр: 441
Название: weather; Основной параметр: 167184; Вторичный параметр: 441
Название: variety; Основной параметр: 58320; Вторичный параметр: 576
Название: twist; Основной параметр: 106848; Вторичный параметр: 6889
Название: towards; Основной параметр: 0; Вторичный параметр: 6889
Название: strings; Основной параметр: 133920; Вторичный параметр: 14641
Название: satisfy; Основной параметр: 13920; Вторичный параметр: 14641
Название: scripts; Основной параметр: 333900; Вторичный параметр: 14641
Название: shelf; Основной параметр: 105270; Вторичный параметр: 14641
Название: sizes; Основной параметр: 62532; Вторичный параметр: 14641
Название: started; Основной параметр: 0; Вторичный параметр: 14641
Название: stephen; Основной параметр: 0; Вторичный параметр: 14641
Название: receptors; Основной параметр: 0; Вторичный параметр: 15376
Название: recycling; Основной параметр: 18232; Вторичный параметр: 15376
Название: religious; Основной параметр: 21576; Вторичный параметр: 15376
Название: ruled; Основной параметр: 6948; Вторичный параметр: 15376
Название: regular; Основной параметр: 35856; Вторичный параметр: 15376
Название: peers; Основной параметр: 0; Вторичный параметр: 2209
Название: preceding; Основной параметр: 35856; Вторичный параметр: 2209
Название: precipitation; Основной параметр: 87480; Вторичный параметр: 2209
Название: premium; Основной параметр: 12672; Вторичный параметр: 2209

```

Рисунок 6 – Запись списка файла и вывод списка из файла

Ператсовка списка

Название: satisfy; Основной параметр: 13920; Вторичный параметр: 14641
Название: gentleman; Основной параметр: 729540; Вторичный параметр: 3364
Название: stephen; Основной параметр: 0; Вторичный параметр: 14641
Название: lease; Основной параметр: 133632; Вторичный параметр: 6561
Название: courage; Основной параметр: 473184; Вторичный параметр: 4356
Название: preceding; Основной параметр: 35856; Вторичный параметр: 2209
Название: precipitation; Основной параметр: 87480; Вторичный параметр: 2209
Название: sizes; Основной параметр: 62532; Вторичный параметр: 14641
Название: deviation; Основной параметр: 0; Вторичный параметр: 1296
Название: cocks; Основной параметр: 0; Вторичный параметр: 4356
Название: cumshot; Основной параметр: 0; Вторичный параметр: 4356
Название: strings; Основной параметр: 133920; Вторичный параметр: 14641
Название: receptors; Основной параметр: 0; Вторичный параметр: 15376
Название: coast; Основной параметр: 46440; Вторичный параметр: 4356
Название: scripts; Основной параметр: 333900; Вторичный параметр: 14641
Название: onion; Основной параметр: 139392; Вторичный параметр: 6561
Название: recycling; Основной параметр: 18232; Вторичный параметр: 15376
Название: weather; Основной параметр: 167184; Вторичный параметр: 441
Название: premium; Основной параметр: 12672; Вторичный параметр: 2209
Название: religious; Основной параметр: 21576; Вторичный параметр: 15376
Название: advertisers; Основной параметр: 0; Вторичный параметр: 17689
Название: naturally; Основной параметр: 267120; Вторичный параметр: 11236
Название: chaos; Основной параметр: 0; Вторичный параметр: 4356
Название: ruled; Основной параметр: 6948; Вторичный параметр: 15376
Название: prominent; Основной параметр: 0; Вторичный параметр: 2209
Название: younger; Основной параметр: 933394; Вторичный параметр: 1849

Рисунок 7 – Вывод перетасованного списка из файла