Лабораторная работа №6

Записать в двумерный массив размера 8Х8 случайные вещественные числа, значения которых от 0 до 100. Вывести полученный массив на экран и записать в файл в виде выровненной матрицы с двумя знаками после запятой.

#include<iostream>

#include<ctime>

#include<fstream>

#include<iomanip>

using namespace std;

int main() {

const int n = 8;

double array[n][n];

srand(time(NULL));

for (int i = 0; i < 8; i++) {

for (int j = 0; j < 8; j++) {

array[i][j] = (rand() % 10000) / 100.0;

}

}

ofstream file("first.txt");

for (int i = 0; i < 8; i++) {

for (int j = 0; j < 8; j++) {

cout << "\t" << fixed << setprecision(2)<< array[i][j]; file << "\t" << array[i][j];

}

cout << endl;

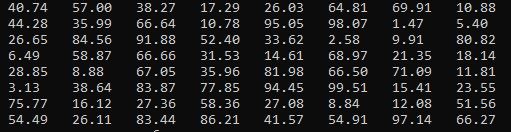
}

file.close();

system("pause");

return 0;

}



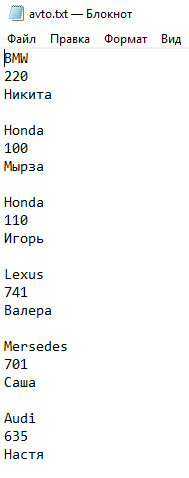


В файле есть сведения об автомобилях: марка автомобиля, номер и фамилия владельца.

а) Вывести сведения о владельцах и номерах автомобилей каждой марки автомобиля.

б) Подсчитать количество автомобилей каждой марки.

Так выглядит текстовый файл, из которого будет считываться информация



Код:

#include <iostream>

#include <fstream>

#include <cstring>

#include<iomanip>

using namespace std;

struct Car {

int number;

char model[50];

char surname[50];

};

int main() {

setlocale(LC\_ALL, "rus");

Car tmp;

string file = "avto.txt";

ifstream fin; // чтение

Car infoCar[50];

fin.open(file);

if (!fin)

{

cout << "Файл не открыт\n\n";

system("pause");

return 0;

}

else

{

char Bmw[] = "BMW";

int kolBmw = 0;

char Honda[] = "Honda";

int kolHonda = 0;

char Mersedes[] = "Mersedes";

int kolMersedes = 0;

char Lexus[] = "Lexus";

int kolLexus = 0;

char Audi[] = "Audi";

int kolAudi = 0;

cout << "Файл открыт" << endl;

for (int i = 0; i <= 5; i++)

{

fin >> infoCar[i].model;

cout << "Марка: " << infoCar[i].model << endl;

fin >> infoCar[i].number;

cout << "Номер: " << infoCar[i].number << endl;

fin >> infoCar[i].surname;

cout << "Владелец: " << infoCar[i].surname << endl;

if (strcmp(infoCar[i].model, Bmw) == 0) {

kolBmw++;

}

else if (strcmp(infoCar[i].model, Honda) == 0) {

kolHonda++;

}

else if (strcmp(infoCar[i].model, Mersedes) == 0) {

kolMersedes++;

}

else if (strcmp(infoCar[i].model, Lexus) == 0) {

kolLexus++;

}

else if (strcmp(infoCar[i].model, Audi) == 0) {

kolAudi++;

}

}

for (int i = 0; i <= 14; i++) {

for (int j = i + 1; j <= 14; j++) {

if (strcmp(infoCar[i].surname, infoCar[j].surname) > 0)

{

tmp = infoCar[i];

infoCar[i] = infoCar[j];

infoCar[j] = tmp;

}

}

}

cout << endl;

cout << "Марка Bmw: количество -> " << kolBmw << endl;

cout << "Марка Honda: количество -> " << kolHonda << endl;

cout << "Марка Lexus: количество -> " << kolLexus << endl;

cout << "Марка Audi: количество -> " << kolAudi << endl;

cout << "Количество ненайденных марок -> " << kolBmw+ kolHonda+ kolLexus+kolAudi << endl;

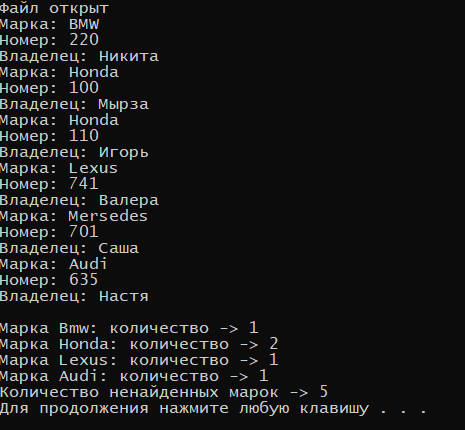
}

fin.close();

system("pause");

return 0;

}

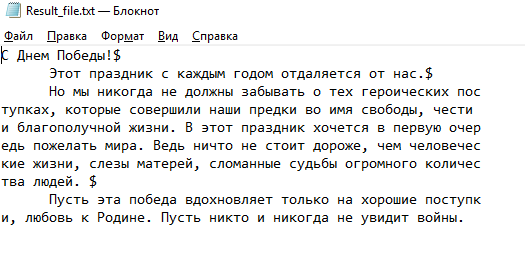
Результат на консоле

Текст записан одной длинной строкой. Признаком красной строки служит символ $. Переформатировать текст в 60-символьные строки, формируя абзацы. Исходный текст должен быть взят из файла, название которого будет введено с клавиатуры, а результирующий текст должен быть выведен в файл «Result\_file.txt».

Текст написанный в строку.



После запуска кода в папке результата:



#include <iostream>

#include <fstream>

#include <string>

using namespace std;

int main() {

string s, d;

ifstream a;

ofstream b;

int i = 0;

a.open("text.txt");

b.open("Result\_file.txt");

getline(a, s); // сравнивает строки между собой

int k = 0;

for (i = 0; i < s.length(); i++) {

if (k % 60 == 0) {

if (k != 0) {

d += "\n";

}

}

d += s[i];

if (s[i] == '$') {

d += "\n";

d += " ";

k = 4;

}

k++;

}

b << d;

a.close();

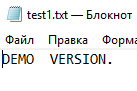
b.close();

system("pause");

}

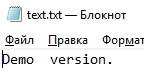
Текст, не содержащий собственных имён и сокращений, набран полностью прописными буквами. Заменить все прописные буквы, кроме букв, стоящих после точки, строчными буквами. Исходный текст должен быть взят из файла, название которого будет введено с клавиатуры, а результирующий текст должен быть выведен в файл «Result\_file.txt».

Текст из которого мы будем читать



При запуске мы вводим название этого блокнота.





#include <iostream>

#include <fstream>

#include<string>

using namespace std;

int main()

{

setlocale(LC\_ALL, "ru");

fstream f, f2;

string filename;

string line;

cout << "введите имя файла: ";

cin >> filename;

f.open(filename);

f2.open("text.txt");

getline(f, line);

int i = 1;

int x = line.size();

while (i < x) {

if (isalpha(line[i])) {

line[i] = tolower(line[i]);

}

i++;

}

i = 0;

while (i < x) {

line[0] = toupper(line[0]);

if (line[i] == '.') {

line[i + 1] = toupper(line[i + 1]);

}

f2 << line[i];

i++;

}

f.close();

system("pause");

return 0;

}

За стоянку до трех часов парковочный гараж запрашивает плату ми­нимум $2.00. В случае стоянки более трех часов гараж дополнительно запрашивает $0.50 за каждый полный или неполный час сверх трех часов. Максимальная плата за сутки составляет $10.00. Допустим, что никто не паркуется более, чем на сутки за раз. Напишите программу расчета и печати оплаты за парковку для каждого из трех  
клиентов, которые парковали свои автомобили вчера в этом гараже. Вы должны вводить длительность парковки для каждого клиента. Ваша программа должна печатать результаты в аккуратном табулированном формате и должна рассчитывать и печатать общий вчерашний доход.  
Программа должна использовать функцию calculateCharges, чтобы оп­ределять плату для каждого клиента. Результаты работы должны пред­ставляться в следующем формате:

Автомобиль Часы Плата

1 1.5 2.00

2 4.0 2.50

3 24.0 10.00  
 Итого 29.5 14.50

#include <iostream>

#include <fstream>

#include<string>

#include<iomanip>

using namespace std;

double calculateCharges(double time)

{

double sum;

if (time > 0 && time <= 3)

sum = 2.00;

else if (time > 3 && time < 24)

{

sum = ceil(time - 3) \* 0.50 + 2.00;

}

return sum;

}

int main() {

setlocale(LC\_ALL, "ru");

double time1, time2, time3, timeavto;

double sum1, sum2, sum3, timesum;

while (true)

{

cout << "ВРЕМЯ СТОЯНКИ " << endl;

cout << "время первой машины: ";

cin >> time1;

cout << "время второй машины: ";

cin >> time2;

cout << "время третей машины: ";

cin >> time3;

if (time1 < 0 || time1 > 24 || time2 < 0 || time2 > 24 || time3 < 0 || time3 > 24)

{

cout << "Ошибка!!!" << endl;

return 0;

}

break;

}

timeavto = time1 + time2 + time3;

sum1 = calculateCharges(time1);

sum2 = calculateCharges(time2);

sum3 = calculateCharges(time3);

timesum = sum1 + sum2 + sum3;

cout << setw(10) << "Авто" <<setw(10) << "Часы" << setw(10) << "Плата" << endl;

cout << setw(10) << "1" << setw(10) << time1 << setw(10) << sum1 << endl;

cout << setw(10) << "2" << setw(10) << time2 << setw(10) << sum2 << endl;

cout << setw(10) << "3" << setw(10) << time3 << setw(10) << sum3 << endl;

cout << setw(10) << "Итого: " << setw(10) << timeavto << setw(10) << timesum << endl;

return 0;

}

