// Item и вспомагательные функции

#pragma once

#include <iostream>

#include <string>

#include <iomanip>

struct Item {

std::string

nick = "",

course = "",

group = "";

int gradebook\_id,

fcode,

dpcode,

scode,

mark;

void SetMark(std::string s) {

this->mark = atoi(s.c\_str());

}

void SetBGID(std::string s) {

this->gradebook\_id = atoi(s.c\_str());

}

void SetFcode(std::string s) {

this->fcode = atoi(s.c\_str());

}

void SetDpCode(std::string s) {

this->dpcode = atoi(s.c\_str());

}

void SetScode(std::string s) {

this->scode = atoi(s.c\_str());

}

};

using namespace std;

void header() {

cout << "|" << "Код факультету"

<< "|" << "Код спеціальності"

<< "|" << "Курс"

<< "|" << "Група"

<< "|" << "Код дисципліни"

<< "|" << "Прізвище"

<< "|" << "Ном. з/к"

<< "|" << "Оцінка" << "|" << endl;

}

void PrintAll(const vector<Item> v) {

cout << endl;

header();

for (size\_t i = 0; i < v.size(); i++) {

auto p = v[i];

cout << " [" << i + 1 << "] " <<

"|" << setw(5) << p.fcode << setw(5) << "|" << setw(5) <<

setw(5) << p.scode << setw(5) << "|" << setw(5) <<

setw(5) << p.course << setw(5) << "|" << setw(5) <<

setw(10) << p.group << setw(5) << "|" << setw(5) <<

setw(5) << p.dpcode << setw(5) << "|" << setw(5) <<

setw(10) << p.nick << setw(5) << "|" << setw(5) <<

setw(5) << p.gradebook\_id << setw(5) << "|" << setw(5) <<

setw(5) << p.mark << "|" << endl;

} cout << endl;

}

// продолжение Item и вспомагательные функции

vector<Item>\* ReadFile(string file) {

vector<Item> \*vec = new vector<Item>();

ifstream ins(file.c\_str(), ios::in);

if (ins.is\_open()) {

char\*buff = new char[256]; string trash = "";

while (!ins.eof()) {

ins.getline(buff, 256);

if (buff[0] != '\0') {

int j = 0;

Item it;

string tmp = "";

while (buff[j] != '|') { // fcode

tmp += buff[j++];

}j++;

it.SetFcode(tmp); tmp = "";

while (buff[j] != '|') { //spcode

tmp += buff[j++];

}j++;

it.SetScode(tmp); tmp = "";

while (buff[j] != '|') { //course

it.course += buff[j++];

}j++;

while (buff[j] != '|') {//group

it.group += buff[j++];

}j++;

while (buff[j] != '|') {//dpcode

tmp += buff[j++];

}j++;

it.SetDpCode(tmp); tmp = "";

while (buff[j] != '|') {//nick

it.nick += buff[j++];

}j++;

while (buff[j] != '|') {//gb\_id

tmp += buff[j++];

}j++;

it.SetBGID(tmp); tmp = "";

while (true) { // mark

if (buff[j] == '\0') break;

tmp += buff[j++];

}

it.SetMark(tmp);

vec->push\_back(it);

}

} ins.close();

return vec;

}

return nullptr;

}

void WriteItems(string file, const vector<Item> v) {

ofstream of(file, ios::trunc);

cout << endl << "Відкриваємо файл для запису...";

for (size\_t i = 0; i < v.size(); i++) {

auto p = v[i];

of <<

p.fcode << "|" <<

p.scode << "|" <<

p.course << "|" <<

p.group << "|" <<

p.dpcode << "|" <<

p.nick << "|" <<

p.gradebook\_id << "|" <<

p.mark << endl;

} cout << endl << "Файл успішно збережено! " << endl;

of.close(); }

// Задача 1

#include <iostream>

#include <vector>

#include <iomanip>

#include <fstream>

#include <Windows.h>

#include <conio.h>

#include "Item.h"

using namespace std;

bool contains(const vector<int> v, int s) {

for (size\_t i = 0; i < v.size(); i++)

if (v[i] == s) return true;

return false;

}

void Calc(vector<Item> v) {

vector<int> made;

vector<Item> sum, res = v;

made.push\_back(v.back().fcode);

cout << "|\t\t | Питома вага |" << endl;

cout << "| Факультет\t" << " | 2 3 4 5 |" << endl;

for (size\_t i = 0; i < v.size(); i++) {

int lsum = 0, cnt = 0, resf;

int \*wmark = new int[6];

for (int i = 2; i <= 5; i++) wmark[i] = 0;

for (size\_t i = 0; i < v.size(); i++) {

if (!contains(made, v[i].fcode)) continue;

if (v[i].fcode == made.back()) {

lsum += v[i].mark; // aver mark

cnt++;

//weight

++wmark[v[i].mark];

}

}

cout << "| Факультет: " << setw(5) << made.back();

for (int i = 2; i <= 5; i++) {

if (wmark[i] != 0) {

cout << " | " << setw(1) << setprecision(2) << (1.0\*wmark[i] / cnt \* 100);

}

else cout << " | " << 0;

}

cout << " | Середній бал: " << lsum / cnt << endl;

resf = v.back().fcode;

while (contains(made, resf)) {

if (v.size() > 0) {

v.pop\_back();

if (v.size() > 0)

resf = v.back().fcode;

}

else break;

}

if (v.size() > 0)

made.push\_back(v.back().fcode);

}

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

//задача 1

auto f1 = ReadFile("./data.txt");

Calc(\*f1);

\_getch();

return 0;

}

// Задача 2

#include <iostream>

#include <Windows.h>

#include <vector>

#include <fstream>

#include <iomanip>

#include <conio.h>

#include "Item.h"

using namespace std;

//Задача 2

void RemoveItems(vector<Item> \*v) {

while (true) {

PrintAll(\*v);

cout << endl << "Номер запису для видалення (-1 відміна): " << endl;

int x; cin >> x; cin.get();

if (x == -1) break;

v->erase(v->begin() + x - 1);

cout << endl << "Видалити ще запис [1/0]: ";

int fg; cin >> fg;

if (fg != 1) break;

}

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

//задача 2

auto f2 = ReadFile("./data.txt");

RemoveItems(f2);

WriteItems("./data.txt", \*f2);

\_getch();

return 0;

}

//Задача 3

#include <iostream>

#include <Windows.h>

#include <vector>

#include <conio.h>

#include <fstream>

#include <iomanip>

#include "Item.h"

using namespace std;

//Задача 3

vector<Item>\* GetItems() {

vector<Item> \*v = new vector<Item>();

while (true)

{

Item it;

cout << "Введіть код факультету: ";

cin >> it.fcode; cin.get();

cout << "Введіть код спеціальності: ";

cin >> it.scode; cin.get();

cout << "Введіть курс: ";

cin >> it.course;

cout << "Введіть групу: ";

cin >> it.group;

cout << "Введіть код дисципліни: ";

cin >> it.dpcode; cin.get();

cout << "Введіть прізвище студента: ";

cin >> it.nick;

cout << "Введіть номер залікової книжки: ";

cin >> it.gradebook\_id; cin.get();

cout << "Введіть оцінку: ";

cin >> it.mark; cin.get();

v->push\_back(it);

cout << endl << "Додати ще запис [1/0]: ";

int fg; cin >> fg;

if (fg != 1) break;

} return v;

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

//задача 3

auto f3 = GetItems();

PrintAll(\*f3);

WriteItems("./data.txt", \*f3);

\_getch();

return 0;

}

//Задача 4

#include <iostream>

#include <Windows.h>

#include <vector>

#include <conio.h>

#include <fstream>

#include <iomanip>

#include "Item.h"

using namespace std;

void CorrectPrint(const Item p) {

cout <<"\n| [1]" << setw(5) << p.fcode << setw(5) << "| [2]" << setw(5) <<

setw(5) << p.scode << setw(5) << "| [3]" << setw(5) <<

setw(5) << p.course << setw(5) << "| [4]" << setw(5) <<

setw(10) << p.group << setw(5) << "| [5]" << setw(5) <<

setw(5) << p.dpcode << setw(5) << "| [6]" << setw(5) <<

setw(10) << p.nick << setw(5) << "| [7]" << setw(5) <<

setw(5) << p.gradebook\_id << setw(5) << "| [8]" << setw(5) <<

setw(5) << p.mark << "|" << endl;

}

void CorrectItems(vector<Item> \*v) {

while (true)

{

PrintAll(\*v);

cout << "Введіть номер запису для коригування (-1 відміна): ";

int x, y; cin >> x; cin.get();

if (x == -1) break;

Item tmp = v->at(x - 1);

while (true)

{

cout << "Введіть номер поля для коригування (-1 відміна): ";

CorrectPrint(tmp);

cin >> y; cin.get();

if (x == -1) break;

cout << ">> Редагуємо ";

switch (y)

{

case 1: {cout << "код факультету: "; cin >> tmp.fcode; cin.get();

break;

}

case 2: {cout << "код спеціальності: "; cin >> tmp.scode; cin.get();

break;

}

case 3: {cout << "курс: "; cin >> tmp.course; break;

}

case 4: {cout << "групу: "; cin >> tmp.group; break;

}

case 5: {cout << "код дисципліни: "; cin >> tmp.dpcode; cin.get();

break;

}

case 6: {cout << "прізвище студента: "; cin >> tmp.nick;

break;

}

case 7: {cout << "номер зал.кн.: "; cin >> tmp.gradebook\_id; cin.get();

break;

}

case 8: {cout << "оцінку: "; cin >> tmp.mark; cin.get();

break;

}

default:

break;

}

CorrectPrint(tmp);

cout << endl << "Скорегувати ще поле [1/0]: ";

int fg; cin >> fg; if (fg != 1) break;

}

v->at(x - 1) = tmp;

cout << endl << "Скорегувати ще запис [1/0]: ";

int fg; cin >> fg; if (fg != 1) break;

}

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

//задача 4

auto f4 = ReadFile("./data.txt");

CorrectItems(f4);

WriteItems("./data.txt", \*f4);

\_getch();

return 0;

}

//Задача 5

#include <iostream>

#include <conio.h>

#include <iomanip>

#include <Windows.h>

using namespace std;

//Задача 5

void Min(int\*\*m, int sizeN) {

int index;

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

int min = MAXINT;

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

if (m[i][j] < min) {

min = m[i][j];

index = j;

}

}

m[i][index] = m[i][i];

m[i][i] = min;

}

}

void TestMaxrix() {

int n = 5;

int \*\*m = new int\*[5];

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

m[i] = new int[n];

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

m[i][j] = rand() % 10 + 1;

cout << m[i][j] << " ";

}cout << endl;

}cout << endl;

Min(m, n);

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

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

cout << m[i][j] << " ";

}cout << endl;

}

\_getch();

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

//задача 5

TestMaxrix();

\_getch();

return 0;

}