МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра защиты информации

**

**ОТЧЁТ**

**по практической работе №3**

**«**Итераторы**»**

**по дисциплине: «***Программирование***»**

Выполнил:Проверил:

Студент гр. «АБс-324», «АВТФ» *Доцент кафедры ЗИ*

*Клаус В. А. Архипова А. Б.*

«\_\_» \_\_\_\_\_\_ 2024 г.«\_\_\_» \_\_\_\_\_\_ 2024 г.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись) (подпись)

Новосибирск 2024

**Цели и задачи работы:** изучение итераторов и алгоритмов, методов разработки эффективных алгоритмов.

# Задания:

1. Преподаватель заполняет статистику об успеваемости студентов. За семестр было проведено 3 контрольные работы. Вывести на экран информацию о студентах, имеющих долг по данной дисциплине в биткоинах. (1 пропуск - 10 биткоинов). На вход подается N количество студентов, строка(содержащая группу и фамилию студента), кол-во пропущенных контрольных.
2. Кошка разодрала журнал потока на маленькие кусочки, помогите Анастасии Борисовне собрать журнал заново.
3. Студент инициализировал вектор строк с фруктами. Его одногруппник обиделся на него, из-за того, что студент съел все булочки в буфете. За это он испортил его вектор, заменив элементы символами. Смоделируйте данную ситуацию.
4. Преподаватель играет в игру со студентами “Угадай оценку”. Преподаватель оценивает работу студента по 5-бальной шкале, задача студента угадать оценку. Если студент угадывает, то преподаватель ставит эту оценку, увеличенную на единицу(Если оценка 5, то независимо от выбора ученика, оценка будет 5. Если оценка 1, то независимо от выбора ученика, его отправят на пересдачу). Если не угадывает, то ставит эту оценку, уменьшенную на единицу.
5. В гараже у автолюбителя находится множество машин различных марок. Узнайте уникальные марки машин в его гараже.
6. Проводится занятие по физкультуре. Студенты сдают норматив бега на 1 км. Найти самого быстрого студента, который отправится бегать на олимпийские игры.
7. Преподаватель обновляет журнал студентов после завершения сессии. Уберите отчисленных студентов из журнала.
8. Напишите базу данных буфета НГТУ. Пользователь может добавить товар на прилавок, удалить товар(продать) и вывести информацию(Продаваемый товар на данный момент, проданный товар).
9. Помогите Анастасии Борисовне найти студентов, которые не посещали ночь музеев.
10. Проводится лекция по программированию вместо теории вероятности. Преподаватель закрывает дверь и отмечает в журнале студентов, которые посетили пару. Вывести на экран студентов, которые будут отчислены за непосещение пары. На вход подается кол-во студентов N, строка(содержащая группу и фамилию студента), присутствовал на паре или нет.

**Решения**

## Задание 1:

**#include <iostream>**

**#include <vector>**

**#include <windows.h>**

**using namespace std;**

**struct Student {**

**string group;**

**string surname;**

**int propusks;**

**};**

**void info(vector<Student> students) { // ф-ия вывода должников**

**int count = 0;**

**cout << endl << "Должники: " << endl;**

**for (auto it = students.begin(); it != students.end(); ++it) {**

**if (it->propusks > 0) {**

**cout << it->surname << " (" << it->propusks\*10 << " BTC)" << endl;**

**count++;**

**}**

**}**

**if (count == 0) {**

**cout << "Нет таких, все зайки!" << endl;**

**}**

**}**

**int main(){**

**SetConsoleCP(1251);**

**SetConsoleOutputCP(1251);**

**Student st;**

**int n;**

**cout << "Введите кол-во студентов: ";**

**cin >> n;**

**vector<Student> students(n);**

**for (auto it = students.begin(); it != students.end(); ++it) {**

**cout << "Введите группу и фамилию студента: ";**

**cin >> it->group >> it->surname;**

**cout << "Введите кол-во пропущенных кнтр.работ студента " << it->surname << ": ";**

**while (true) {**

**cin >> it->propusks;**

**if (it->propusks < 0) { // обработка ошибки ввода пропусков**

**cout << "Ошибка, вы не можете вводить отрицательное значение! Выберите число от 0 до 3: ";**

**} else if (it->propusks > 3) {**

**cout << "Ошибка, введено слишком много пропусков! Выберите число от 0 до 3: ";**

**} else {**

**break;**

**}**

**}**

**}**

**info(students);**

**return 0;**

**}**

## Задание 2:

**#include <iostream>**

**#include <vector>**

**#include <algorithm>**

**using namespace std;**

**int main(){**

**setlocale(LC\_ALL, "Rus");**

**vector<pair<string, string>> jurnal = { // журнал по кусочкам**

**{"АБс-324", "Клаус"},**

**{"АБс-124", "Бурлаков"},**

**{"АБс-224", "Деревянкин"},**

**{"АБ-222", "Иванов"},**

**{"АБ-124", "Петров"},**

**{"АБс-324", "Арахнидов"}**

**};**

**sort(jurnal.begin(), jurnal.end()); // сортируем кусочки**

**cout << "Восстановленный журнал потока: " << endl;**

**for (auto it = jurnal.begin(); it != jurnal.end(); ++it) {**

**cout << it->first << " " << it->second << endl;**

**}**

**return 0;**

**}**

## Задание 3:

**#include <iostream>**

**#include <vector>**

**#include <algorithm>**

**using namespace std;**

**int main(){**

**setlocale(LC\_ALL, "Rus");**

**srand(time(0));**

**vector<string> fruits = { "Банан", "Апельсин", "Яблоко", "Груша", "Мандарин"}; // инициализация вектора студентом**

**fill(fruits.begin(), fruits.end(), "/\*@%^"); // одногруппник испортил вектор**

**cout << "Изменённый вектор: ";**

**for (auto it = fruits.begin(); it != fruits.end(); ++it) {**

**cout << \*it << " ";**

**}**

**return 0;**

**}**

## Задание 4:

**#include <iostream>**

**#include <vector>**

**using namespace std;**

**int main(){**

**setlocale(LC\_ALL, "Rus");**

**srand(time(0));**

**int n;**

**cout << "Введите кол-во студентов: ";**

**cin >> n;**

**vector<int> estimation(n); // оценки**

**for (auto it = estimation.begin(); it != estimation.end(); ++it) { // преподаватель выставляет оценки студентам**

**\*it = 1 + rand() % 5;**

**}**

**for (auto it = estimation.begin(); it != estimation.end(); ++it) { // игра**

**int est;**

**cout << "Угадайте оценку: ";**

**cin >> est;**

**if (\*it == 5) {**

**cout << "У вас оценка 5, независимо от вашего выбора!" << endl;**

**} else if (\*it == 1) {**

**cout << "У вас оценка 1, независимо от вашего выбора вы отправляетесь на пересдачу!" << endl;**

**} else if (est == \*it) {**

**cout << "Вы угадали! Оценка повышена" << endl;**

**est++;**

**\*it = est;**

**} else if (est != \*it) {**

**cout << "Вы не угадали! Оценка понижена" << endl;**

**est = \*it;**

**est--;**

**\*it = est;**

**}**

**}**

**cout << "Оценки студентов: ";**

**for (auto it = estimation.begin(); it != estimation.end(); ++it) { // вывод итоговых оценок**

**cout << \*it << " ";**

**}**

**return 0;**

**}**

## Задание 5:

**#include <iostream>**

**#include <vector>**

**#include <algorithm>**

**using namespace std;**

**int main(){**

**setlocale(LC\_ALL, "Rus");**

**vector<string> garage = { {"BMW", "Mercedes", "BMW", "Audi", "Audi", "Hummer"} }; // гараж автолюбителя**

**sort(garage.begin(), garage.end()); // сортируем контейнер**

**garage.erase(unique(garage.begin(), garage.end()), garage.end()); // перемещаем дубликаты в конец вектора и удаляем их**

**for (auto it = garage.begin(); it != garage.end(); ++it) { // вывод уникальных**

**cout << \*it << " ";**

**}**

**return 0;**

**}**

## Задание 6:

**#include <iostream>**

**#include <vector>**

**#include <algorithm>**

**using namespace std;**

**int main(){**

**setlocale(LC\_ALL, "Rus");**

**vector<pair<string, int>> statistika = { {"Иванов", 70}, {"Петров", 54}, {"Сидоров", 98} };**

**vector<int> time;**

**for (auto it = statistika.begin(); it != statistika.end(); ++it) {**

**time.push\_back(it->second);**

**}**

**for (auto it = statistika.begin(); it != statistika.end(); ++it) {**

**if (it->second == \*min\_element(time.begin(), time.end())) {**

**cout << "На олимпиаду поедет: " << it->first;**

**}**

**}**

**return 0;**

**}**

## Задание 7:

**#include <iostream>**

**#include <vector>**

**#include <algorithm>**

**using namespace std;**

**struct Student {**

**string name;**

**bool check;**

**};**

**int main(){**

**setlocale(LC\_ALL, "Rus");**

**vector<Student> journal = { {"Иванов", false}, {"Петров", true}, {"Клаус", false}, {"Смыков", true}};**

**auto newEnd = remove\_if(journal.begin(), journal.end(), [](const Student& student) { return student.check; });**

**journal.erase(newEnd, journal.end());**

**for (auto it = journal.begin(); it != journal.end(); ++it) {**

**cout << it->name << " ";**

**}**

**return 0;**

**}**

## Задание 8:

**#include <iostream>**

**#include <vector>**

**using namespace std;**

**struct Tovar {**

**string name;**

**int quantity;**

**};**

**class Buffet {**

**private:**

**vector<Tovar> buffetik;**

**vector<Tovar> prodaja;**

**public:**

**void add(string nazvanie, int kolvo) {**

**bool check = false;**

**for (auto it = buffetik.begin(); it != buffetik.end(); ++it) {**

**if (it->name == nazvanie) { // если такой товар есть, то плюсуем кол-во**

**it->quantity += kolvo;**

**check = true;**

**}**

**}**

**if (!check) { // если нету, то добавляем структуру в целом в вектор**

**buffetik.push\_back({nazvanie, kolvo});**

**}**

**}**

**void del(string nazvanie, int kolvo) {**

**if (buffetik.empty()) { // Если буфет пустой, то выводим соответствующие сообщение**

**cout << "Буфет пуст!" << endl;**

**} else {**

**bool check = false;**

**for (auto it = buffetik.begin(); it != buffetik.end(); ++it) {**

**if (it->name == nazvanie && it->quantity - kolvo < 0) { // если не хватает товара для продажи, выводим сообщение**

**cout << "Ошибка, не хватает товара для продажи!" << endl;**

**} else if (it->name == nazvanie) { // Если все хватает, то производим продажу**

**it->quantity -= kolvo;**

**check = true;**

**bool check\_in\_prodaja = false;**

**for (auto iter = prodaja.begin(); iter != prodaja.end(); ++iter) { // записываем продажу в вектор**

**if (iter->name == nazvanie) { // если такой товар есть, то плюсуем кол-во**

**iter->quantity += kolvo;**

**check\_in\_prodaja = true;**

**}**

**}**

**if (!check\_in\_prodaja) { // если нету, то добавляем структуру в целом в вектор**

**prodaja.push\_back({nazvanie, kolvo});**

**}**

**}**

**}**

**if (!check) {**

**cout << "Ошибка! Такого товара нет в продаже!" << endl;**

**}**

**}**

**}**

**void info() {**

**if (buffetik.empty()) {**

**cout << "Буфет пуст!" << endl;**

**} else {**

**cout << "Товар на прилавке: " << endl;**

**for (auto it = buffetik.begin(); it != buffetik.end(); ++it) {**

**cout << it->name << " " << it->quantity << " штук" << endl;**

**}**

**cout << "Проданный товар: " << endl;**

**for (auto it = prodaja.begin(); it != prodaja.end(); ++it) {**

**cout << it->name << " " << it->quantity << " штук" << endl;**

**}**

**}**

**}**

**};**

**int main(){**

**setlocale(LC\_ALL, "Rus");**

**Buffet bufet;**

**Tovar tovar;**

**while(true) {**

**string comand;**

**cout << "Выберите команду(add - добавить, del - удалить, info - вывести информацию, exit - выход): ";**

**cin >> comand;**

**if (comand == "add") {**

**cout << "Введите название товара и его кол-во: ";**

**cin >> tovar.name >> tovar.quantity;**

**bufet.add(tovar.name, tovar.quantity);**

**} else if (comand == "del") {**

**cout << "Введите название товара и его кол-во: ";**

**cin >> tovar.name >> tovar.quantity;**

**bufet.del(tovar.name, tovar.quantity);**

**} else if (comand == "info") {**

**bufet.info();**

**} else if (comand == "exit") {**

**cout << "Выход..." << endl;**

**break;**

**}**

**else {**

**cout << "Ошибка! Команда введена неверно" << endl;**

**}**

**}**

**return 0;**

**}**

## Задание 9:

**#include <iostream>**

**#include <vector>**

**#include <algorithm>**

**using namespace std;**

**int main(){**

**setlocale(LC\_ALL, "Rus");**

**vector<pair<string, bool>> students = { // вектор студентов**

**{"Клаус", true},**

**{"Бурлаков", false},**

**{"Петров", false},**

**};**

**for (auto it = students.begin(); it != students.end(); ++it) {**

**if (it->second == false) { // если не посещал, то выводим**

**cout << it->first << endl;**

**}**

**}**

**return 0;**

**}**

## Задание 10:

**#include <iostream>**

**#include <vector>**

**#include <limits>**

**#include <windows.h>**

**using namespace std;**

**struct Student {**

**string group;**

**string surname;**

**bool check;**

**};**

**void info(vector<Student> students) { // ф-ия вывода отчисленных студентов**

**int count = 0;**

**cout << endl << "Ликвидированы: " << endl;**

**for (auto it = students.begin(); it != students.end(); ++it) {**

**if (!it->check) {**

**cout << it->group << " " << it->surname << endl;**

**count++;**

**}**

**}**

**if (count == 0) {**

**cout << "Нет таких, все зайки!" << endl;**

**}**

**}**

**int main() {**

**SetConsoleCP(1251);**

**SetConsoleOutputCP(1251);**

**Student st;**

**int n;**

**cout << "Введите кол-во студентов: ";**

**cin >> n;**

**vector<Student> students(n);**

**for (auto it = students.begin(); it != students.end(); ++it) {**

**cout << "Введите группу и фамилию студента: ";**

**cin >> it->group >> it->surname;**

**while (true) {**

**cout << "Был ли студент " << it->surname << " на паре?(1 - да, 0 - нет): ";**

**cin >> it->check;**

**if (cin.fail() || (it->check != 0 && it->check != 1)) { // обработка ошибки ввода на bool**

**cin.clear(); // Сбрасываем флаг ошибки**

**cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // Очищаем буфер ввода**

**cout << "Ошибка! Введите корректное значение (1 - да, 0 - нет)." << endl;**

**} else {**

**break;**

**}**

**}**

**}**

**info(students);**

**return 0;**

**}**

**Вывод:**

В ходе выполнения практической работы №3 я познакомился с итераторами и алгоритмами, которые помогут мне в дальнейшем, при изучении C++.

Ссылка на репозиторий: [нажмите](https://github.com/JamesSevil/pract3.git)