Министерство образования и науки Российской Федерации  
Федеральное‌ ‌государственное‌ ‌бюджетное‌ ‌образовательное‌ ‌учреждение‌

высшего‌ ‌образования‌

**«Пермский национальный исследовательский  
политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЁТ**

**по лабораторной работе №18.7**

Дисциплина: «Информатика»

Тема: Шаблоны классов

Вариант 3

Выполнил:

Студент группы ИВТ-20-2б

Солдатов Алексей Павлович

Проверила:

Доцент кафедры ИТАС

Полякова О. А.

**Пермь, 2021**

**Цель задачи**

1) Создание консольного приложения, состоящего из нескольких файлов в системе программирования Visual Studio.

2) Реализация класса-контейнера.

Для достижения поставленной цели необходимо решить следующие задачи:

* Проанализировать теоретические сведения о ООП
* Абстрактные типы данных. Контейнеры
* Реализовать алгоритм поставленной задачи
* Создать отдельные файлы: Main.cpp для описания методов класса, Vector.cpp для описания главной функции, Vector.h для описания класса

Постановка задачи

Класс-контейнер ВЕКТОР с элементами типа int.

Реализовать операции: [] – доступа по индексу, + вектор – сложение элементов векторов a[i]+b[i], + число – добавляет константу ко всем элементам вектора.

Пользовательский класс Time для работы с временными интервалами. Интервал должен быть представлен в виде двух полей: минуты типа int и секунды типа int. При выводе минуты отделяются от секунд двоеточием.

Анализ задачи

1. Определить какие операции должны быть выполнены по заданию:

* Создание класса

class Vector

* Создание множества и заполнение класса
* Создание методов получения размера множества, пересечений и доступа по индексу
* Отчистка памяти

1. Для решения задачи используются переменные:

Несколько переменных целочисленного значения для ввода чисел и реализации меню

int s, con, in, menu = 4;

Указатель для работы с динамической памятью

1. Ввод данных осуществляется посредством функции

cin >> rub;

Вывод данных осуществляется посредством функции

cout << "Искомый элемент = " << a.give(in);

1. Создание метода класса , который ищет пересечения класса:

void Vector::cross()

{

int k = 0;

for (int i = 0; i < size - 1; i++)

{

for (int j = i + 1; j < size; j++)

{

if (data[i] == data[j])

k++;

}

if (k != 0)

cout << "\nЭлемент " << data[i] << " встречается " << k + 1 << " раз";

k = 0;

}

1. Создание методов класса для нахождения размера и доступа по индексу

int give(int index);

int givesize();

Создание контейнера

Vector::Vector(int s)

{

size = s;

min = new int[size];

sec = new int[size];

cout << "Введите элементы множества в формате 'mm ss': " << endl;

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

{

cin >> min[i] >> sec[i];

while (min[i] > 60 || sec[i] > 60)

{

cout << "Некорректный ввод!" << endl;

cin >> min[i] >> sec[i];

}

}

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

{

cout << "\n" << i + 1 << ". " << min[i] << ":" << sec[i];

}

}

Код

Main.cpp:

#include "Vector.h"

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

int s, con, in, menu = 4;

cout << "Введите кол-во элементов множества: ";

cin >> s;

Vector a(s);

while (menu == 1 || menu == 2 || menu == 3 || menu == 4)

{

cout << endl << "- - - - - - - - - - - - - - - " << endl << "Меню: \n\t0. Выход \n\t1. Получить элемент по индексу \n\t2. Сложение двух элементов \n\t3. Увеличение всех элементов множества на заданое колличество секунд(не более 60)";

cout << endl << "\nВыберите пункт меню : "; cin >> menu;

if (menu == 1)

{

cout << "Введи индекс элемента, который хочешь получить: ";

cin >> in;

a.dostup\_po\_index(in);

}

if (menu == 2)

{

int first, second;

cout << "Введите индексы элементов, которые нужно сложить: ";

cin >> first >> second;

a.sum(first, second);

}

if (menu == 3)

{

cout << "Введите кол-во секунд: ";

cin >> con;

while (con >= 60)

{

cout << "Введите колличество секунд не более 60: ";

cin >> con;

}

a.cons(con);

}

if (menu == 0)

cout << "Программа завершена";

}

}

Vector.cpp:

#include "Vector.h"

#include <iostream>

using namespace std;

Vector::Vector(int s)

{

size = s;

min = new int[size];

sec = new int[size];

cout << "Введите элементы множества в формате 'mm ss': " << endl;

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

{

cin >> min[i] >> sec[i];

while (min[i] > 60 || sec[i] > 60)

{

cout << "Некорректный ввод!" << endl;

cin >> min[i] >> sec[i];

}

}

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

{

cout << "\n" << i + 1 << ". " << min[i] << ":" << sec[i];

}

}

int Vector::dostup\_po\_index(int index)

{

if (index > size)

cout << "Ошибка ввода";

else

cout << "Искомый эллемент: " << min[index - 1] << ":" << sec[index - 1];

return 0;

}

int Vector::sum(int first, int second)

{

int cel, drob;

if (sec[first - 1] + sec[second - 1] >= 60)

{

cel = (sec[first - 1] + sec[second - 1]) / 60;

drob = (sec[first - 1] + sec[second -1]) - 60;

}

else

{

drob = sec[first - 1] + sec[second - 1];

cel = 0;

}

cout << "Сумма " << first << " элемента и " << second << " элемента = " << min[first - 1] + min[second - 1] + cel << ":" << drob;

return 0;

}

int Vector::cons(int c)

{

cout << "\nИзменённое множество: ";

int cel, drob;

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

{

if (sec[i] + c >= 60)

{

min[i] += (sec[i] + c) / 60;

sec[i] = (sec[i] + c) - 60;

}

else sec[i] += + c;

cout << "\n" << i + 1 << ". " << min[i] << ":" << sec[i];

}

return 0;

}

Vector.h:

class Vector

{

public:

Vector(int s);

int dostup\_po\_index(int index);

int sum(int first, int second);

int cons(int c);

int size;

int\* sec;

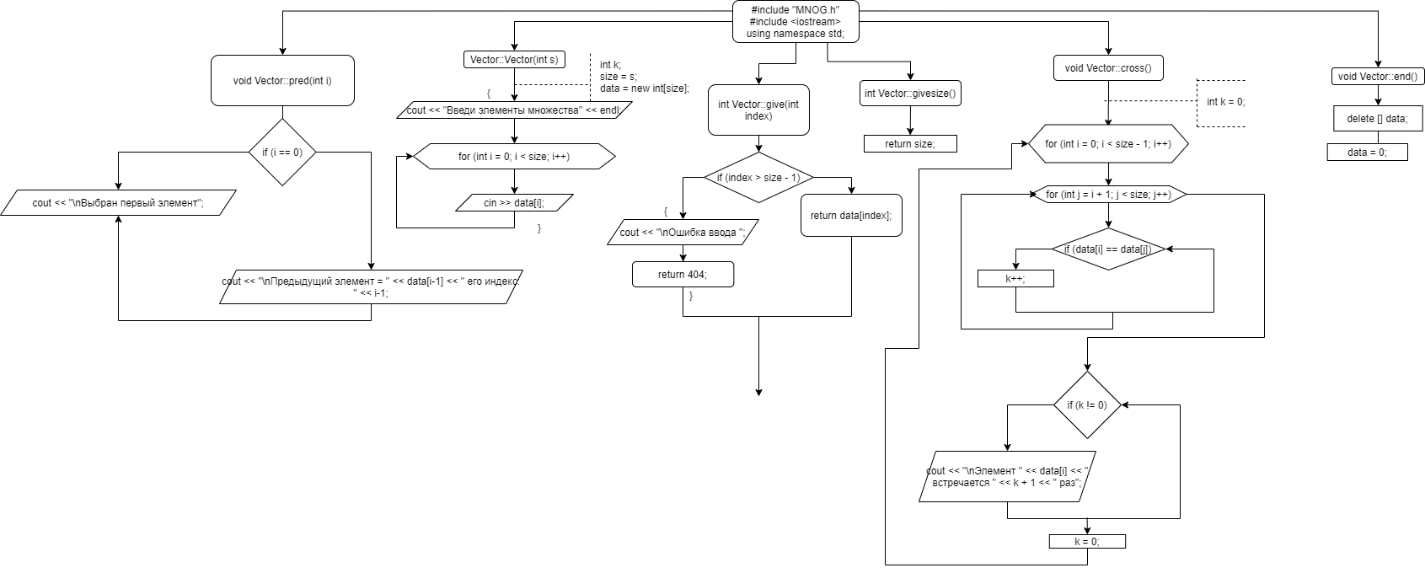
int\* min;

};

Блок-схема



Vector.cpp:



Vector.h:



Работа кода

