Министерство образования и науки Российской Федерации  
Федеральное‌ ‌государственное‌ ‌бюджетное‌ ‌образовательное‌ ‌учреждение‌

высшего‌ ‌образования‌

**«Пермский национальный исследовательский  
политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т**

**по лабораторной работе №18.13**

Дисциплина: «информатика»

Тема: Стандартные обобщенные алгоритмы библиотеки STL

Вариант 5

Выполнил работу

студент группы РИС-20-1б

Зверев А.Д.

Проверила

Доцент кафедры ИТАС

Полякова О.А.

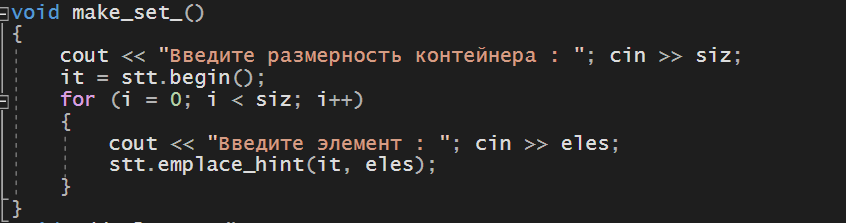
Пермь, 2021

**Постановка задачи**

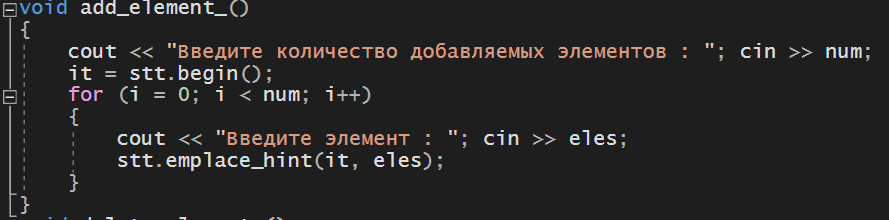
1. Создать последовательный контейнер.
2. Заполнить его элементами пользовательского типа (тип указан в варианте). Для пользовательского типа перегрузить необходимые операции.
3. Контейнер—список
4. Найти минимальный элемент
5. Найти элементы больше среднего арифметического и удалить их из контейнера
6. Каждый элемент домножить на максимальный элемент контейнера.

**Анализ задачи**

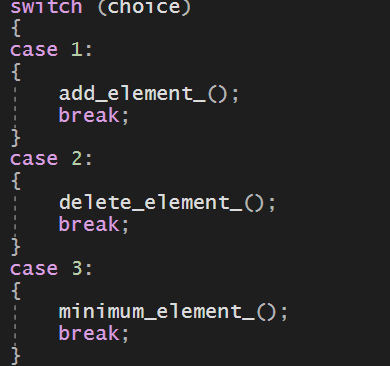
1. Для решения задачи необходимо…
   1. Написать функцию make\_set\_ типа void, в которой в программу заносится значения элементов с клавиатуры;



* 1. Написать функцию add\_element\_ типа void, которая позволяет добавлять в программу новые элементы;



* 1. Написать switch, который будет отвечать за корректность введенных данных с клавиатуры и вызывать различные функции;



1. В программе были использованы следующие типы данных:
   1. Тип данных int для хранения размерности контейнера и различных данных внутри программы;



* 1. Тип данных float для хранения других данных;



* 1. Тип данных void для различных функций;



**Решение**

#pragma once

#include<iostream>

#include<set>

using namespace std;

typedef set<float>sT;

set<float>::iterator it;

sT stt;

int siz = 0, i = 0, num = 0, choice = 0;

float eles, sum;

void make\_set\_()

{

cout << "Введите размерность контейнера : "; cin >> siz;

it = stt.begin();

for (i = 0; i < siz; i++)

{

cout << "Введите элемент : "; cin >> eles;

stt.emplace\_hint(it, eles);

}

}

void add\_element\_()

{

cout << "Введите количество добавляемых элементов : "; cin >> num;

it = stt.begin();

for (i = 0; i < num; i++)

{

cout << "Введите элемент : "; cin >> eles;

stt.emplace\_hint(it, eles);

}

}

void delete\_element\_()

{

cout << "Введите колво удаляемых : "; cin >> num;

for (i = 0; i < num; i++)

{

cout << "Введите элемент : "; cin >> eles;

it = stt.find(eles);

stt.erase(\*it);

}

}

void delete\_elements\_greater\_than\_middel\_()

{

for (auto st : stt)

{

sum += st;

}

sum = sum / stt.size();

it = stt.begin();

for (unsigned i = 0; i < stt.size(); i++)

{

if (\*it > sum)

{

stt.erase(\*it);

}

else

{

it++;

continue;

}

}

}

void minimum\_element\_()

{

it = stt.begin();

eles = \*it;

cout << "Минимум : " << eles << endl;

cout << "куда вы хотите добавить этот элемент? Введите число ";

int number;

cin >> number;

}

void multiply\_elements\_by\_maximum\_()

{

auto ite = stt.end();

ite--;

for (auto c : stt)

{

sum = c \* (\*ite);

cout << sum << " ";

}

cout << endl;

}

void print\_set\_()

{

cout << "Элементы : " << endl;

for (auto s : stt)

{

cout << s << " ";

}

cout << endl;

}

int main()

{

system("chcp 1251>nul");

make\_set\_();

while (true)

{

cout << endl;

cout << " 1: Добавление элементов \n"

<< " 2: Удаление элементов \n"

<< " 3: Минимальный элемент \n"

<< " 4: Удалить жлементы больше среднего значения \n"

<< " 5: Умножить элементы на максимум \n"

<< " 6: Печать \n"

<< " 7: Конец \n";

cout << endl;

cout << "Введите ваш выбор : "; cin >> choice;

switch (choice)

{

case 1:

{

add\_element\_();

break;

}

case 2:

{

delete\_element\_();

break;

}

case 3:

{

minimum\_element\_();

break;

}

case 4:

{

delete\_elements\_greater\_than\_middel\_();

break;

}

case 5:

{

multiply\_elements\_by\_maximum\_();

break;

}

case 6:

{

print\_set\_();

break;

}

case 7:

{

stt.clear();

system("pause");

return 0;

break;

}

default:

cout << "Ваш выбор некорректен \n";

break;

}

}

system("pause");

return 0;

}

**Скриншоты**

