МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

**«Вятский государственный университет»**

**(ФГБОУ ВО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Шаблоны и перегруженные функции

Отчет

Лабораторная работа №1 по дисциплине

«Технологии программирования»

Выполнил студент группы ИВТ-21 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Кудяшев Я.Ю./

Проверил преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Долженкова М.Л./

Киров 2021

**Задание**

Разработать консольное приложение, позволяющее высчитать седловые элементы заданной пользователем матрицы и обработать данные типа int и float, с использованием перегруженных функций и шаблонов. Обеспечить обработку некорректного ввода данных. Сравнить применимость шаблонных и перегруженных функций.

**Листинг программы**

#include <iostream>

#include <string>

using namespace std;

int a = 0;

int b = 0;

char st;

float\*\* B = new float\* [a];

int\*\* BB = new int\* [a];

int exceptionn(string aa) {

int v = 0;

int a = 0;

char inputt[255];

do {

for (int i = 0; i < 254; i++)

inputt[i] = 0;

v = 0;

getline(cin, aa);

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

if ((aa[i] == '1' || aa[i] == '2' || aa[i] == '3' || aa[i] == '4' || aa[i] == '5' || aa[i] == '6' || aa[i] == '7' || aa[i] == '8' || aa[i] == '9') && v == 0) {

inputt[v] = aa[i];

v++;

}

else if ((aa[i] == '0' || aa[i] == '1' || aa[i] == '2' || aa[i] == '3' || aa[i] == '4' || aa[i] == '5' || aa[i] == '6' || aa[i] == '7' || aa[i] == '8' || aa[i] == '9') && v != 0) {

inputt[v] = aa[i];

v++;

}

else

if (v > 0)

break;

}

if (inputt[0] == 0 && inputt[1] == 0 && inputt[2] == 0 && inputt[3] == 0)

a = 0;

else { aa = string() + inputt[0] + inputt[1] + inputt[2] + inputt[3];

a = stoi(aa);

}

if (a > 100)

cout << "No,no,no! Your value is too big. Enter it within 100.\n";

else

if (a == 0)

cout << "No,no,no! Invalid input! Enter an integer more then 0 and not an empty string.\n";

} while (a > 100 || a <= 0);

return a;

}

void matrix(int\*\* AA, int str, int stlb)

{

int\* minn = new int[str];

int\* maxx = new int[stlb];

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

{

minn[i] = INT\_MAX;

for (int j = 0; j < stlb; j++)

{

if (AA[i][j] < minn[i])

{

minn[i] = AA[i][j];

}

}

}

for (int j = 0; j < stlb; j++)

{

maxx[j] = INT\_MIN;

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

{

if (AA[i][j] > maxx[j])

{

maxx[j] = AA[i][j];

}

}

}

cout << '\n';

cout << "The endexes of the saddle points:" << endl;

int k = 0;

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

{

for (int j = 0; j < stlb; j++)

{

if (minn[i] == maxx[j])

{

cout << i << " " << j << endl;

k++;

}

}

}

if (!(k))

cout << "There are no saddle points." << endl;

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

{

delete[] AA[i];

}

delete[] AA;

delete[] minn;

delete[] maxx;

string gg;

getline(cin, gg);

}

void matrix(float\*\* A, int str, int stlb)

{

float\* min = new float[a];

float\* max = new float[b];

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

{

min[i] = INT\_MAX;

for (int j = 0; j < stlb; j++)

{

if (A[i][j] < min[i])

{

min[i] = A[i][j];

}

}

}

for (int j = 0; j < stlb; j++)

{

max[j] = INT\_MIN;

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

{

if (A[i][j] > max[j])

{

max[j] = A[i][j];

}

}

}

cout << '\n';

cout << "The endexes of the saddle points:" << endl;

int k = 0;

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

{

for (int j = 0; j < stlb; j++)

{

if (min[i] == max[j])

{

cout << i << " " << j << endl;

k++;

}

}

}

if (!(k))

cout << "There are no saddle points." << endl;

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

{

delete[] A[i];

}

delete[] A;

delete[] min;

delete[] max;

string gg;

getline(cin, gg);

}

template <typename T>

T shablon(T str, T stlb) {

float\* min = new float[str];

float\* max = new float[stlb];

float\*\* A = new float\* [str];

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

A[i] = new float[stlb];

int\* minn = new int[str];

int\* maxx = new int[stlb];

int\*\* AA = new int\* [str];

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

AA[i] = new int[stlb];

string aa;

char inputt[255];

int v = 0;

int i = 0;

int j = 0;

bool flag = false;

bool fflag = false;

cout << "What type of data do you prefer to use: int or float?\n";

cout << "Enter 'i' for int or 'f' for float.\n";

while (fflag==false) {

cin >> st;

if (st == 'i' || st == 'f') {

fflag = true;

}

else

cout << "Incorrect data type. Enter 'i' or 'f'.\n";

}

cout << "Enter the matrix"<<endl; //Ввод матрицы в зависимости от типа данных

if (st == 'i') {

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

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

do {

for (int c = 0; c < 254; c++)

inputt[c] = 0;

v = 0;

getline(cin,aa);

for (int ii = 0; ii < aa.length(); ii++) {

if (v == 0 && aa[ii] == '-' && (aa[ii + 1] == '1' || aa[ii + 1] == '2' || aa[ii + 1] == '3' || aa[ii + 1] == '4' || aa[ii + 1] == '5' || aa[ii + 1] == '6' || aa[ii + 1] == '7' || aa[ii + 1] == '8' || aa[ii + 1] == '9')) {

inputt[v] = aa[ii];

v++;

}

else

if ((aa[ii] == '1' || aa[ii] == '2' || aa[ii] == '3' || aa[ii] == '4' || aa[ii] == '5' || aa[ii] == '6' || aa[ii] == '7' || aa[ii] == '8' || aa[ii] == '9') && v == 0) {

inputt[v] = aa[ii];

v++;

}

else

if ((aa[ii] == '0' || aa[ii] == '1' || aa[ii] == '2' || aa[ii] == '3' || aa[ii] == '4' || aa[ii] == '5' || aa[ii] == '6' || aa[ii] == '7' || aa[ii] == '8' || aa[ii] == '9') && v != 0) {

inputt[v] = aa[ii];

v++;

}

else

if (aa[ii] == '0' && v == 0) {

inputt[v] = aa[ii];

break;

}

else

if (v > 0)

break;

}

if (inputt[0] == 0 && inputt[1] == 0 && inputt[2] == 0 && inputt[3] == 0 && inputt[4] == 0 && inputt[5]==0)

AA[i][j] = 20000;

else {

aa = string() + inputt[0] + inputt[1] + inputt[2] + inputt[3] + inputt[4] + inputt[5];

AA[i][j] = stoi(aa);

}

if (aa == "")

cout <<endl;

else

if (AA[i][j] > 10000 || AA[i][j] < -10000)

cout << "No,no,no! Invalid input. Enter a number between -10000 and 10000. The string must not be empty.\n";

} while (AA[i][j] < -10000 || AA[i][j] > 10000);

}

}

}

else {

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

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

do {

bool flag = false;

for (int c = 0; c < 254; c++)

inputt[c] = 0;

v = 0;

getline(cin, aa);

for (int ii = 0; ii < aa.length(); ii++) {

if (v == 0 && aa[ii] == '-' && (aa[ii + 1] == '1' || aa[ii + 1] == '2' || aa[ii + 1] == '3' || aa[ii + 1] == '4' || aa[ii + 1] == '5' || aa[ii + 1] == '6' || aa[ii + 1] == '7' || aa[ii + 1] == '8' || aa[ii + 1] == '9')) {

inputt[v] = aa[ii];

v++;

}

else

if (v == 0 && aa[ii] == '0' && aa[ii + 1] == '.') {

inputt[v] = aa[ii];

v++;

}

else

if (v == 0 && aa[ii] == '-' && aa[ii + 1] == '0' && aa[ii + 2] == '.') {

inputt[v] = aa[ii];

v++;

}

else

if ((aa[ii] == '1' || aa[ii] == '2' || aa[ii] == '3' || aa[ii] == '4' || aa[ii] == '5' || aa[ii] == '6' || aa[ii] == '7' || aa[ii] == '8' || aa[ii] == '9') && v == 0) {

inputt[v] = aa[ii];

v++;

}

else

if ((aa[ii] == '0' || aa[ii] == '1' || aa[ii] == '2' || aa[ii] == '3' || aa[ii] == '4' || aa[ii] == '5' || aa[ii] == '6' || aa[ii] == '7' || aa[ii] == '8' || aa[ii] == '9') && v != 0) {

inputt[v] = aa[ii];

v++;

}

else

if (aa[ii] == '.' && flag == false) {

flag = true;

inputt[v] = aa[ii];

v++;

}

else

if (aa[ii] == '0' && v == 0) {

inputt[v] = aa[ii];

break;

}

else

if (v > 0)

break;

}

if (inputt[0] == 0 && inputt[1] == 0 && inputt[2] == 0 && inputt[3] == 0 && inputt[4] == 0 && inputt[5] == 0 && inputt[6] == 0 && inputt[7] == 0)

A[i][j] = 20000;

else {

aa = string() + inputt[0] + inputt[1] + inputt[2] + inputt[3] + inputt[4] + inputt[5] + inputt[6] + inputt[7];

A[i][j] = stof(aa);

}

if (aa == "")

cout << endl;

else

if (A[i][j] > 10000 || A[i][j] < -10000)

cout << "No,no,no! Invalid input. Enter a number between -10000 and 10000. The string must not be empty.\n";

} while (A[i][j] < -10000 || A[i][j] > 10000);

}

}

}

cout << "Your matrix:";

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

cout << "\n";

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

if (st == 'f') {

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

B = A;

}

else {

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

BB = AA;

}

}

}

return 0;

}

int main()

{

cout << "Hello! This program can find the saddle element of the matrix.\n";

cout << "\n";

string aa, bb;

cout << "Enter the number of rows in the matrix:\n";

a = exceptionn(aa);

cout << "Enter the number of columns in the matrix:\n";

b = exceptionn(bb);

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

B[i] = new float[b];

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

BB[i] = new int[b];

shablon(a,b); //Заполнение матрицы

cout << endl;

if (st == 'i') //Перегрузка в зависимости от типа данных

matrix(BB,a,b);

else

matrix(B,a,b);

return 0;

}

**Вывод**

В ходе выполнения лабораторной работы было разработано первое консольное приложение на языке программирования C++.

Главной особенностью данной лабораторной работы являлась необходимость использования перегрузок и шаблонов при вводе пользователем данных и подсчёте результата. Для ввода данных матрицы было принять решение использовать шаблон, а для поиска седлового элемента матрицы – перегрузку.

После ознакомления с материалами и выполнения лабораторной работы стало ясно, что шаблоны больше подходят при выполнении одного и того же действия с разными типами данных, а перегрузка для выполнения различных операций в зависимости от типа данных. В данной лабораторной работе было бы лучше использовать шаблоны, т.к. для типов int и float выполняются одинаковые действия по вводу матрицы и поиску седлового элемента.