

**Hassaan Akbar Cheema**

**174351**

**CS250: Data Structure and Algorithms**

**Class: BSCS-6C**

**Lab 2: Pointers and Dynamic Memory**

**Instructor: Mr. Abid Rauf**



CS250: Data Structure and Algorithms Page 1

**Lab 2: Pointers and Dynamic Memory**

**Task 3:**

To write a C++ program to perform matrix addition and subtraction using pointers. Matrices are basically stored in 2D arrays, which are created using dynamic memory allocation

**CODE:**

//Hassan Akbar Cheema

//174351

//BSCS 6C

#include<iostream>

#include <stdlib.h>

#include<time.h>

using namespace std;

bool comparison(int r1, int r2, int C1, int C2) {

if (r1 == r2 && C1 == C2) {

return true;

}

else {

cout << "\nMatrix with same order can be added or subtracted try again\n\n";

return false;

}

}

void MatrixAddition() {

int R1, R2,C1, C2;

bool Flag;

do {

cout << "\nM1 Rows number: ";

cin >> R1;

cout << "\nM1 column number: ";

cin >> C1;

cout << "\nM2 Rows number: ";

cin >> R2;

cout << "\nM2 column number: ";

cin >> C2;

Flag = comparison(R1, R2, C1, C2);

} while (!Flag);

int\* M1 = new int[R1\*C1];

int\* M2 = new int[R2\*C2];

cout << "\nMatrx M1\n";

for (int i = 0; i < R1\*C1; i++)

{

M1[i] = rand() % 20;

cout << "\t" << M1[i] << "\t";

if (((i + 1) % C1) == 0)

cout << endl;

}

cout << "\nMatrx M2\n";

for (int j = 0; j < R2\*C2; j++)

{

M2[j] = rand() % 20;

cout << "\t" << M2[j] << "\t";

if (((j + 1) % C2) == 0)

cout << endl;

}

cout << "\n OUTPUT Matrix \n" << endl;

for (int c = 0; c < R1\*C1; c++) {

cout << "\t" << M1[c] + M2[c] << "\t";

if (((c + 1) % C1) == 0)

cout << endl;

}

getchar();

getchar();

}

void MatrixSubtraction() {

int R1, R2,C1, C2;

bool Flag;

do {

cout << "\nM1 Rows number: ";

cin >> R1;

cout << "\nM1 column number: ";

cin >> C1;

cout << "\nM2 Rows number: ";

cin >> R2;

cout << "\nM2 column number: ";

cin >> C2;

Flag = comparison(R1, R2, C1, C2);

} while (!Flag);

int\* M1 = new int[R1\*C1];

int\* M2 = new int[R2\*C2];

cout << "\nMatrx M1\n";

for (int i = 0; i < R1\*C1; i++)

{

M1[i] = rand() % 20;

cout << "\t" << M1[i] << "\t";

if (((i + 1) % C1) == 0)

cout << endl;

}

cout << "\nMatrx M2\n";

for (int j = 0; j < R2\*C2; j++)

{

M2[j] = rand() % 20;

cout << "\t" << M2[j] << "\t";

if (((j + 1) % C2) == 0)

cout << endl;

}

cout << "\n OUTPUT Matrix \n" << endl;

for (int c = 0; c < R1\*C1; c++) {

cout << "\t" << M1[c] - M2[c] << "\t";

if (((c + 1) % C1) == 0)

cout << endl;

}

getchar();

getchar();

}

int main(void) {

int R1=1, R2=1,C1=1, C2=1;

int Foo;

srand(time(NULL));

cout << "For Sum press 1\n\nFor Subtraction press 2: \n";

cin >> Foo;

if (R1 == R2 && C1 == C2) {

if (Foo == 1) {

MatrixAddition();

}

if (Foo == 2) {

MatrixSubtraction();

}

}

}



