Multiplicación de Matriz NO CONCURRENTE “Guía”

En el main.cpp

Declaramos 2 filas y 2 columnas, a su vez el trabajo se divide en 2 partes:

Parte 1- Donde se ingresa dos matrices nxm y te da la opción de ingresar datos; con fines que un usuario desee probarlo con cualquier tipo de matriz.

Parte 2- Una matriz en la cual puedes ingresar filas y columnas en las dos matrices, pero los valores serán colocarlos en modo random.

Parte 1:

cout<<"Multiplication matrix..."<<endl;  
cout<<".............Part-1 Enter Values...........!!!"<<endl;  
cout<<"Ingrese row de Matrix1"<<endl;  
cin>>row1;  
cout<<"Ingrese Column de Matrix1"<<endl;  
cin>>column1;  
cout<<"Ingrese row de Matrix2"<<endl;  
cin>>row2;  
cout<<"Ingrese Column de Matrix2"<<endl;  
cin>>column2;  
  
Matriz<int> A(row1,column1), B(row2,column2),C;  
A.fill\_Manual();  
B.fill\_Manual();  
C = A \* B;  
cout<<"Matrix 1->"<<endl;  
A.Show();  
cout<<endl;  
cout<<"Matrix 2->"<<endl;  
B.Show();  
cout<<endl;  
C.Show();

Parte2:

cout<<".............Random Entry Part-2...........!!!"<<endl;  
cout<<"Ingrese Matrix 1 row"<<endl;  
cin>>row1;  
cout<<"Ingrese Matrix 1 column"<<endl;  
cin>>column1;  
cout<<"Ingrese Matrix 2 row"<<endl;  
cin>>row2;  
cout<<"Ingrese Matrix 2 column"<<endl;  
cin>>column2;  
  
Matriz<int> m1(row1, column1), m2(row2, column2),m3;  
  
m1.fill\_random();  
m2.fill\_random();  
  
m1.Show();  
cout << endl;  
m2.Show();  
m3 = m1 \* m2;  
cout << endl;  
m3.Show();

Ahora se crearon las carpetas de Matriz.cpp / Matriz.h

Matriz.h:

Donde añadimos Templete, en la clase Matriz tenemos de modo privado la fila y columna con los elementos en modo de ingreso diferentes. Para el tipo publico creamos los constructores, los métodos de Show, fill\_random, fill\_Manual, para ver la matriz y rellenarla de forma aleatoria y manual. Tenemos los operadores también de Operator= y Operador\* que es necesario para nuestra matriz que pueda multiplicar.

Metodos Utilizados:

<random>: Utilizamos esta librería para el modo random para tener un mejor desempeño.

#include <iostream>

Dificultades:

-Agregar los templetes anterior mente usaba float pero tiene que ser genérico y agregamos por eso T (Templete).

-Otra dificultar fue que la matriz no esta llenada y me botaba un error al usar los siguiente:

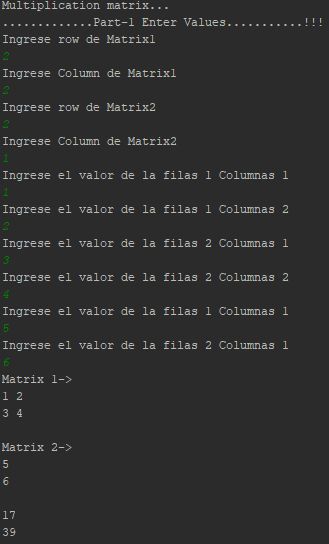
Matriz<T>::Matriz(const int f, const int c){}

La solución fue agregar un FOR para llenarlos desde 0 ya que sacaba la matriz:

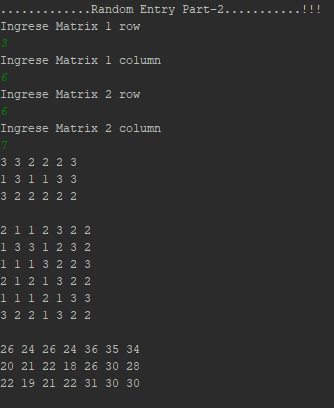
for (int j=0;j<rows;j++){  
 for(int k=0;k<columns;k++){  
 elementos[j][k]=0;  
 }  
}

Pruebas: Cuando la matriz concuerda con la otra.

Parte1

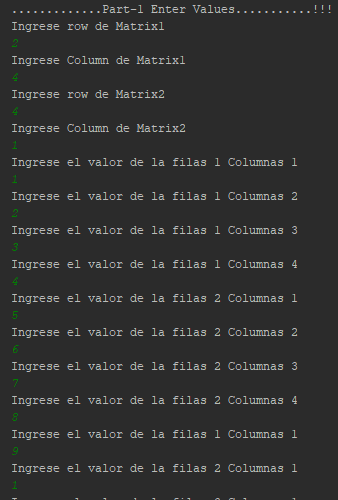


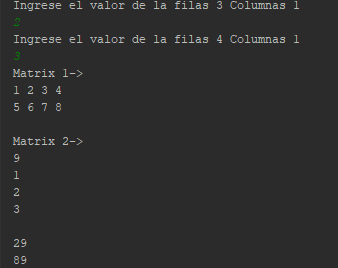
Parte 2



Pruebas cuando la Matriz1 - columna y Matriz2-fila no son iguales y no se puede hacer la multiplicación:

Parte 1:





Parte 2:

