COMPUTACION

<u>Área personal</u> / Mis cursos / <u>Cursos Basicos</u> / <u>2o Semestre</u> / <u>Compu</u> / <u>Semana 21 - FINAL 1</u> / <u>Tema 3 - 1er Final - Fila 2</u>

Descripción

Ver entrega

Tema 3 - 1er Final - Fila 2

Disponible desde: miércoles, 2 de diciembre de 2020, 10:30 **Límite de entrega**: miércoles, 2 de diciembre de 2020, 11:50

Ficheros requeridos: t3F1F2.cpp (Descargar)

Tipo de trabajo: Individual

Universidad Nacional de Asunción - Facultad de Ingeniería

CATEDRA DE COMPUTACIÓN - EXAMEN FINAL 1 - 02/dic/2020 - TEMA 3

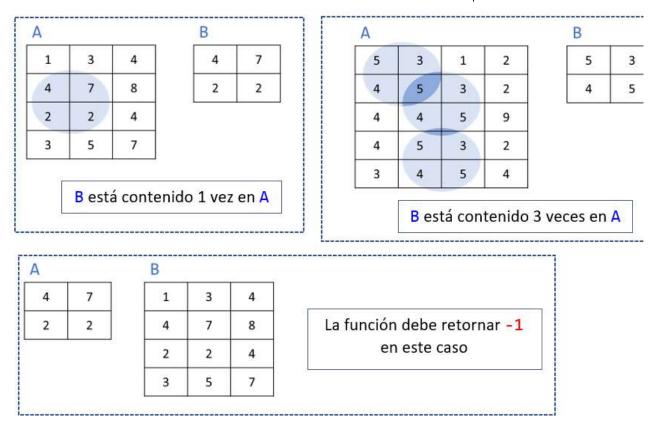
Fila 2

Escribir una función en C++ llamada contarApariciones que:

- Lea (por teclado) las dimensiones de una matriz A (de m filas y n columnas) de números enteros.
- Lea (por teclado) los elementos de la matriz A.
- Lea (por teclado) las dimensiones de otra matriz B (de p filas y q columnas) de números enteros.
- Lea (por teclado) los elementos de la matriz B.
- Retorne la cantidad de veces que B está contenido en A. Si no existe la posibilidad de que A esté contenida en B (por las dimensiones de ambas), la función debe retornar -1.

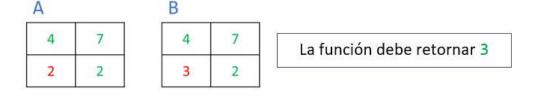
Nota: Las dimensiones de las matrices A y B deben validarse (es decir, asegurarse de que sean números enteros y positivos). Esto se hará a través de la función int validarDim(), que deberá implementarse. No hace falta validar que los elementos ingresados para las matrices A y B sean números enteros.

Algunos ejemplos son los siguientes:



Una función útil para resolver este ejercicio (que también debe implementarse) es:

int cantidadDeCoincidencias(int *b, int *c, int m, int n) , que recibe la direcciones de los elementos en la posición [0][0] de dos matrices (ambas de tamaño m * n y contienen números enteros), y devuelve la cantidad de elementos (correspondientes) que son iguales. Por ejemplo:



OTRAS DIRECTIVAS A SER NECESARIAMENTE TENIDAS EN CUENTA:

- NO MODIFICAR LA FUNCIÓN PRINCIPAL (main). LA MODIFICACIÓN SERÁ PENALIZADA, pues puede alterarse la secuencia de lecturas en la entrada y la salida esperada para el programa.
- **DEFINIR NECESARIAMENTE LA FUNCIÓN QUE REALICE LO SOLICITADO** . No se aceptará que la funcionalidad se desarrolle en el main.
- **SE PROPORCIONA la función** void leerElementosMatriz(int *x, int y, int z), que carga los elementos de una matriz (de y filas y z columnas). El puntero x contiene la dirección del elemento inicial [0][0] de la matriz.
- EN EL VPL, en la función contarNroApariciones se encuentra una sección inicial de código que carga las dimensiones y elementos de las matrices A y B, utilizando la implementación de validarDim y la función auxiliar leerElementosMatriz. Se recomienda no modificar esa sección.
- LA SOLUCIÓN DEBE SER LO SUFICIENTEMENTE GENERAL para resolver cualquier entrada diferente a las propuestas en los ejemplos y los casos de prueba. Por lo tanto, la calificación mostrada por el sistema no necesariamente corresponderá con la calificación final.

- La distribución de puntajes para cada función es la siguiente:
 - Función validarDim: 10%
 - Función cantidadDeCoincidencias : 40%
 - Función contarApariciones : 50%

Ficheros requeridos

t3F1F2.cpp

```
#include<iostream>
     using namespace std;
     /*Función auxiliar que carga los elementos de una matriz (de "y" filas y "z" columnas).
     El puntero x contiene la dirección del elemento inicial [0][0] */
     void leerElementosMatriz(int *x, int y, int z){
          int i,j;
         for(i=0;i< y;i++) for(j=0;j< z;j++) cin>>*(x+i*z+j);
 9
     }
10
11
     /*Esta función debe leer números por teclado, tantas veces sea necesario, de tal manera a asegurarse de que el mismo
     sea entero y positivo. Retornar el número que cumpla con esta condición. */
13
     int validarDim(){
14
         //Comience aqui su implementación
15
16
    }
17
     /*Esta función recibe las direcciones de los elementos en [\theta][\theta] de dos matrices B y C (de las mismas dimensiones - m fila
18
19
     y devuelve la cantidad de elementos (correspondientes) que son iguales.*/
     int cantidadDeCoincidencias(int *b, int *c, int m, int n){
21
         //Comience aqui su implementación
22
23
24
    }
25
     /*La sección inicial de código para esta función se encarga de llamar a validarDim() para cargar las dimensiones de las ma
26
27
     y a leerElementosMatriz() para cargar sus elementos.
28
29
     La función debe retornar la cantidad de apariciones de B en A.*/
30
     int contarNroApariciones(){
31
32
          /*---NO MODIFICAR ESTA SECCIÓN---*/
33
         int m=validarDim(); //cantidad de filas de A
         int n=validarDim(); // cantidad de columnas de A
34
         int A[m][n]; // matriz A de m filas y n columnas
35
         leerElementosMatriz(&A[0][0],m,n); //lectura de los elementos de A
36
         int p=validarDim(); //cantidad de filas de B
         int q=validarDim(); // cantidad de columnas de B
39
         int B[p][q]; // matriz B de p filas y q columnas
40
         leerElementosMatriz(&B[0][0],p,q); //lectura de los elementos de B
41
         /*----FIN DE LA SECCIÓN-----
42
43
         //Comience aqui su implementación
44
45
46
     }
47
48
     int main(){
          /*---NO MODIFICAR El MAIN*/
49
50
         int opcion;
51
         cin>>opcion;
52
         if(opcion==1){
              cout<<validarDim()<<endl;</pre>
53
54
         if(opcion==2){
55
              int m=validarDim(); //cantidad de filas de A
56
              int n=validarDim(); // cantidad de columnas de A
int A[m][n]; // matriz A de m filas y n columnas
57
58
             leerElementosMatriz(&A[0][0],m,n); //lectura de los elementos de A int p=validarDim(); //cantidad de filas de B int q=validarDim(); // cantidad de columnas de B int B[p][q]; // matriz B de p filas y q columnas
59
60
61
62
              leerElementosMatriz(\&B[0][0],p,q); \ //lectura \ de \ los \ elementos \ de \ B
63
                                            cout<<cantidadDeCoincidencias(&A[0][0],&B[0][0],p,q)<<endl;</pre>
64
              if(opcion==2\&\&m==p\&\&n==q)
65
         if(opcion==3)
                          cout<<contarNroApariciones()<<endl;</pre>
66
67
         return 0;
68
```

▼ Final 1 - Tema 2 - Fila 2

Ir a...

Programa de estudio semana 1 ►

