



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación Salas A y B

Profesor: Juan Alfredo Cruz Carlón.

Asignatura: Fundamentos de programación.

Grupo: 1107

No de Práctica(s): #10

Integrante(s): Montiel Vera Daniel.

Semestre: 2018-1

Fecha de entrega: 3 de Noviembre del 2017

Observaciones:

CALIFICACIÓN: _____

LLENADO DE MATRICES

DIAGRAMA DE FLUJO

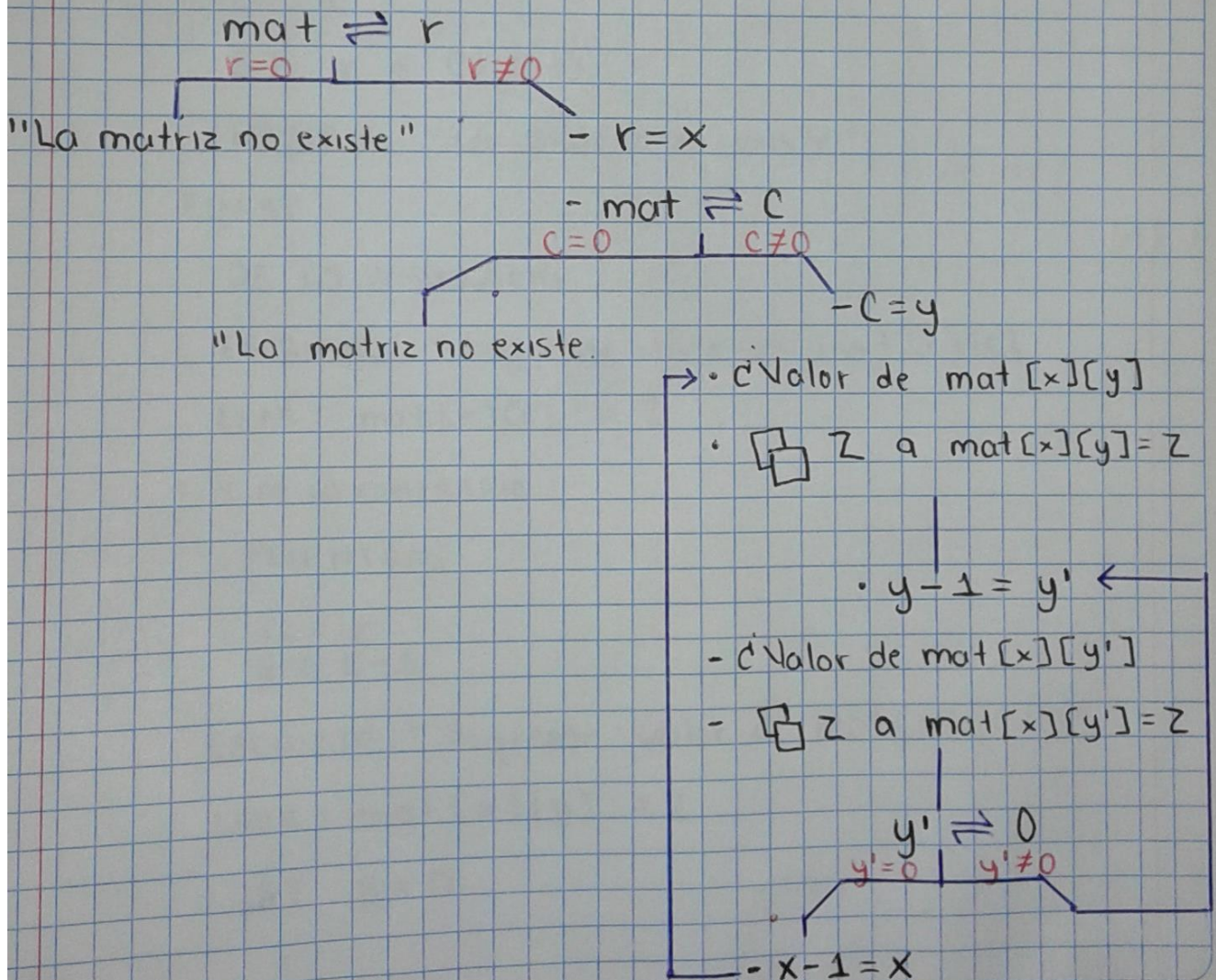
r = renglones

C = columnas

Z = Valor ingresado

mat = matriz

x, y = coordenados.



INICIO

mat [x] [y] : ENTERO

r : ENTERO

c : ENTERO

z : ENTERO

ESCRIBIR : "Ingresa el número de renglones"

LEER : r

ESCRIBIR : "Ingresa el número de columnas"

LEER : c

SI $r \text{ ó } c = 0$

ESCRIBIR : "La matriz no existe"

FIN SI

DE LO CONTRARIO

ESCRIBIR : "Ingresar valor de mat[r][c]"

LEER : $\text{mat}[r][c] := z$

FIN DE LO CONTRARIO

MIENTRAS

$x := r$

$y := c - 1$

ESCRIBIR : "Ingresar valor de mat[r][c-1]"

LEER : $\text{mat}[x][y] := z$

SI $y = 0$

ENTONCES

$x := r - 1$

$y := c$

FIN SI

FIN MIENTRAS

SI

$x := 0$

$y := 0$

ESCRIBIR : " Ingreso el valor de $\text{mat}[x][y]$ "

LEER : $\text{mat}[x][y] := z$

FIN SI

FIN

CODIGO EN C

```
#include<stdio.h>
#include<stdlib.h>

main ()
{
    int x,y;

    printf("¿De cuantos renglones sera tu matriz?:");
    scanf("%i",&x);

    printf("¿De cuantas columnas sera tu matriz?:");
    scanf("%i",&y);

    #define r= ("%i",x)
    #define c= ("%i",y)

    /*Al momento de definir las variables (x,y) tuve algunos problemas porque
    no sabía si dirigirlas solamente a (x,y) o agregarle el (%i). */

    int a[100][100],b[100][100],d[100][100],z,k;

    /*Estuve atorado en esta parte porque al principio definía las matrices exactas y
    el producto resultaba erróneo, después decidí ampliarlas a 100x100 y ya funcionaba. */

    printf("matriz_1 [%i][%i]",x,y);

    for(z=1;z<=x;z++)//renglones
        for(k=1;k<=y;k++)//columnas

    /* En los ciclos "for" también tuve problemas porque no sabía cómo asignarle valor al (<=),
    luego definí las variables (x,y) y funciono. */

    {
        printf("\n introducir valor de [%i][%i] :",z,k);
        scanf("%d",&a[z][k]);
    }
    system("cls");
}
```


MULTIPLICACION DE MATRICES

DIAGRAMA DE FLUJO.

r = renglones

r : de 1 a X

C = columnas

C : de 1 a Z

K = constante

K : de 1 a y

M_1 = matriz 1

M_2 = matriz 2

M_3 = matriz resultante.

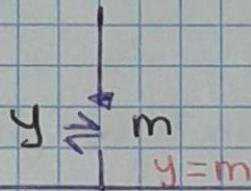
X = renglones de M_1

y = columnas de M_1

m = renglones de M_2

Z = columnas de M_2

$$[M_1 [x][y] * M_2 [m][z] = M_3 [r][c]]$$



"Las matrices no se pueden multiplicar"

r : de 1 a X

C : de 1 a C

$$M_3 [r][c] = 0$$

K : de 1 a y

$$M_3 [r][c] = M_3 [r][c] + M_1 [r][K] * M_2 [K][c]$$

PSEUDOCODIGO.

INICIO

mat_1 [x][y] : ENTERO

mat_2 [m][z] : ENTERO

mat_3 [x][z] : ENTERO

r := 1 a x

c := 1 a z

k := 1 a y

ESCRIBIR: "Ingresa el número de renglones"

LEER: x

ESCRIBIR: "Ingresa el número de columnas"

LEER: z

SI $x \neq z$

ESCRIBIR: "No se pueden multiplicar las matrices"

LEER: error

FIN SI

DE LO CONTRARIO

MIENTRAS

r := 1 a x

c := 1 a z

k := 1 a y

mat_3[x][z] := 0

FUNC mult (mat_1[r][k] * mat_2[k][c])

FUNC suma (mat_3[r][c] + (mat_1[r][k] * mat_2[k][c])

FIN FUNC mult

FIN FUNC suma

FIN MIENTRAS

FIN DE LO CONTRARIO

FIN

PSEUDOCODIGO DE LLENADO, MULTIPLICACION E IMPRESIÓN DE MATRICES.

```
#include<stdio.h>
#include<stdlib.h>

main ()
{
int x,y;

printf("¿De cuantos renglones sera tu matriz?:");
scanf("%i",&x);

printf("¿De cuantas columnas sera tu matriz?");
scanf("%i",&y);

#define r= ("%i",x)
#define c= ("%i",y)

/*Al momento de definir las variables (x,y) tuve algunos problemas porque
no sabía si dirigir las solamente a (x,y) o agregarle el (%i). */

int a[100][100],b[100][100],d[100][100],z,k;

/*Estuve atorado en esta parte porque al principio definía las matrices exactas y
el producto resultaba erróneo, después decidí ampliarlas a 100x100 y ya funcionaba. */

printf("matriz_1 [%i][%i]",x,y);

for(z=1;z<=x;z++)//renglones
for(k=1;k<=y;k++)//columnas

/* En los ciclos "for" también tuve problemas porque no sabía cómo asignarle valor al (<=),
luego definí las variables (x,y) y funcionó. */

{
printf("\n introducir valor de [%i][%i] :",z,k);
scanf("%d",&a[z][k]);
}
system("cls");

printf("matriz_2 [%i][%i]",x,y);
for(z=1;z<=x;z++)//renglones
for(k=1;k<=y;k++)//columnas

{
```

```

        printf("\n introducir valor de [%i][%i] :",z,k);
        scanf("%d",&b[z][k]);
    }
    system("cls");

    for(z=1;z<=x;z++)//renglones
    for(k=1;k<=y;k++)//columnas
        d[z][k]= (a[1][k]*b[z][1])+(a[2][k]*b[z][2])+(a[3][k]*b[z][3])+(a[4][k]*b[z][4])+0;

/* Para darle la instrucción de multiplicar, fueron una serie de pruebas
que consistían en relacionar las coordenadas de renglones y columnas para
lograr un orden al realizar la operación */


        for(z=1;z<=x;z++)//renglones
    {
        for(k=1;k<=y;k++)//columnas

            printf("%i", d[z][k]);
            printf("\n");

/* Aquí para la impresión de la matriz primero había intentado usar las funciones
de (\t) y (\v), pero no me funcionaron como quería.*/


    }

    system ("pause");
}

```