



Ingeniería en Seguridad Informática y Redes

Ingeniería en Sistemas Computacionales

Algoritmos de Solución Numérica

Algoritmos de Primer Sesión

SI727576 - Edgar Guzmán Claustro

IS727272 - Marco Ricardo Cordero Hernández

Jal., 22 de agosto de 2023

Contexto

A manera introductoria al curso al cual se realiza esta actividad y con el propósito de comenzar a desarrollar habilidades para el diseño secuencial y lógico de problemas, se solicita la resolución de los siguientes algoritmos mediante diagramas de flujo:

1. Encontrar el factorial de un número dado por el usuario y mostrar el resultado.
Restricciones: El número dado por el usuario no puede ser negativo, menor o igual a 20.
2. Mostrar la serie de Fibonacci hasta un límite dado por el usuario.
3. Entregar como resultado el número de días que faltan para el final del año desde una fecha dada por el usuario. Restricción: No considerar años bisiestos.
4. Resolver la multiplicación de un par de matrices cuadradas de una dimensión dada por el usuario.

Resultados

Problema 1

Pseudocódigo

Algoritmo factorial

 Escribir Sin Saltar "Ingrese término "; Leer n

 Si $n > 0$ y $n < 20$ Entonces

$res \leftarrow n$

 Mientras $n > 1$ Hacer

$res \leftarrow res * (n - 1)$

$n \leftarrow n - 1$

 Fin Mientras

 Escribir "Resultado: ", res

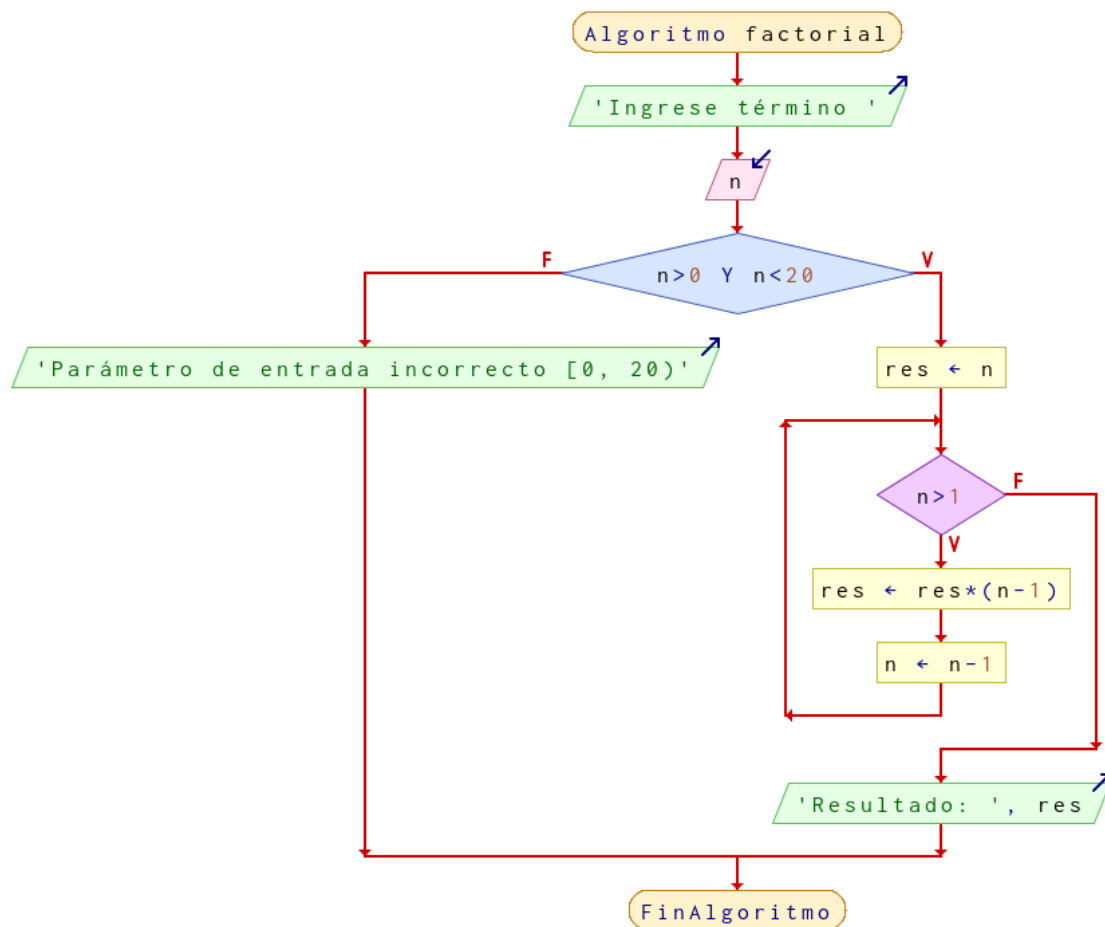
 SiNo

 Escribir "Parámetro de entrada incorrecto [0, 20)"

 Fin Si

FinAlgoritmo

Diagrama de flujo



Problema 2

Pseudocódigo

Algoritmo Serie_de_Fibonacci

init1 <- 0

init2 <- 1

init3 <- 0

Repetir

 Escribir 'Ingrese el número de elementos: '

 Leer length

Hasta Que length>0

Para i<-1 Hasta length Con Paso 1 Hacer

 Escribir init1

 init3 <- init1+init2

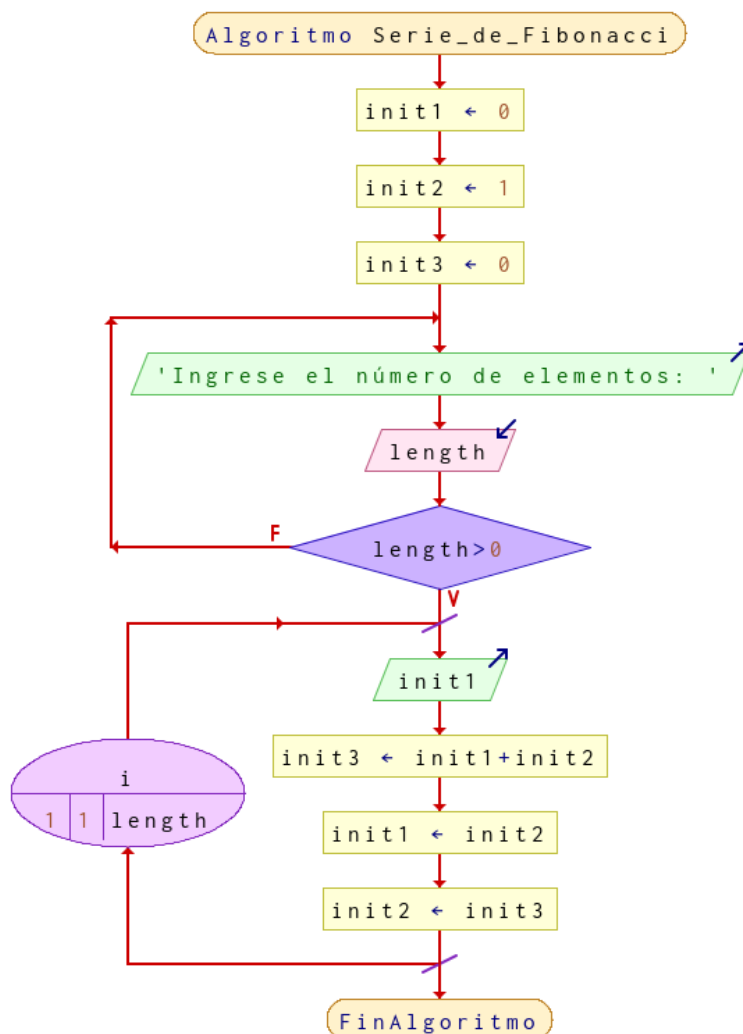
 init1 <- init2

 init2 <- init3

FinPara

FinAlgoritmo

Diagrama de flujo



Problema 3

Pseudocódigo

Algoritmo dias_restantes

 Escribir Sin Saltar "Ingrese día "; Leer d

 Si $0 < d$ y $d \leq 31$ Entonces

 Escribir Sin Saltar "Ingrese mes "; Leer m

 Si $0 < m$ y $m \leq 12$ Entonces

 Segun m Hacer

 1, 3, 5, 7, 8, 10, 12:

 res \leftarrow 31 - d

 4, 6, 9, 11:

 res \leftarrow 30 - d

 De Otro Modo:

 res \leftarrow 28 - d

 Fin Segun

 Si res ≥ 0 Entonces

 Mientras m \neq 12 Hacer

 Segun m Hacer

 1, 3, 5, 7, 8, 10, 12:

 res \leftarrow res + 31

 4, 6, 9, 11:

 res \leftarrow res + 30

 De Otro Modo:

 res \leftarrow res + 28

 Fin Segun

 m \leftarrow m + 1

 FinMientras

 Escribir "Días restantes: ", res

 SiNo

 Escribir "Combinación de día (", d, ") y mes (", m, ") inválida.

Revisa tus entradas."

 FinSi

 SiNo

 Escribir "Mes inválido. Revisa tu entrada."

 FinSi

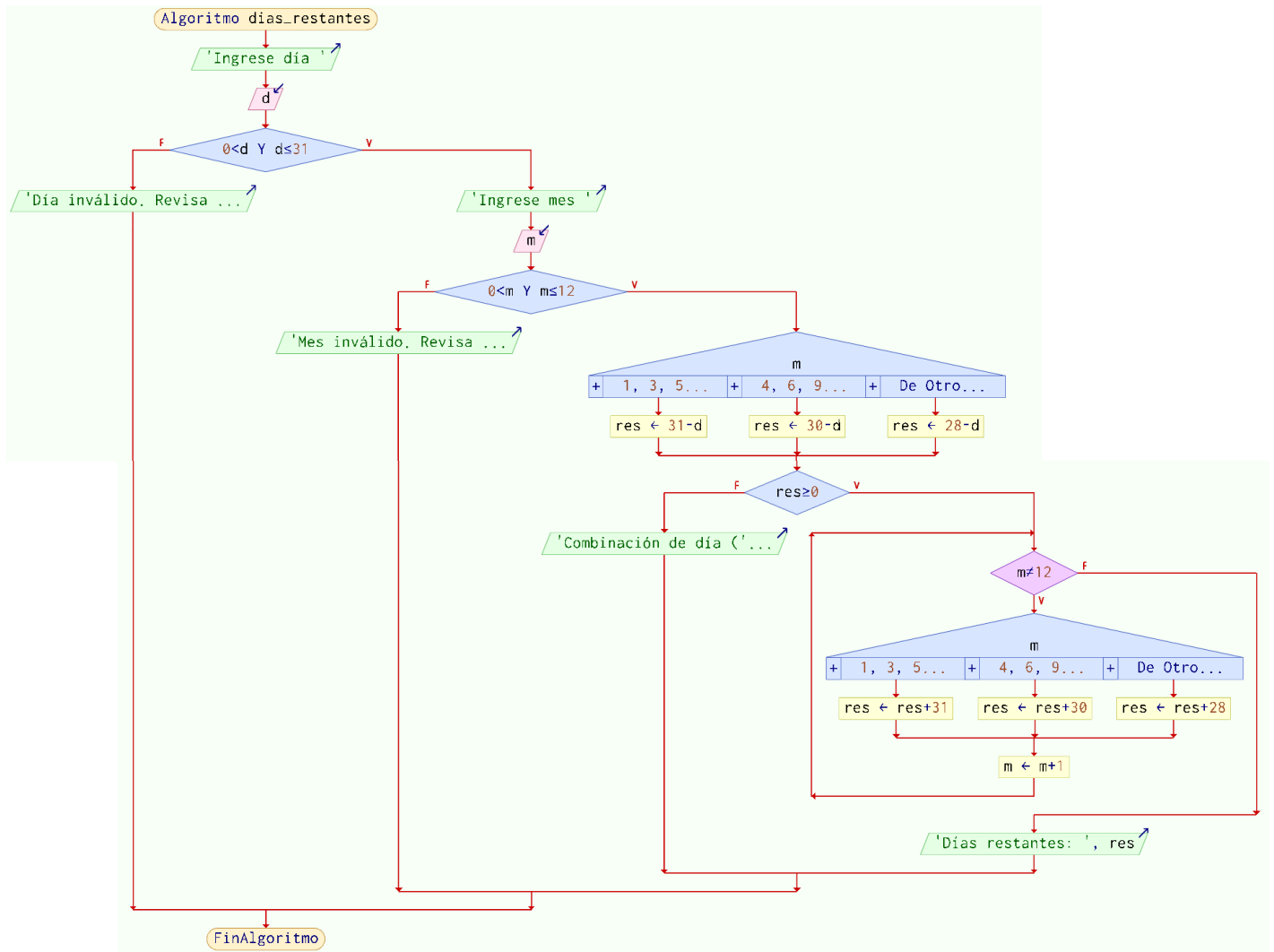
 SiNo

 Escribir "Día inválido. Revisa tu entrada."

 FinSi

FinAlgoritmo

Diagrama de flujo



Problema 4

Pseudocódigo

Función llenarMatriz (mat Por Referencia,row,column,number)

Para i<-1 Hasta row Con Paso 1 Hacer

Para j<-1 Hasta column Con Paso 1 Hacer

Escribir 'Ingrese los datos para la matriz: ', number, ' :[' , i, ' : ', j, ']'

Leer mat[i,j]

FinPara

FinPara

FinFunción

Función mostrarMat (mat Por Referencia,row,column,number)

Escribir 'Matriz: ', number

Para i<-1 Hasta row Con Paso 1 Hacer

Para j<-1 Hasta column Con Paso 1 Hacer

Escribir 'Ingrese los datos para la matriz: ', number, ' :[' , i, ' : ', j, ']: ',

mat[i,j]

FinPara

FinPara

FinFunción

Algoritmo Multiplicacion_matrices

Definir row, column, a, b, c Como Entero

Repetir

Escribir 'Ingrese el número de la matriz : '

Leer column

Hasta Que column>0

row<-column

Dimension a(row,column), b(row,column), c(row,column)

llenarMatriz(a,row,column,1)

llenarMatriz(b,row,column,2)

Para i<-1 Hasta row Con Paso 1 Hacer

Para j<-1 Hasta column Con Paso 1 Hacer

Para k<-1 Hasta row Con Paso 1 Hacer

$c[i, j] \leftarrow c[i, j] + a[i, k] * b[k, j]$

FinPara

FinPara

FinPara

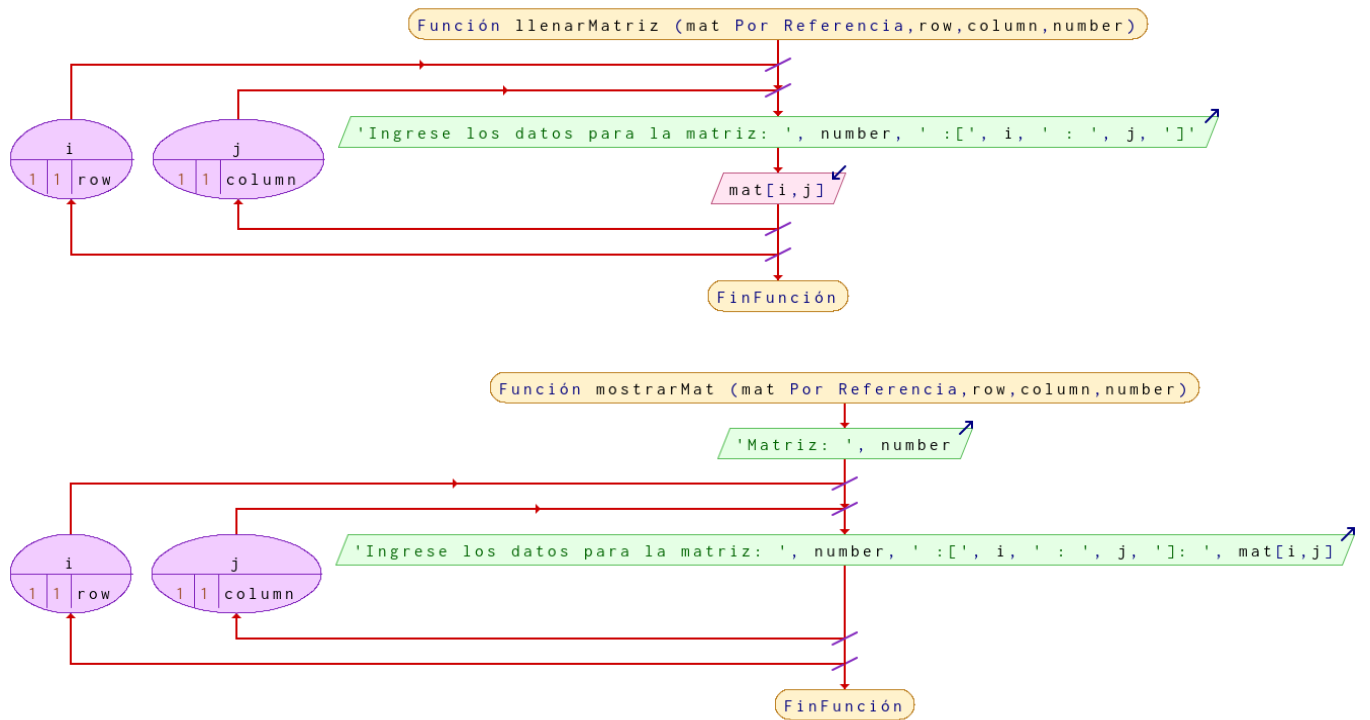
mostrarMat(a,row,column,1)

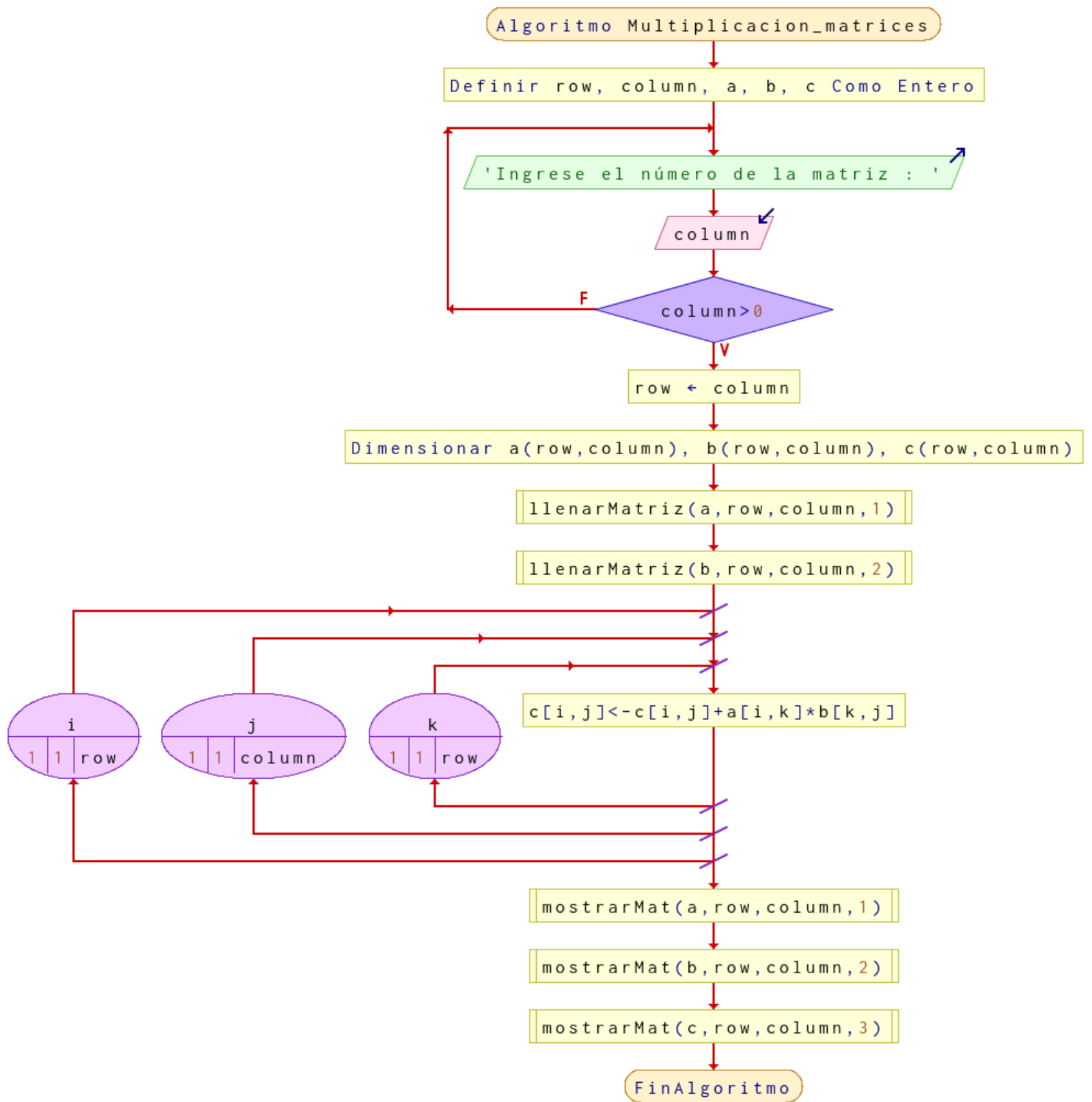
mostrarMat(b,row,column,2)

mostrarMat(c,row,column,3)

FinAlgoritmo

Diagrama(s) de flujo





Conclusiones

Guzmán Claustro, Edgar

Dentro del diseño de algoritmos, siempre se debe tomar en cuenta la solución que menos recursos consume para el sistema. En este caso, traté de optimizar lo más que pude manteniendo la funcionalidad del algoritmo y la experiencia de usuario.

Otra cosa destacable dentro de esto, es el manejo de errores. Se deben manejar los errores de manera óptima porque nunca se sabe qué tipo de razonamiento tenga el usuario al momento de ejecutar el programa.

Cordero Hernández, Marco R.

La introducción al curso actual por medio de diseño de algoritmos básicos ha sido un gran ejercicio para revisar las bases de la programación: la resolución de problemas por medio de automatización. Al ya haber cursado la materia de “Algoritmos y Programación”, analizar estos escenarios recuerda las enseñanzas encontradas en dicho curso, en donde la solución aparente no siempre es la correcta, y el análisis puntual de todos los posibles casos para un problema conlleva un reto lógico adicional.

Quizás la complejidad de estos ejercicios no ha sido demasiado elevada para alguien con trasfondo informático, y no se hace mención de esto con el propósito de alardear ni mucho más, al contrario, se menciona porque se espera un aumento exponencial de la dificultad del material venidero.