



INSTITUTO POLITÉCNICO NACIONAL



Escuela Superior de Cómputo

Unidad de Aprendizaje: Sistemas operativos

Segundo Parcial

Práctica 2

Profesor: Josué Rangel Gonzalez

Integrantes del equipo:

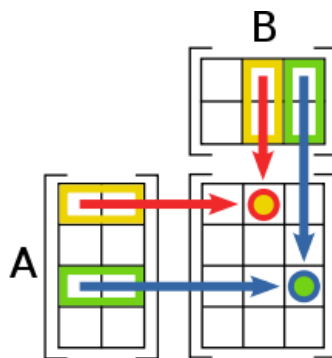
Martín Eduardo Barriga Vargas

José Manuel Ramirez Vives

Fecha de entrega: Jueves 5 de abril del 2018

## Introducción:

Una multiplicación de matrices consiste en efectuar el producto entre dos matrices en la que una de ellas comparte el mismo número de filas que el número de columnas de la otra, y esta se da siguiendo un algoritmo en el que la matriz resultante tendrá valores obtenidos de la suma de las multiplicaciones de los números de las filas de la primera matriz por los de la columna de la segunda matriz.

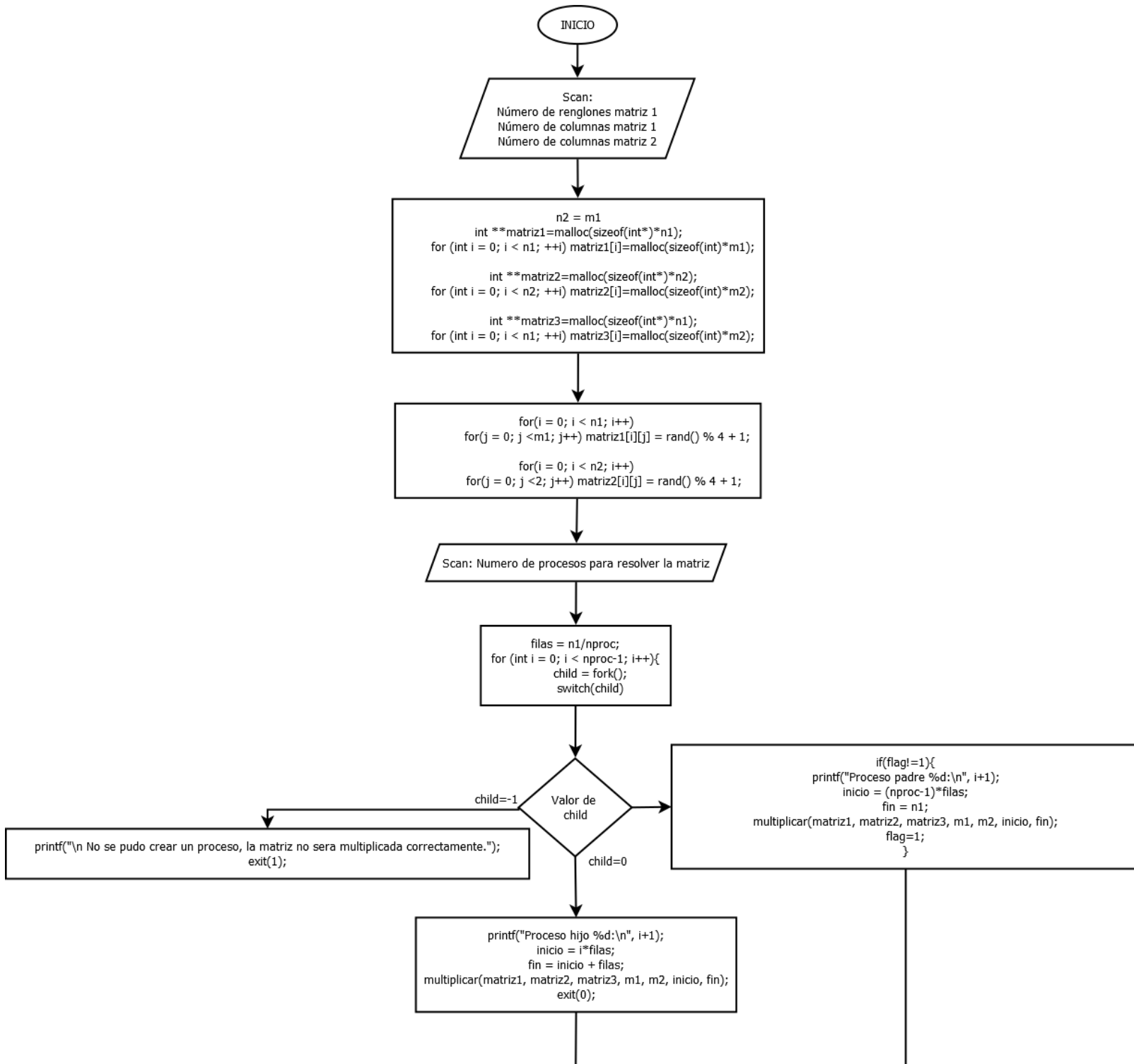


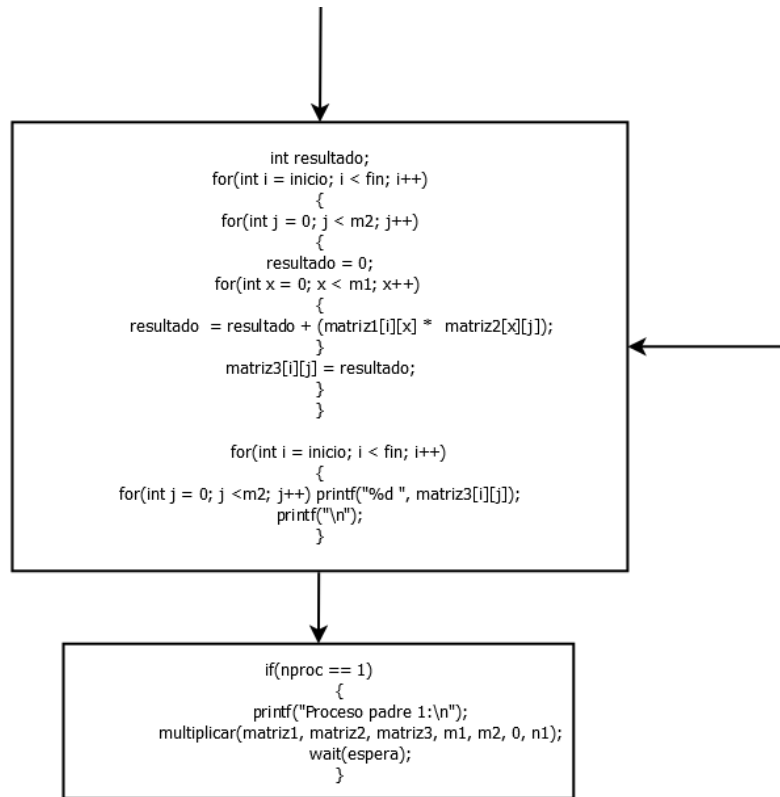
Los resultados en las posiciones marcadas dependen de las filas y columnas de sus respectivos colores.

Continuando nuestro aprendizaje del tema de procesos, deberemos de realizar la multiplicación de dos matrices pidiéndole al usuario con anterioridad el número de procesos que se encargarán de efectuar la multiplicación, es decir, cada proceso se encargará de multiplicar cierta cantidad de filas por las columnas de la segunda matriz, asegurándonos de que al final todas las multiplicaciones de las filas habrán sido parte de alguno de los procesos que el usuario pidió que se efectuaran.

## Descripción de la solución:

### Diagrama de flujo





## Descripción

Una vez iniciado el programa se le pide al usuario ingresar la cantidad de filas y columnas de la primera matriz, así como la cantidad de columnas de la segunda matriz, sin embargo no le pediremos la cantidad de filas de la segunda matriz, puesto que le asignaremos automáticamente el valor de las columnas de la primera matriz ya que es necesario que tengan el mismo valor para poder efectuar la multiplicación. Luego procedemos a crear y llenar automáticamente las dos matrices con números entre el 1 y el 4, además, se le muestra al usuario las matrices llenas para poder evaluar nuestros resultados posteriores correctamente. A continuación, se le pide al usuario ingresar la cantidad de procesos en las que se van a repartir las filas a multiplicar y le asignamos a una variable la cantidad de filas que va a realizar cada proceso, exceptuando el último ya que en algunos casos este contendrá más o menos filas a multiplicar que los demás. En un ciclo de 0 a la cantidad de procesos que nos pidieron menos uno, se va a crear un proceso y en caso de que sea un proceso hijo deberemos de mandar a multiplicar desde la fila con el índice del proceso por la cantidad de filas que debe resolver, hasta este número encontrado más la cantidad de filas que debe resolver. Sin embargo, si el proceso es padre, deberemos de checar que sea el primero y que no tenga que ver con los hijos. Una vez que ya checamos que sea el primero con ayuda de una bandera, deberemos de realizar la multiplicación de las filas restantes hasta el final, es decir, el número de procesos menos uno por el numero de filas hasta el numero de filas, y una vez multiplicado marcar en uno la bandera. Al igual para un caso en el que el número de procesos sea uno, se implementó una copia del mismo código para el padre que había dentro del for.

## Conclusiones:

Martín Eduardo Barriga Vargas:

De nueva cuenta utilizamos la instrucción fork para el desarrollo de nuestra práctica, pues era necesario crear a nuestros procesos hijos. Tuvimos que dar un repaso a las reglas para la multiplicación de matrices e implementarlo al programa. Sin embargo, mientras más se avanzaba con la práctica, más se complicaba. Esto debido a que no sabíamos bien como distribuir la cantidad de filas por procesos y cómo organizar nuestras condiciones para generar los hijos correspondientes. Para resolver esta problemática, sólo dividimos la cantidad de filas entre los procesos y lo que nos sobrara se lo dejaríamos al proceso padre, además para evitar que se realizaran múltiples multiplicaciones en el código del proceso padre se colocó una bandera para que sólo se ejecutara una vez.

Jose Manuel Ramirez Vives:

La práctica realizada nos ayudó a reforzar los conocimientos adquiridos sobre la función fork y cómo la podemos utilizar para realizar procesos largos o complejos de manera más rápida pues podemos seccionarlo y de esta forma hacer que varios procesos lo resuelvan por partes y, de esta manera, se puede decir que el procesador dedica más tiempo para nuestro programa que para los demás procesos que debe atender. Considero que la parte que más se complicó de la práctica fue el determinar qué partes de la multiplicación de las matrices debe realizar cada proceso y cómo enviar la instrucción de que sólo hiciera esas partes. Finalmente lo determinados creo que de una manera sencilla que nos da el resultado esperado.