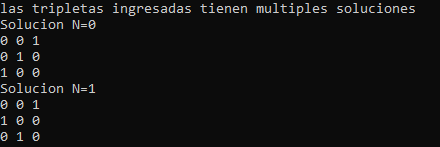
*Jose Toro Bustos – 20.239.232-5 - Sección A1*

***Representación de una posible solución.***

Una posible solución se representa gráficamente mediante una matriz con 1’s y 0’s donde los 1’s representan cuadrados negros y los 0’s representan cuadrados blancos

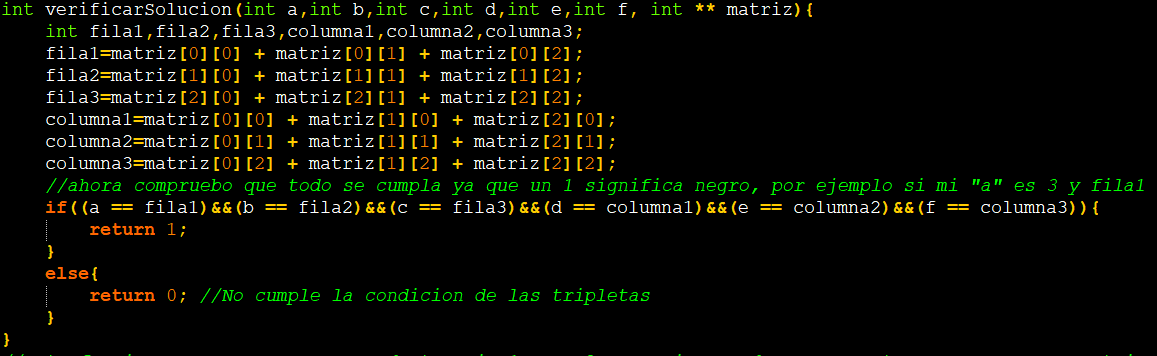
Como se ve en la imagen



Estas dos de las distintas soluciones para las tripletas (1,1,1) y (1,1,1)

***Listado de restricciones o conjunto de restricciones que poseen, explicando para cada una:***

el código posee una función llamada verificarSolucion, la cual se encarga de revisar si una matriz generada es una solución de las tripletas entregadas por el usuario

******

Ya que mi código genera todas las variables de matrices posibles (aproximadamente 500) mi función de verificar solución me entrega un valor 1 o 0 dependiendo si la matriz que estoy generando cumple o no con las tripletas asignadas

Para ello sumo la cantidad de 1’s de cada fila y de cada columna y la comparo con su valor correspondiente dentro de cada tripleta, si estas comparaciones resultan ser iguales, entonces esa matriz si es una solución

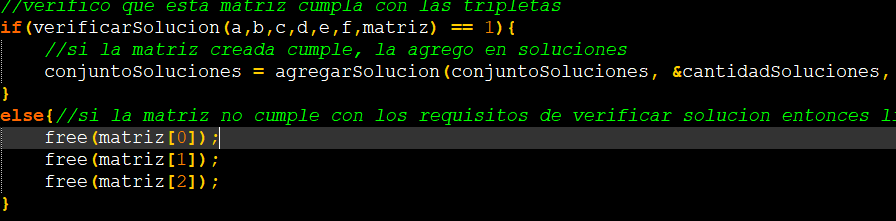
***Criterio de optimización, indicando por qué es un criterio de optimización y justificando a***

***este, o, en caso de no existir, justificar su inexistencia.***

Para optimizar la ejecución de mi código utilice la función verificarSolucion la cual en caso de que cumpla todo lo anteriormente mencionado me regresa un valor 1.

Con esto mi programa agrega esa matriz a una lista de soluciones

En caso de que no me regrese el valor 1, hago free de la matriz generada y no hay necesidad de agregarla en soluciones, de esta forma si bien genero todas las posibilidades, solo almaceno las matrices que cumplan las tripletas

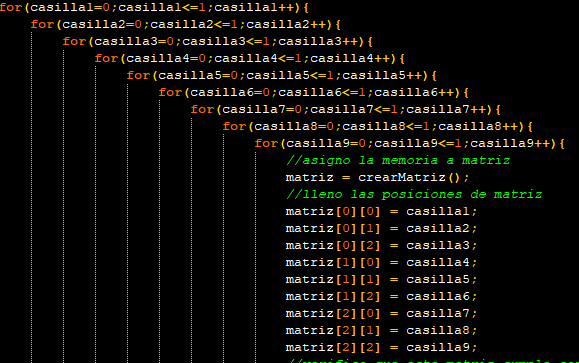


***El algoritmo de Búsqueda en espacio de soluciones aplicado.***

***(no supe a que se refiere)***

***Código que obtiene la solución.***

Para generar la solución utilice 9 ciclos for anidados como se ve en la imagen



Los cuales van variando el valor de cada variable casilla (1,2,3,4,5,6,7,8 y 9)

Haciéndolas variar entre los valores 0 y 1 de esta forma represento las matrices pintadas donde 0 es el color blanco y 1 el color negro, con estos ciclos for genero todas las posibles soluciones y luego gracias a la función verificarSolucion almaceno solo las que cumplen con las tripletas.

Luego dependiendo de cuantas soluciones tenga almacenadas, le entrego al usuario tres posibles mensajes

-“no hay solución para las tripletas” en caso de no hayan soluciones

-“las tripletas ingresadas tienen solución única” en caso de que solo haya una solución

-“las tripletas ingresadas tienen múltiples soluciones” en caso de tener mas de una solución

Además, le muestro al usuario la/s solucion/es de manera grafica como se ve en el punto “***Representación de una posible solución.*** “