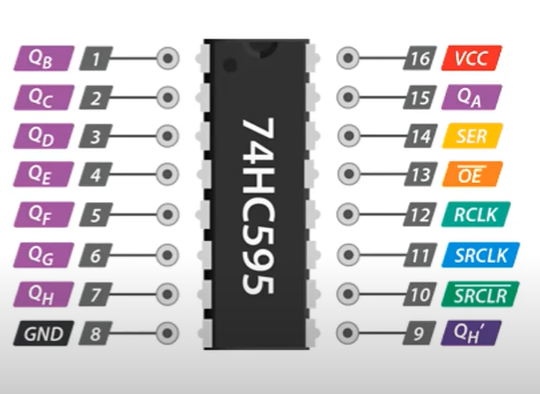
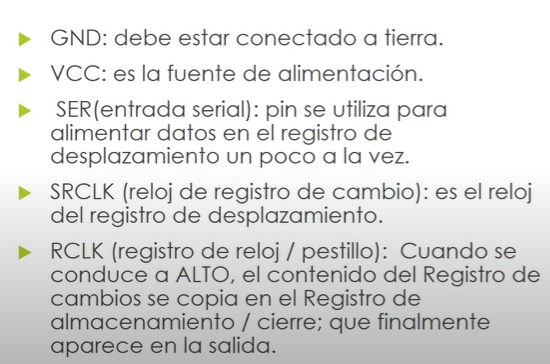
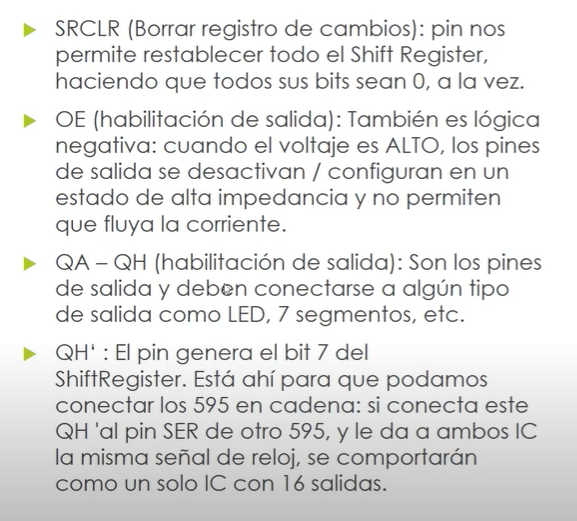
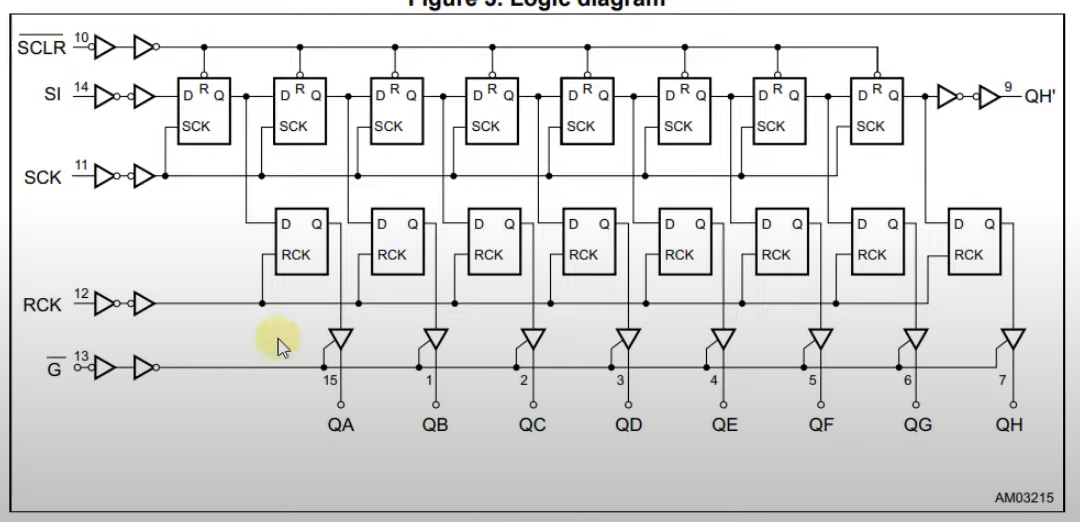
La primera vez que me enfrenté al problema, pensé que seria demasiado difícil, pero gracias a la información que podemos encontrar y por volver a releer muchas veces el problema, pude llegar a la solución de este mismo. Encontré tres estructuras, una con dos integrado y cada uno conectado a tres pines digitales, otra con dos integrados también, pero estos estarían conectados secuencialmente, ósea uno estaría conectado al otro, y el otro a los tres pines digitales del Arduino, y el ultimo seria cada fila conectada a un integrado, ósea 8 integrados conectados secuencialmente. Me encontraba con el dilema si utilizar los dos secuenciales, o el de 8, al final me decido al de 8 por el análisis realizado mostrando así que este sería mucho más dinámico y practica y con mas potencial a la hora de aplicarlo.

A continuación les mostraré mi estructura, como podemos ver y como he dicho antes, cada fila esta conectada a un integrado 74hc595, y se preguntaran porqué, y les diré que esta estructura me permite el manejo de 64 bits, ósea un bit para cada led, teniendo así cada led independiente, los 8 integrados se encuentran conectados secuencialmente por el pin(buscar), y uno es el que esta conectado directamente hacia el Arduino, les explicare en breves palabras el funcionamiento de un integrado 74hc595, (mostrar imagen)





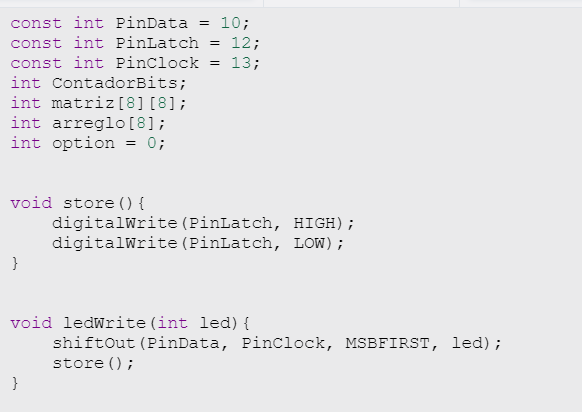




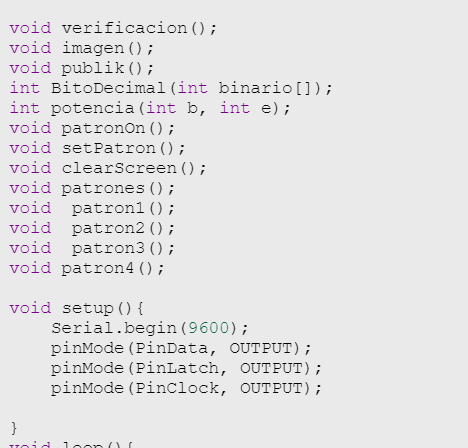
Además me gustaría decr que El 74hc595 cuenta con una gran particularidad, la cual es su uotput enable, ósea en este diagrama de flujo podemos ver que, el recibe los datos, los almacena pero tiene una llave, la cual se debe abrir para que muestre los datos en la salida, y se preguntarán, como se abre esta llave, esto va conectado al pin latch, y se abre dando un tiempo de reloj, ósea dándole 5v por ese pin y luego no darle nada, o viceversa.

Viendo ya la estructura podemos empezar viendo cómo funciona la situación planteada

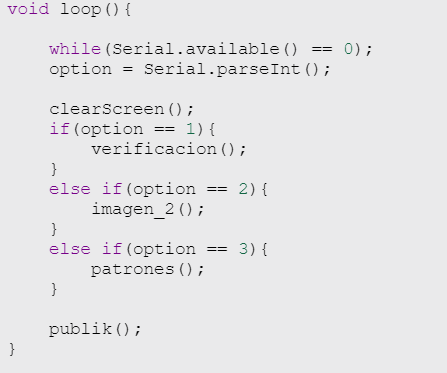
Explicación del código



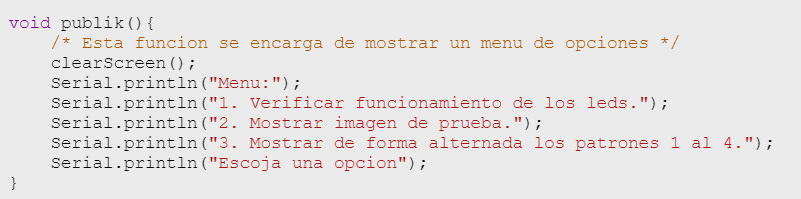
Al principio tenemos las declaraciones de variables, vemos que declaramos los pines digitales en los cuales están conectados los integrados 74hc595, y después tendremos una variable contadorbits, que se verá su utilización más adelante y option. tenemos la matriz que como su nombre lo dice es una matriz, utilice esta, debido a que necesito guardar el estado de cada led, ósea cada led tiene una posición especifica en una fila y columna de dicha estructura.



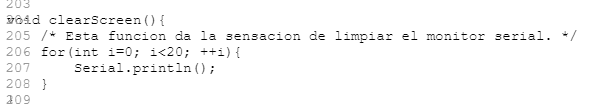
Luego tendremos la declaración de las funciones, y el setup, en el cual inicializamos el monitor serial, y ponemos los tres pines digitales en modo salda.



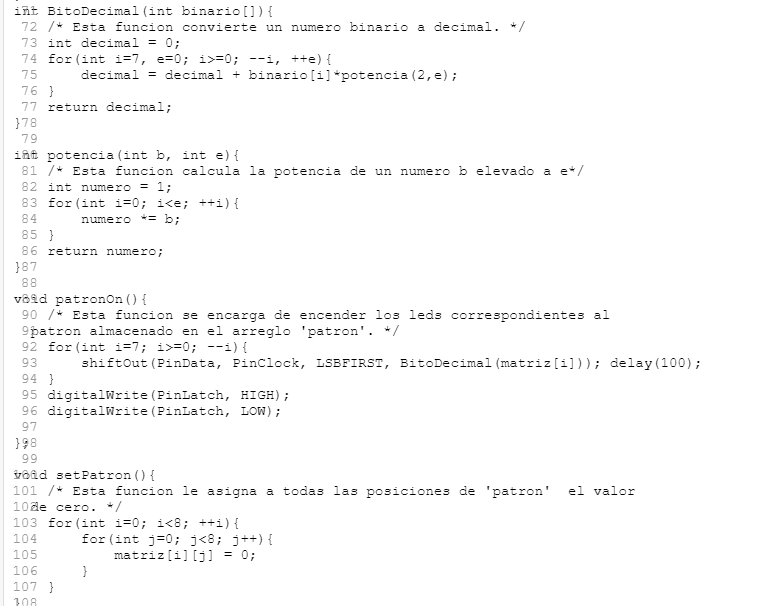
En el void loop encontraremos la forma de recibimiento para ingresar al menú, el cual esta contenido en la función publik, la lógica que podemos ver, es que mientras no halla nada en el serial, va a leer una variable, y ejecuta el menú. podemos ver la lógica condicional de que dependiendo de lo que el usuario ingrese, se ira a otras funciones, las cuales las explicaremos más adelante



Aquí tenemos la función publik, la cual es el menú, este ejecuta una función la cual desplaza lo que hay en el monitor serial, hacia arriba, para que se vea más limpio, esta función esta más adelante, pero básicamente imprime 20 enters en el monitor serial, la logica del public no es la mayor cosa, simplemente nos ayudamos con la funcion .println para poder imprimir en el serial.

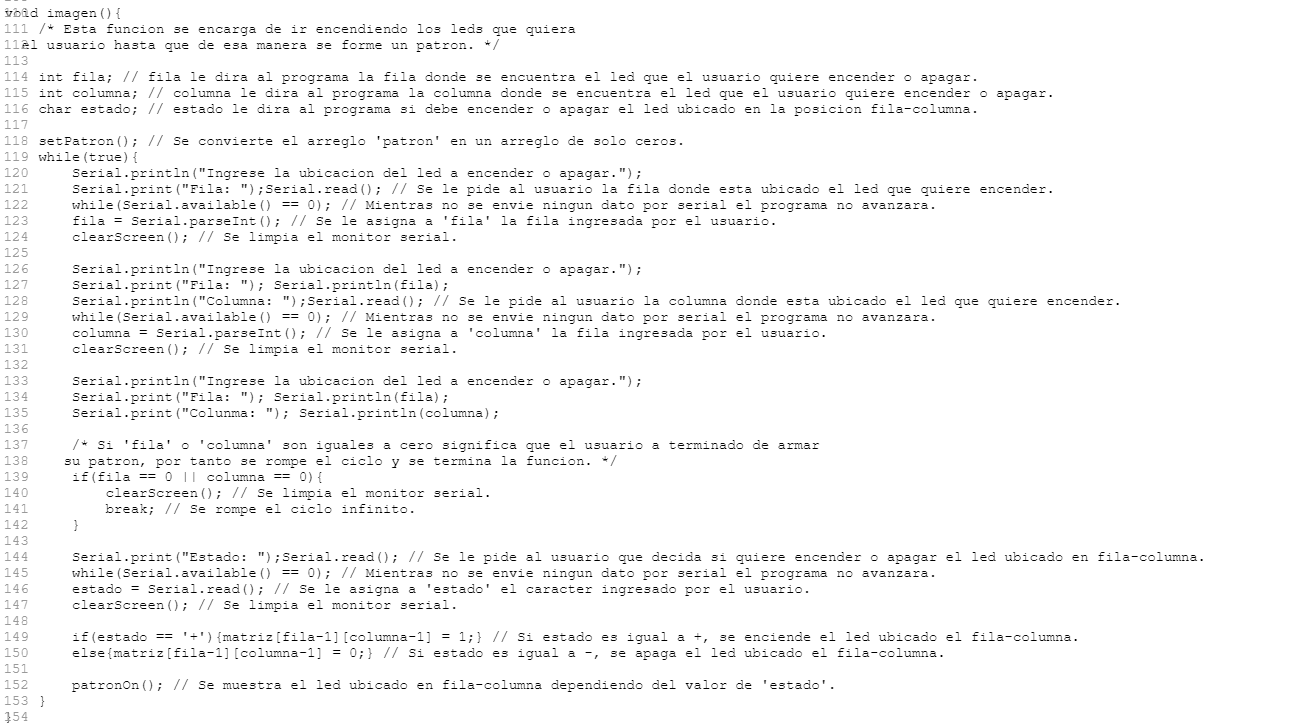


Esta función es muy sencilla, lo único que hace es imprimir 20 saltos de línea en el monitor serial con ayuda de la funicion println, dando como ilusión el hecho de que hemos limpiado el monitor serial.

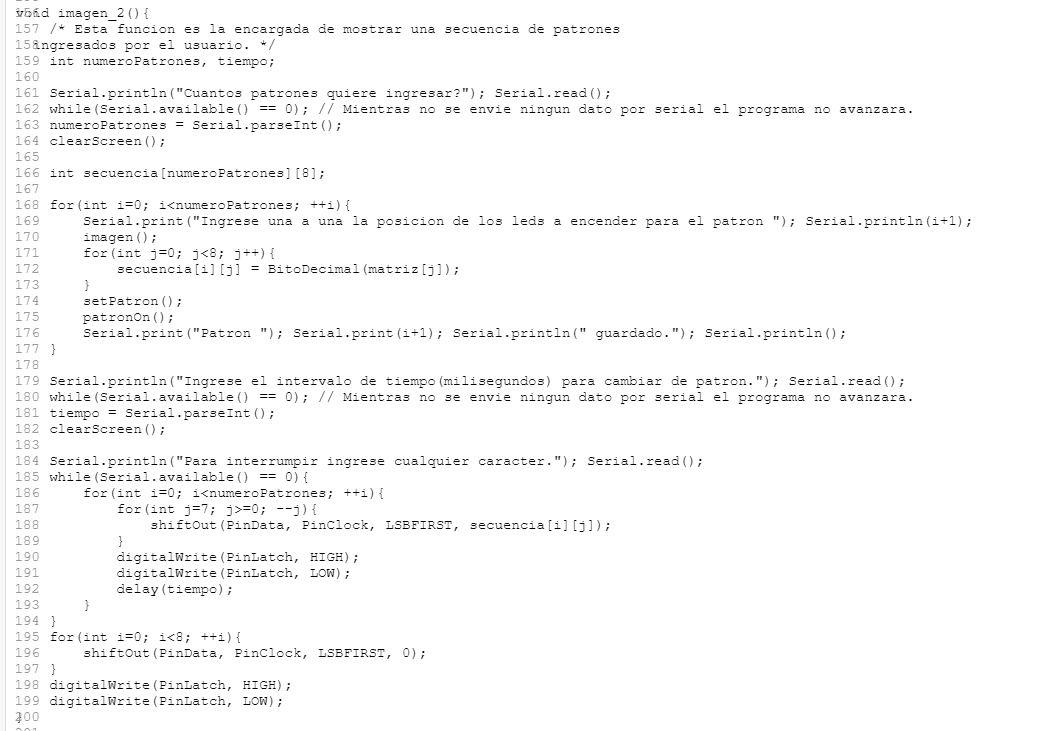


Luego tenemos estas 4 funciones las cuales, realmente son muy básicas y nos ayudan al funcionamiento de la matriz, la primer lo que hace es recibir un arreglo que representa un numero binario, y esta se ayuda con la función potencia para convertirlo en número decimal. Entonces la  La función potencia se usa para calcular las potencias de 2 que se multiplican por cada bit del arreglo binario. Y al final esos dos digitalwrite nos sirve para abrir el uotput enable.

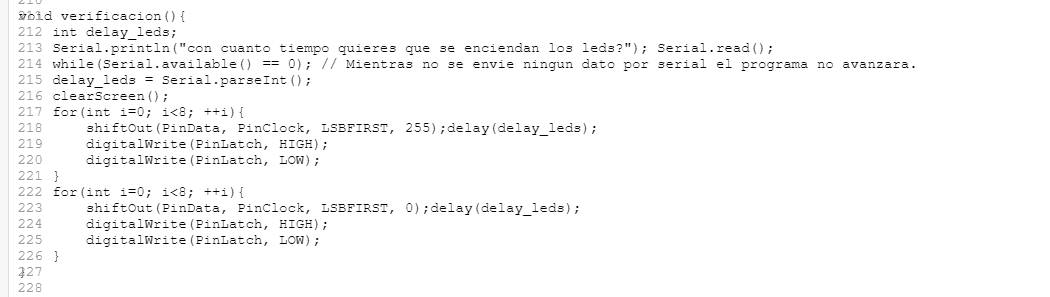
Luego sigue la función patronon, que lo que hace es iterar respecto a las filas de la variable matriz para mostrar lo que hay dentro de ellas, y para que se pueda mostrar en los leds se utiliza la función bintodecimal, y todo esto acompañado de la función shiofout que es la encargada de conectar los datos con los integrados. la función setpatron lo que hace es llenar esa matriz de puros ceros, con la ayuda de dos ciclos for anidados, que nos ayudaran a navegar entre filas y columnas.



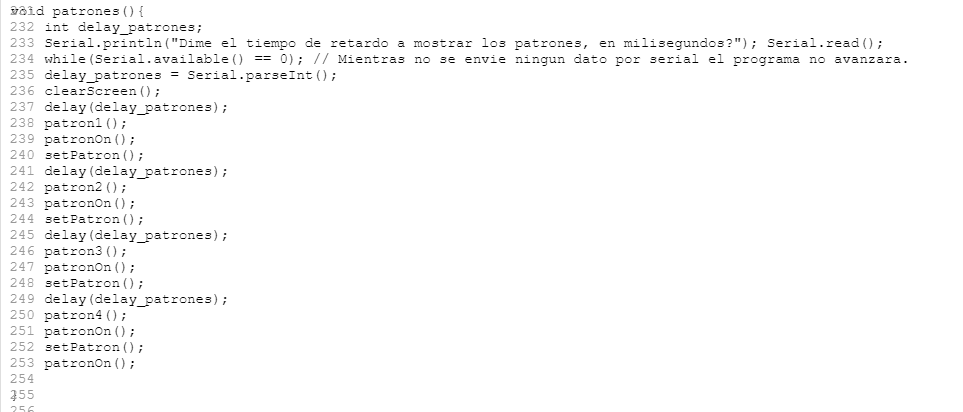
La función imagen en mi código es una función secundaria, que más adelante se completará con otra función para asi darle solución al problema planteado, primero creamos tres variables dos tipos int, y una char, luego inicializamos la matriz en ceros, y empezamos un ciclo el cual nos ayudara a registrar todo lo que introduzca el usuario, hasta que este ingrese cuando se le pida la fila y la columna, llene esto con ceros, pero bueno sigamos con el códgo, imprimimos con la función serial, lo que se ve ahí, luego utilizamos otro ciclo while, para que no siga con las otras sentencias hasta que el usuario ingrese un dato, se lee ese dato con la funcion seria.parseint ósea recibirá solo un dato entero, se limpia el monitor con la función vista anteriormente, y se vuelve otra vez a pedir pero en este caso la columna, y utilizamos la misma estructura para esperar que el usuario ingrese un dato. Evaluamos si el usuario ingreso cero en la fila y la columna, ya que esto indica que no ingresará nada más, y terminará con la función, luego comparado esto, se le pida que ingrese el estado, y utilizamos la misma estructura para esperar el dato, que en este caso en un tipo char, y si se quiere que se prenda se pondrá el signo +, y luego tenemos el if que evalua si el led se quiere enceder y lo que hace es pararse en esa fila y columa que el usuario ingreso, tiene menos 1, porque recordemos que se recorre desde 0, pero el usuario no debería de lidiar con esto, y luego se ayuda con la función patronon, para mostrar el patron.



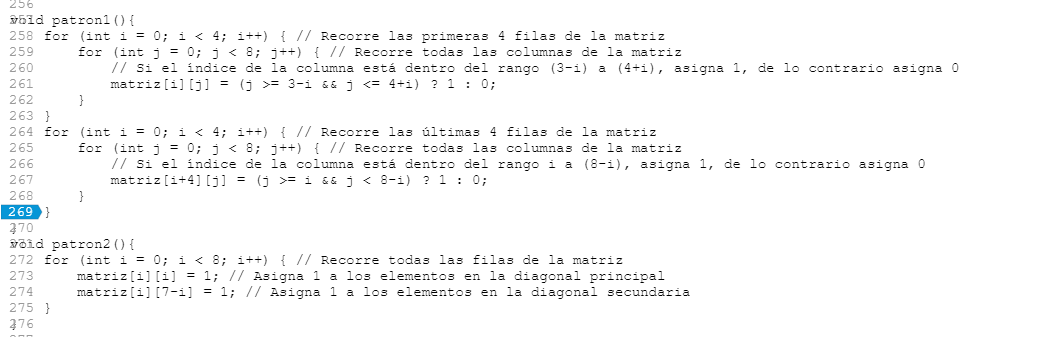
Esta función lo que hace es guardar los patrones que haya ingresado el usuario, entonces ´primero declaramos dos variables las cuales guardaran el numero de patrones que el usuario querrar guardar y la otra es el tiempo de espera en mostrar cada usuario. Se utiliza la misma técnica anterior con el while, y la función viable, la cual revisa si hay algo en el monitor serial, aquí la comparamos si es igual 0, para saber si el usuario ingreso algún dato, en este caso el dato ingresar es el numero de patrones que quiere guardar, luego se declara una matriz la cual tendrá el tamaño en filas de los patrones que el usuario quería guardar, luego tenemos un ciclo for, el cual se encargara de ejecutar todas las veces, que haya ingresado el usuario en el numero patrones, esto se ayudara con la función imagen, la cual permite al usuario mostrar la secuencia que quiera. después de salir de ese for, se encuentra con la pregunta de cuanto delay quiere, y utilizamos la misma técnica anterior para esperar que el usuario ingrese un dato, luego se utilizara un while para mostrar todos los patrones guardados, y esto para cuando el usuario ingrese cualquier carácter.



Esta función lo que hace es prender línea por línea y luego apagar línea por línea, para saber si los leds están funcionando, entonces empezamos declarando el tiempo con el cual se prendera cada linea, después lo que hacemos es pedirlo como lo hemos hecho antes, luego el primer for, lo que hará es ayudarse de la función shifout y el 255 que representa 8 unos y eso prendera todos los leds, y los dos digitalwrite no permite que output enabale se abra, y el otro es para apagar fila por fila los leds.



En esta función lo único que hacemos es seguir un orden y ejecutar funciones las cuales en caso de los patrones lo veremos mas adelante, entonces lo único que hacemos es pedir un numero que será el tiempo que se tardará en mostrar cada patron, entonces se ejecuta cada patron, luego se muestra y luego la matriz se vuelve cero



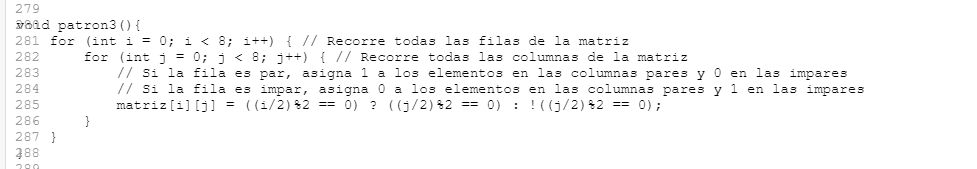
Tenemos los dos primeros patrones. En el primero haremos la matriz mitad a mitad, en la primera mitad utilizaremos dos ciclos for, debido a que debemos modificar las colmunas y entonces, utlizaremos un condicional el cual nos dice que si el índice de la columa esta dentro de la operación tres – i o 4 + i asigna uno, de lo contrario asigna cero, esto se puede hacer gracias al signo de interrogación que es otro condicional

Luego hacemos la otra mitad, la cual sigue la misma lógica, pero cambia al momento de asignar ya que cambian los rangos. Y dice los mismo si esta dentro del rango le asigna uno de lo contrario 0

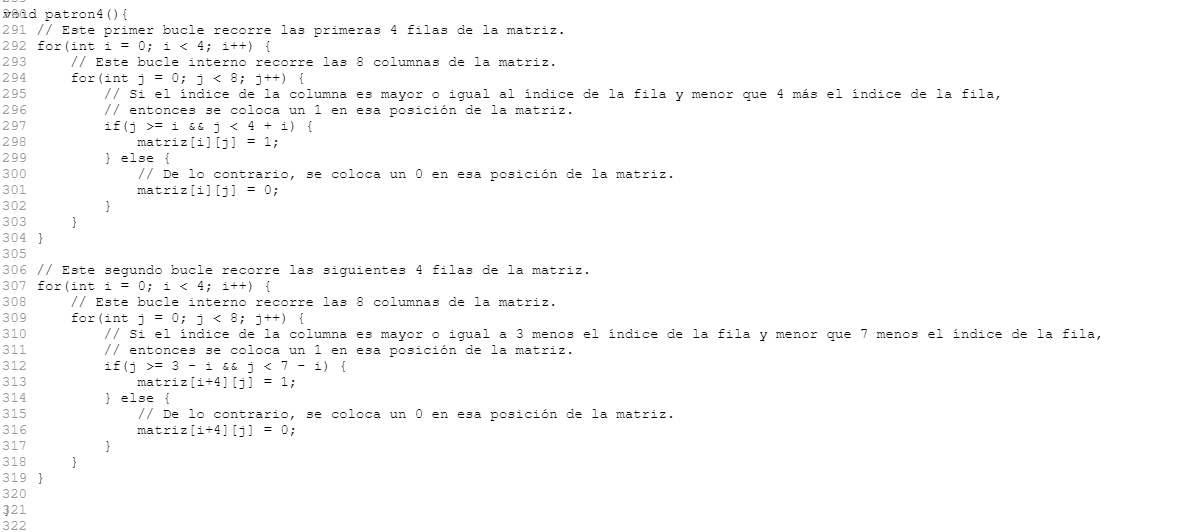
Esta definición de rango sale del análisis de la secuencia que lleva el patron.

Y se debe de tener en cuenta que en cada patron modifica la matriz principal debido a la propiedad que esta otorga debido a que no es necesario pedirla como parámetro de las funciones

Luego el patron 2, lo modificamos es ya desde las filas, esto es debido a que solo tiene dos leds encendidos por secuencia, y esto se puede ver al momento de asignar entones, la primera asignación dará 1 a la diagonal, y la otra los mismo, pero al reves.



Luego tenemos el patrón tres, que si modificamos desde las columas, entonces Esta línea de código asigna un valor booleano a la posición i, j de una matriz, dependiendo de si i y j son pares o impares. El operador ternario ósea el signo de interrogación y los dos? : se usa para evaluar una condición y devolver un valor si es verdadera o otro si es falsa. La condición es que verifica si la mitad de i es par. Si es así, el valor que se asigna es  que verifica si la mitad de j es par. Si no, el valor que se asigna es, que es el opuesto al anterior. Por ejemplo, si i = 4 y j = 6, la condición es verdadera y el valor que se asigna es verdadero, ya que ambas mitades son pares. Si i = 5 y j = 7, la condición es falsa y el valor que se asigna es falso, ya que ambas mitades son impares. Si i = 4 y j = 7, la condición es verdadera y el valor que se asigna es falso, ya que la mitad de j es impar. Si i = 5 y j = 6, la condición es falsa y el valor que se asigna es verdadero, ya que la mitad de j es par.



Lo que vemos en esta función es la partición de la matriz en dos, entonces en la primera parte lo que haremos es irnos a la columna gracias a los dos for, y preguntáramos con el condicional if que si // Si el índice de la columna es mayor o igual al índice de la fila y menor que 4 más el índice de la fila, entonces se coloca un 1 en esa posición de la matriz. Si no de lo contrario pondrá cero, y la i aumentará cada vez que se itere el for más externo

Y en la segunda mitad tenemos que // Si el índice de la columna es mayor o igual a 3 menos el índice de la fila y menor que 7 menos el índice de la fila, entonces se coloca un 1 en esa posición de la matriz. De lo contrario se coloca cero en esa posición, y la i ira variando, para as formar el patrón.

Hasta aquí ha llegado todo muchas gracias por la atención y espero que les haya gustasdo, hasta la próxima.