

Cubo LED 4 x 4 x 4

Objetivo General.

Construir un cubo con 64 LED, elaborar un programa por medio de Arduino para crear secuencias de iluminación.

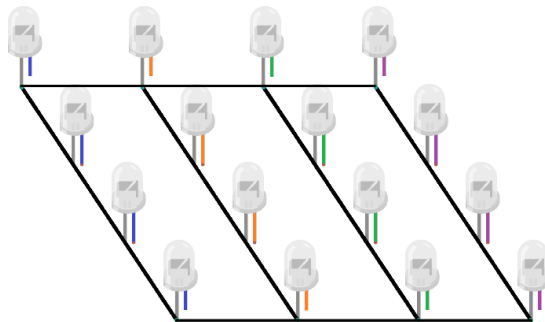
Desarrollo de la Actividad Práctica.

1. Asignar los pines de salida para cada columna y fila del cubo
2. Realizar la conexión de cada columna y fila del cubo
3. Subir el código a la placa ARDUINO.
4. Comprobar la funcionalidad de la práctica y checar si no hay algún LED fundido.

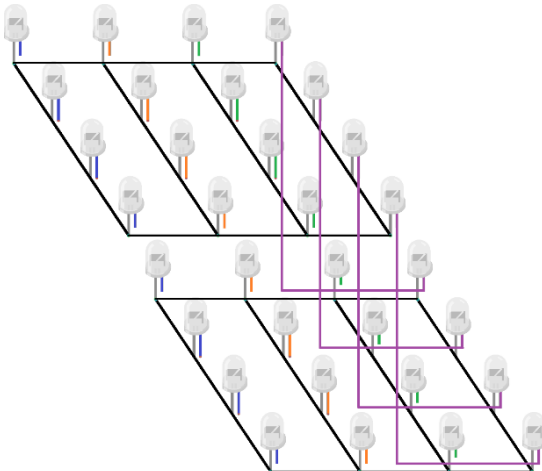
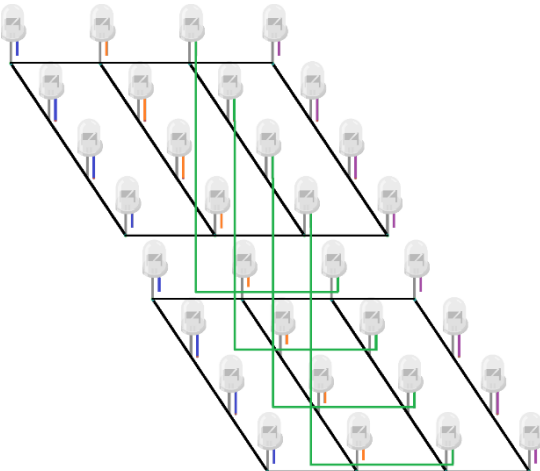
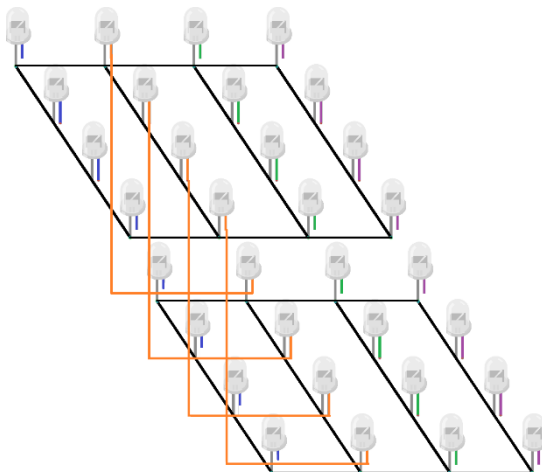
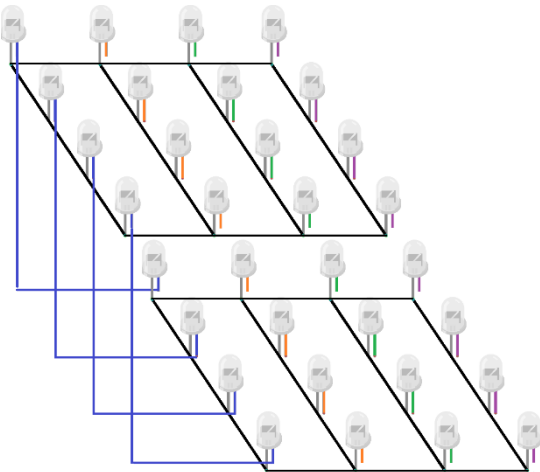
Materiales	
Cantidad	Modelo
1	ARD-010
1	ARD-310
1	509-010
1	USB-490
64	E5/ROJ-SUPER
16	R220 ½
10	A22R-1000

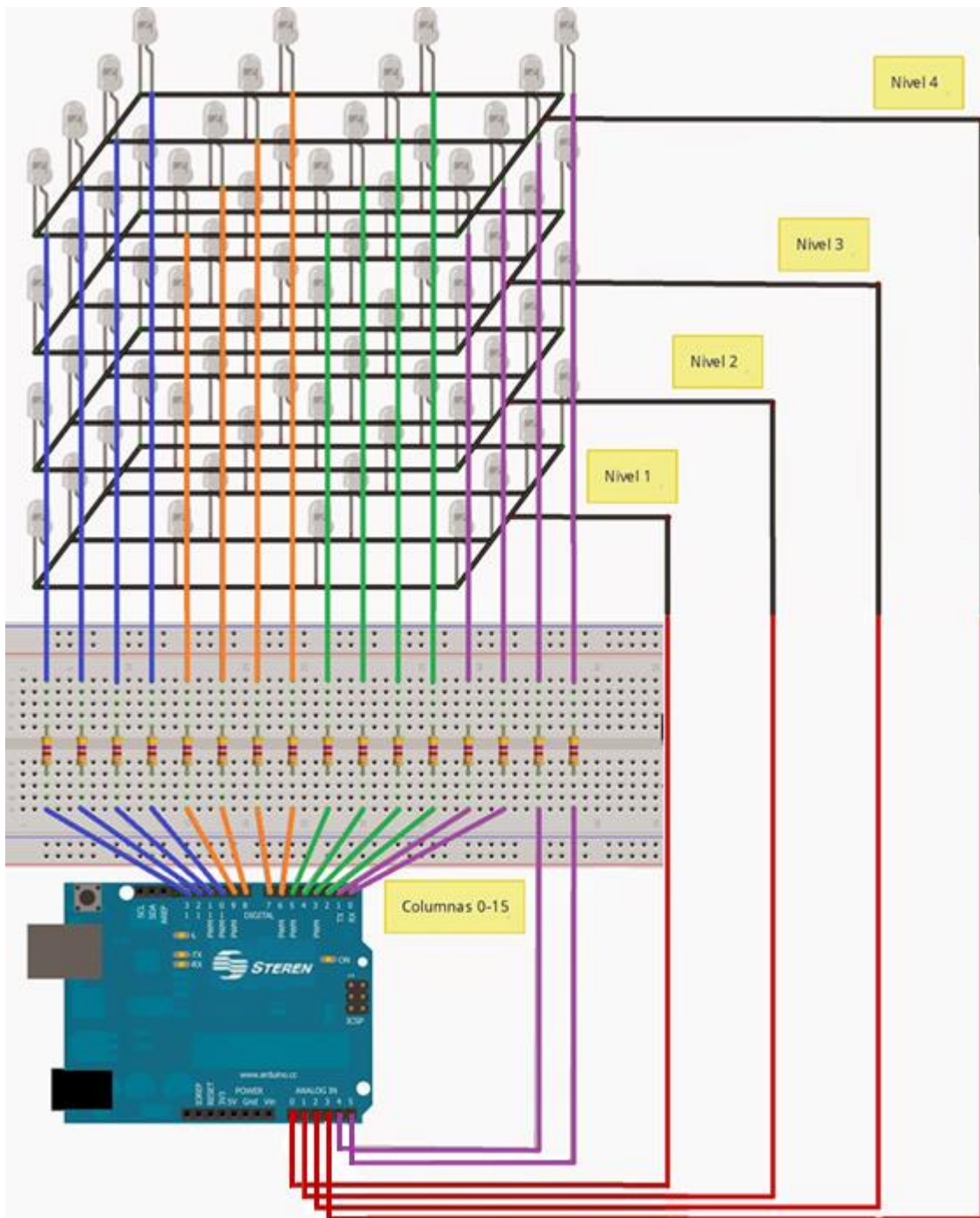
Diagrama de Conexión

Todos los LED de cada nivel están conectados por el lado negativo (cátodo), y las columnas se conectan por el lado positivo (ánodo).



En este ejemplo se observa cómo es la conexión de las cuatro columnas en serie, de los primeros dos niveles, lo cual sirve como guía para conectar el resto de las columnas, las cuales, siguen el mismo procedimiento.





Código

```
int led[] = {13,12,11,10,9,8,7,6,5,4,3,2,1,0,A4,A5};           //Salidas
de las columnas
int lvl[] = {A3,A2,A1,A0};                                   //Salidas
de las filas
int y;                                                       //Variable
para encender las filas

void setup() {                                              //En esta parte se establece la
configuración
    for (int x = 0; x < 16; x ++) {                          //Declarar los pines de las columnas
como salidas
        pinMode(led[x], OUTPUT);
    }
    for (int x = 0; x < 4; x ++) {                            //Declarar los pines de las filas como
salidas
        pinMode(lvl[x], OUTPUT);
    }
}
void loop() {                                               //En esta parte se repite la secuencia infinitas veces
    for (int x = 0; x < 4; x++) {                             //Cada animación se repite 4 veces
        uxu(); //Se llama a la función uxu
    }
    for (int x = 0; x < 4; x++) {
        pxp(); //Se llama a la función pxp
    }
    for (int x = 0; x < 4; x++) {
        cubito(); //Se llama a la función cubito
    }
    for (int x = 0; x < 4; x++) {
        cubote(); //Se llama a la función cubote
    }
    for (int x = 0; x < 4; x++) {
        itz(); //Se llama a la función itz
    }
    for (int x = 0; x < 4; x++) {
        ser(); //Se llama a la función ser
    }
}
//Esta función permite establecer valores de encendido y apagado a cada columna,
donde 1 es encendido y 0 es apagado
void LED (int h, int i, int j, int k, int l, int m, int n, int o, int p, int q,
int r, int s, int t, int u, int v, int w) {
    digitalWrite (led[0], h);
    digitalWrite (led[1], i);
    digitalWrite (led[2], j);
    digitalWrite (led[3], k);
    digitalWrite (led[4], l);
    digitalWrite (led[5], m);
    digitalWrite (led[6], n);
    digitalWrite (led[7], o);
    digitalWrite (led[8], p);
    digitalWrite (led[9], q);
    digitalWrite (led[10], r);
    digitalWrite (led[11], s);
    digitalWrite (led[12], t);
    digitalWrite (led[13], u);
    digitalWrite (led[14], v);
    digitalWrite (led[15], w);
}
void LVL (int h, int i, int j, int k) {                    //Esta función permite establecer
valores de encendido y apagado a cada fila
```

```

digitalWrite (lvl[0], h);
digitalWrite (lvl[1], i);
digitalWrite (lvl[2], j);
digitalWrite (lvl[3], k);
}
void level() { //En esta función se establecen los valores para las filas
donde con un 1 los leds no encienden y con 0 encienden
switch (y) {
case 0: LVL (0, 1, 1, 1); break; //La primera fila encendida
case 1: LVL (1, 0, 1, 1); break; //La segunda fila encendida
case 2: LVL (1, 1, 0, 1); break; //La tercera fila encendida
case 3: LVL (1, 1, 1, 0); break; //La cuarta fila encendida
case 4: LVL (0, 1, 1, 0); break; //Las filas exteriores encendidas
case 5: LVL (1, 0, 0, 1); break; //Las filas interiores encendidas
case 6: LVL (0, 0, 0, 0); break; //Todas las filas encendidas
case 7: LVL (0, 0, 0, 1); break; //Todas excepto la última
case 8: LVL (0, 0, 1, 1); break; //Las primeras dos
case 9: LVL (1, 1, 0, 0); break; //Las últimas dos
}
}

//En esta animación prende led por led
void uxu() {
int t = 100; //Esta variable determina el tiempo que espera el Arduino para
realizar la siguiente acción
for (int j = 0; j < 4; j++) { //Este for se utiliza para pasar de una fila a
otra
y = j; level();
LED (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1); delay(t);
}
}

//En esta animación prende fila por placa
void pxp() {
int t = 500; //Esta variable determina el tiempo que espera el Arduino para
realizar la siguiente acción
for (int j = 0; j < 4; j++) { //Este for se utiliza para pasar de una fila a
otra
y = j; level();
LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
}
}

//En esta animación prende un cubo en el centro y después prende un cubo en el
exterior
void cubito() {
int t = 500; //Esta variable determina el tiempo que espera el Arduino para
realizar la siguiente acción

```

```

    int a = 50; //Esta variable determina las veces que se repetirá una secuencia
    para formar una imagen que no vibre
    for (int x = 0; x < a; x++) { //este for se utiliza para crear una imagen que
    no vibre
        y = 4; level();
        LED (1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1); delay(5);
        y = 5; level();
        LED (1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1); delay(5);
    }
    y = 5; level();
    LED (0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0); delay(t);
    for (int x = 0; x < a; x++) {
        y = 4; level();
        LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(5);
        y = 5; level();
        LED (1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1); delay(5);
    }
    y = 5; level();
    LED (0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0); delay(t);
}

//En esta animación prende fila por fila hasta formar un cubo en el exterior
void cubote() {
    //Prende una fila a la vez para formar un cubo sin centro
    int t = 500;
    int a = 50;
    y = 6; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
    LED (1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
    LED (1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1); delay(t);
    LED (1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1); delay(t);
    LED (1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1); delay(t);
    for (int x = 0; x < a; x++) {
        y = 3; level();
        LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(5);
        y = 7; level();
        LED (1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1); delay(5);
    }
    for (int x = 0; x < a; x++) {
        y = 4; level();
        LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(5);
        y = 5; level();
        LED (1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1); delay(5);
    }
}

//En esta animación prende un cubo de 4 leds que se desplaza en el cubo grande
void itz() {
    int t = 200;
    y = 8; level();
    LED (1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
    LED (0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
    LED (0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0); delay(t);
    y = 6; level();
    LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
    y = 8; level();
    LED (0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0); delay(t);
    LED (0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0); delay(t);
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1); delay(t);
    y = 6; level();
    LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
    y = 8; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1); delay(t);
}

```

```

LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0); delay(t);
y = 6; level();
LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
y = 8; level();
LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0); delay(t);
y = 5; level();
LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0); delay(t);
y = 9; level();
LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0); delay(t);
y = 6; level();
LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
y = 9; level();
LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1); delay(t);
y = 6; level();
LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
y = 9; level();
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1); delay(t);
LED (0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0); delay(t);
LED (0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
y = 6; level();
LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
y = 9; level();
LED (0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
y = 6; level();
LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
y = 9; level();
LED (1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
y = 5; level();
LED (1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
y = 8; level();
LED (1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
y = 6; level();
LED (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1); delay(t);
}

//En esta animación prende una serie de tres leds que recorren todo el cubo
void ser() {
int t = 100;
int a = 10;
int b = 6.5;
for (int x = 0; x < b ; x++) {
y = 0; level();
LED (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
y = 1; level();
LED (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
y = 2; level();
LED (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < b ; x++) {
y = 0; level();
LED (1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
y = 1; level();
LED (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
y = 0; level();
LED (1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
}

```

```

LED (0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0); delay(t);
for (int x = 0; x < a ; x++) {
    y = 0; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0); delay(5);
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 0; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1); delay(5);
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0); delay(5);
}
y = 1; level();
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0); delay(t);
for (int x = 0; x < a ; x++) {
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0); delay(5);
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0); delay(5);
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0); delay(5);
}
y = 2; level();
LED (0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0); delay(t);
for (int x = 0; x < a ; x++) {
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0); delay(5);
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < b ; x++) {
    y = 0; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0); delay(5);
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0); delay(5);
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 0; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0); delay(5);
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 0; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0); delay(5);
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0); delay(5);
}

```



```

for (int x = 0; x < a ; x++) {
    y = 0; level();
    LED (0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 1; level();
    LED (0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 0; level();
    LED (0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 1; level();
    LED (0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 1; level();
    LED (0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < b ; x++) {
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 3; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 3; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0); delay(5);
}
y = 3; level();
LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0); delay(t);
for (int x = 0; x < a ; x++) {
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0); delay(5);
    y = 3; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < b ; x++) {
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0); delay(5);
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0); delay(5);
    y = 3; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 1; level();
    LED (0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0); delay(5);
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0); delay(5);
}
y = 1; level();
LED (0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0); delay(t);
LED (0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
for (int x = 0; x < a ; x++) {
    y = 1; level();
    LED (0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 2; level();
}

```

```

    LED (0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 1; level();
    LED (0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 2; level();
    LED (0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
y = 2; level();
LED (0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(t);
for (int x = 0; x < a ; x++) {
    y = 2; level();
    LED (0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 3; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 2; level();
    LED (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 3; level();
    LED (0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
y = 3; level();
LED (0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
LED (1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(t);
for (int x = 0; x < a ; x++) {
    y = 2; level();
    LED (0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 3; level();
    LED (1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 2; level();
    LED (1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 3; level();
    LED (0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
for (int x = 0; x < a ; x++) {
    y = 1; level();
    LED (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
    y = 2; level();
    LED (1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); delay(5);
}
}
}

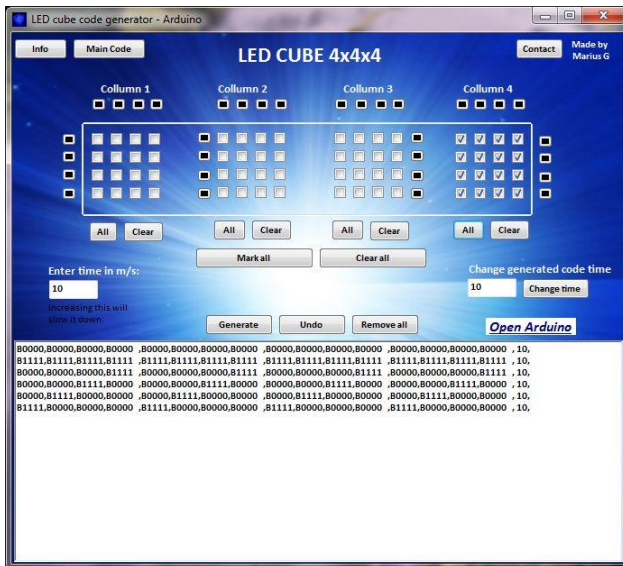
```

Conclusión

En esta práctica se pudo observar el funcionamiento de las secuencias que se implementaron en el programa a través del cubo LED, al principio se presentan dificultades para crear las secuencias, pero hay diferentes programas para crearlo; una recomendación puede ser *LED Cube Code Generator*, que encuentras en línea.

Cómo funciona el programa generador de código:

Hay 4 cuadros con casillas de verificación. Estos cuadros se llaman plano 1, 2, 3 y 4. Los planos representan el cubo como si lo miraras directamente desde el frente, por lo que estos son los cuatro planos verticales del cubo. Esto es para que sea más fácil visualizar tus patrones cuando los haces



Para hacer un patrón, simplemente marca las casillas que representan los LED que deseas iluminar y presiona el botón "Generar". Ahora verás una línea de código publicada en el cuadro de texto a continuación. Si deseas que el código sea más rápido o más lento, puedes ingresar un cierto tiempo en el cuadro "Ingresar tiempo en m / s" antes de generar el código, o cambiarlo más tarde con el cuadro "Cambiar tiempo del código generado".

Así que un código generado haría que los LED se enciendan por esa cierta cantidad de tiempo, y para hacer un patrón simplemente repita este proceso. Hay algunas casillas negras al lado de las casillas de verificación, estas son para marcar la línea completa de LED para ahorrar tiempo sin tener que hacer clic en todas las casillas de verificación.

