



UNIVERSIDAD NACIONAL  
AUTÓNOMA DE MÉXICO  
FACULTAD DE INGENIERÍA



ASIGNATURA: ESTRUCTURA DE DATOS Y  
ALGORÍTMOS I

GRUPO: 17

ALUMNO: ACOSTA RODRÍGUEZ EDER  
ALBERTO

PROFESOR: MARCO ANTONIO MARTÍNEZ  
QUINTANA

PROYECTO: "AutoHome"

**Objetivo:**

El alumno implementará estructuras de datos lineales para la elaboración de este proyecto, puesto esto se fortalecerá la diferenciación entre cada una, así como su uso, declaración, sintaxis e implementación para resolver problemáticas que se susciten en la vida cotidiana.

**Alcance del proyecto:**

AutoHome es un proyecto creado con la finalidad de poder comprender las estructuras de Datos, condicionales simples y complejas y así como también los procesos que implican las funciones externas para poder implementarlas.

De manera más precisa, poder llevar a cabo un proyecto que en un futuro no muy lejano podría hacerse realidad.

**Introducción:**

Los avances tecnológicos<sup>1</sup> siempre han sido importantes para la humanidad, sin lugar a duda la comunicación es uno de los elementos fundamentales ya que siempre hemos buscado manera de reducir el tiempo para poder comunicarnos tanto para un simple un mensaje de texto o algo más complejo como el envío de objetos (documentos, imágenes, videos, links, etc.).

En la actualidad parece muy sencillo comprar algo por internet y que llegue hasta la comodidad de tu casa sin que te veas en la necesidad de desplazarte, sin embargo, para hacer esto pasaron varios años y demasiados avances tecnológicos. Sin duda los avances tecnológicos han cambiado nuestra vida y lo seguirán haciendo, por siempre.

Teniendo esto de contexto, el proyecto surgió de la idea de poder tener un mayor control de tu hogar con solo unas cuantas líneas de código y así poder lograr este objetivo.

**Desarrollo:**

Si bien, al inicio no fue tarea sencilla el poder utilizar lo aprendido en los primeros dos semestres de la carrera, logré juntar e implementarlos de manera correcta.

Siendo así que fue de sumamente importante para utilización de Arduino cono IDE y así mejorar el rendimiento de cada una de

---

<sup>1</sup>Avances Tecnológicos. Aplicaciones y funciones que han adquirido los dispositivos tecnológicos a través del tiempo.

las rutinas a realizar por los sensores, LED, y condiciones utilizadas.



Declaramos funciones externas al "void loop" preestablecido siendo así que las convocamos al momento de que se cumpla una condición múltiple.

***Algoritmo principal:***

- 1 Ingresar cadena de 6 caracteres.
- 2 Si Clave = Clave\_Correcta.
  - 2.1 Mostrar Selecciona una tarea a realizar por AutoHome.
  - 2.2 Mostrar A) Sensor LDR.
  - 2.3 Mostrar B) Sensor del Garage.
  - 2.4 Mostrar C) Abrir la puerta principal.
  - 2.5 Ingresar una letra.
  - 2.6 Si op\_seleccionada = 'A'.
    - 2.6.1 Realizar función A.
  - 2.7 En caso contrario.
    - 2.7.1 Si op\_seleccionada = 'B'.
      - 2.7.1.1 Realizar función B.
    - 2.7.2 En caso contrario.
      - 2.7.2.1 Si op\_seleccionada = 'C'.
        - 2.7.2.1.1 Girar servo motor a 180.
        - 2.7.2.1.2 Esperar 2 segundos.
        - 2.7.2.1.3 Girar servo motor a 0.
        - 2.7.2.1.4 Esperar 2 segundos.
- 3 En caso contrario.
  - 3.1 Negar acceso al programa.
  - 3.2 Activar alarma.

INICIO

```
pinServo,    pulsomax,    pulsomin,    ledPin,    receptor,
transmisor,  garage_ocupado, garage_disponible, distancia,
tiempo, tecla, menú nivel, valor, Buzzer: enteros
```

```
Tecla, Clave[7], Clave_Correcta[7], op_seleccionada,  
keys[FILAS][COLUMNAS]: carácter
```

```
pinesFilas[FILAS],      pinesColumnas[COLUMNAS],      FILAS,
COLUMNAS, Indice: Byte
```

```
HACER    pinServo=10,    pulsomax=2000,    pulsomin=1000,
ledPin=11, receptor=A1, transmisor=A0, garage_ocupado=A2,
garage_disponible=A3, tecla=0, menu_nivel=0, valor=0,
Buzzer=12, Clave_Correcta[7] = "231019",
op_seleccionada=' ', Indice=0, FILAS=4, COLUMNAS=4,
pinesFilas[FILAS]={9,8,7,6}, pinesColumnas[COLUMNAS]={5,4
,3,2}, keys[FILAS][COLUMNAS] = {
```

 $\{'1', '2', '3', 'A'\},$  $\{ '4', '5', '6', 'B' \},$ 

```
{ '7', '8', '9', 'C' },
```

 $\{ '*', '0', '#', 'D' \}$ 

}

LEER Clave

Si Clave = Clave Correcta ENTONCES

```
ESCRIBIR "Selecciona una tarea a realizar por
AutoHome"
```

ESCRIBIR "A) Sensor LDR"

ESCRIBIR "B) Sensor del Garage"

ESCRIBIR "C) Abrir la puerta principal"

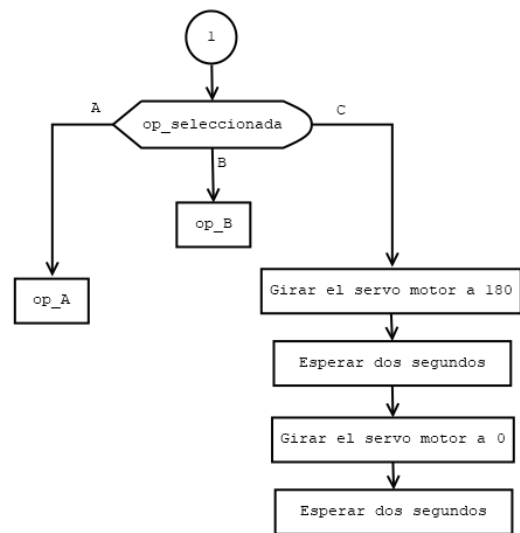
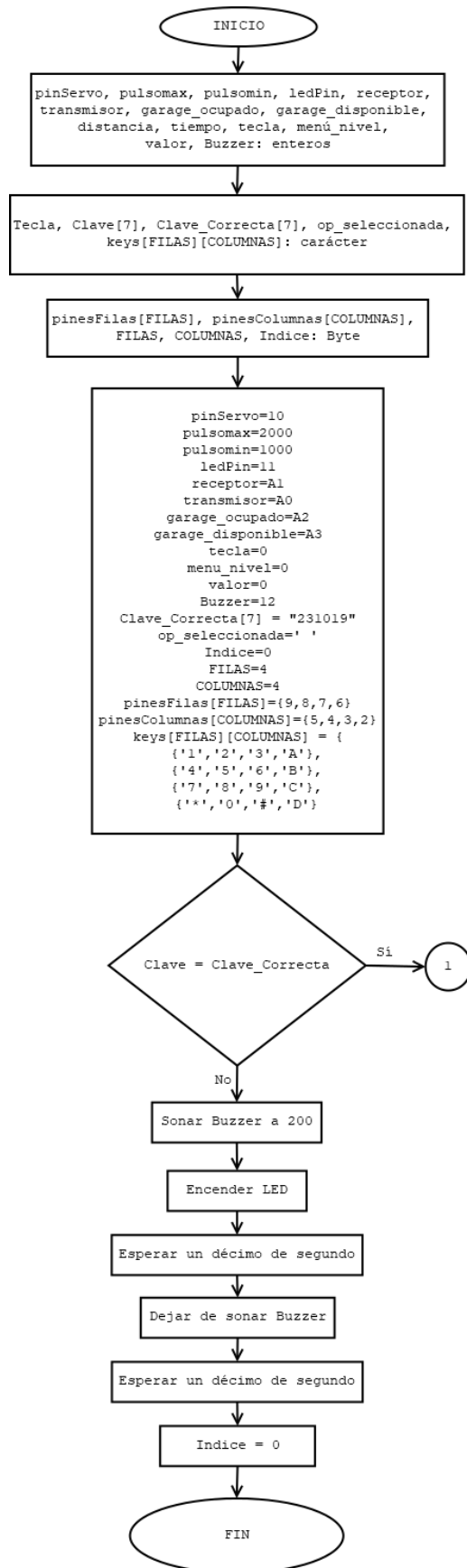
LEER op seleccionada

SELECCIONAR op seleccionada EN

### CASO 'A'

```
HACER op_A
CASO 'B'
    HACER op_B
CASO 'C'
    HACER girar el servo motor a 180
    HACER una espera de 2 segundos
    HACER girar el servo motor a 0
    HACER una espera de 2 segundos
FIN SELECCIONAR
FIN SI
EN CASO CONTRARIO
    HACER sonar el Buzzer a 200
    HACER encendido de LED
    HACER una espera de un décimo de segundo
    HACER que pare de sonar el Buzzer
    HACER una espera de un décimo de segundo
    HACER Indice = 0
FIN EN CASO CONTRARIO
FIN
```

### ***Diagrama de Flujo Principal***



**Algoritmo función A:**

```
1 Mientras tecla ≠ '#' .
2 Realizar lectura de sensor LDR.
3 Realizar ldrValue/10.
4 Si ldrValue >= 0 && ldrValue <= 20.
    4.1 Realizar el apagado del LED.
    4.2 Esperar medio segundo.
5 En caso contrario.
    5.1 Si ldrValue >= 20
        5.1.1 Realizar el encendido del LED.
        5.1.2 Esperar medio segundo.
6 Realizar menú_nivel = 0.
```

**Pseudocódigo función A:**

INICIO

ldrPin, ldrValue: entero

HACER valor = 0, tecla = 0

MIENTRAS tecla ≠ '#' ENTONCES

SI tecla ≠ NO\_KEY ENTONCES

HACER ldrValue = analogRead(ldrPin)

HACER ldrValue = ldrValue/10

FIN SI

SI ldrValue >= 0 & ldrValue <= 20 ENTONCES

HACER el apagado del LED

HACER una espera de medio segundo

FIN SI

EN CASO CONTRARIO

SI ldrValue >= 20 ENTONCES

HACER el encendido del LED

HACER una espera de medio segundo

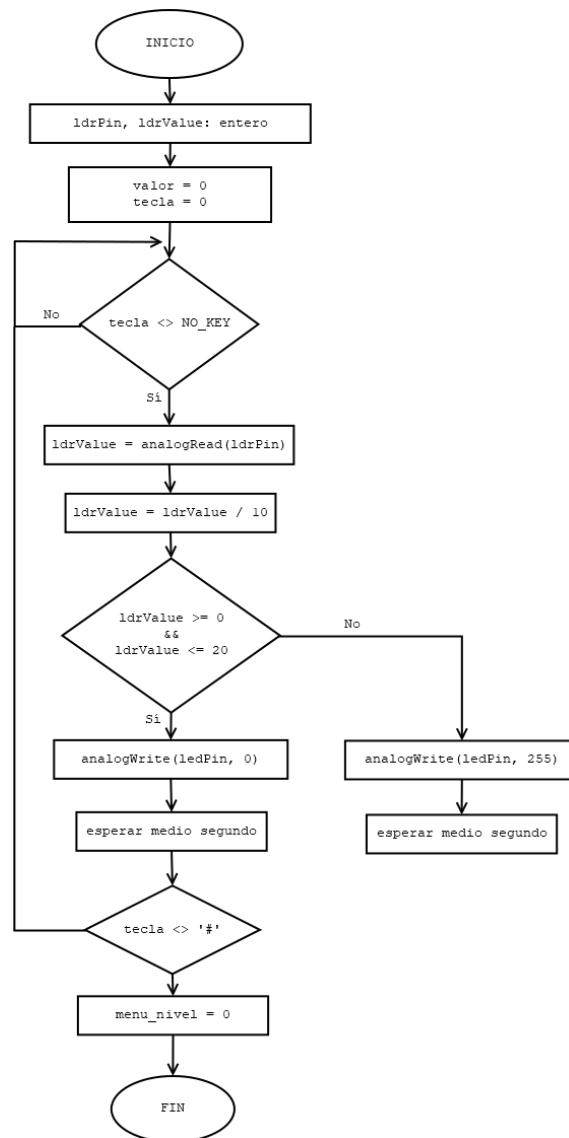
FIN EN CASO CONTRARIO

FIN MIENTRAS

HACER menú\_nivel = 0

FIN

**Diagrama de flujo función A:**



**Algoritmo función B:**

1 Mientras tecla ≠ '#'.

2 Realizar el encendido de transmisor.



```
3 Realizar una espera de un microsegundo.
4 Realizar el apagado de transmisor.
5 Realizar tiempo=analogRead(receptor, 255).
6 Realizar distancia = (tiempo/2)/29.
7 Si distancia = 0.
    7.1 Realizar el encendido de garage_ocupado.
    7.2 Realizar el apagado de garage_disponible.
8 En caso contrario.
    8.1 Realizar el apagado de garage_ocupado.
    8.2 Realizar el encendido de garage_disponible.
9 Realizar menú_nivel = 0.
```

***Pseudocódigo función B:***

INICIO

HACER valor = 0, tecla = 0

MIENTRAS tecla ≠ '#' ENTONCES

SI tecla ≠ NO\_KEY ENTONCES

HACER el encendido del transmisor

HACER una espera de un microsegundo

HACER el apagado del transmisor

HACER tiempo=analogRead(receptor,255)

HACER distancia= (tiempo/2)/29

SI distancia = 0 ENTONCES

HACER el encendido de garage\_ocupado

HACER el apagado de garage\_disponible

FIN SI

EN CASO CONTRARIO

HACER el apagado de garage\_ocupado

HACER el encendido de garage\_disponible

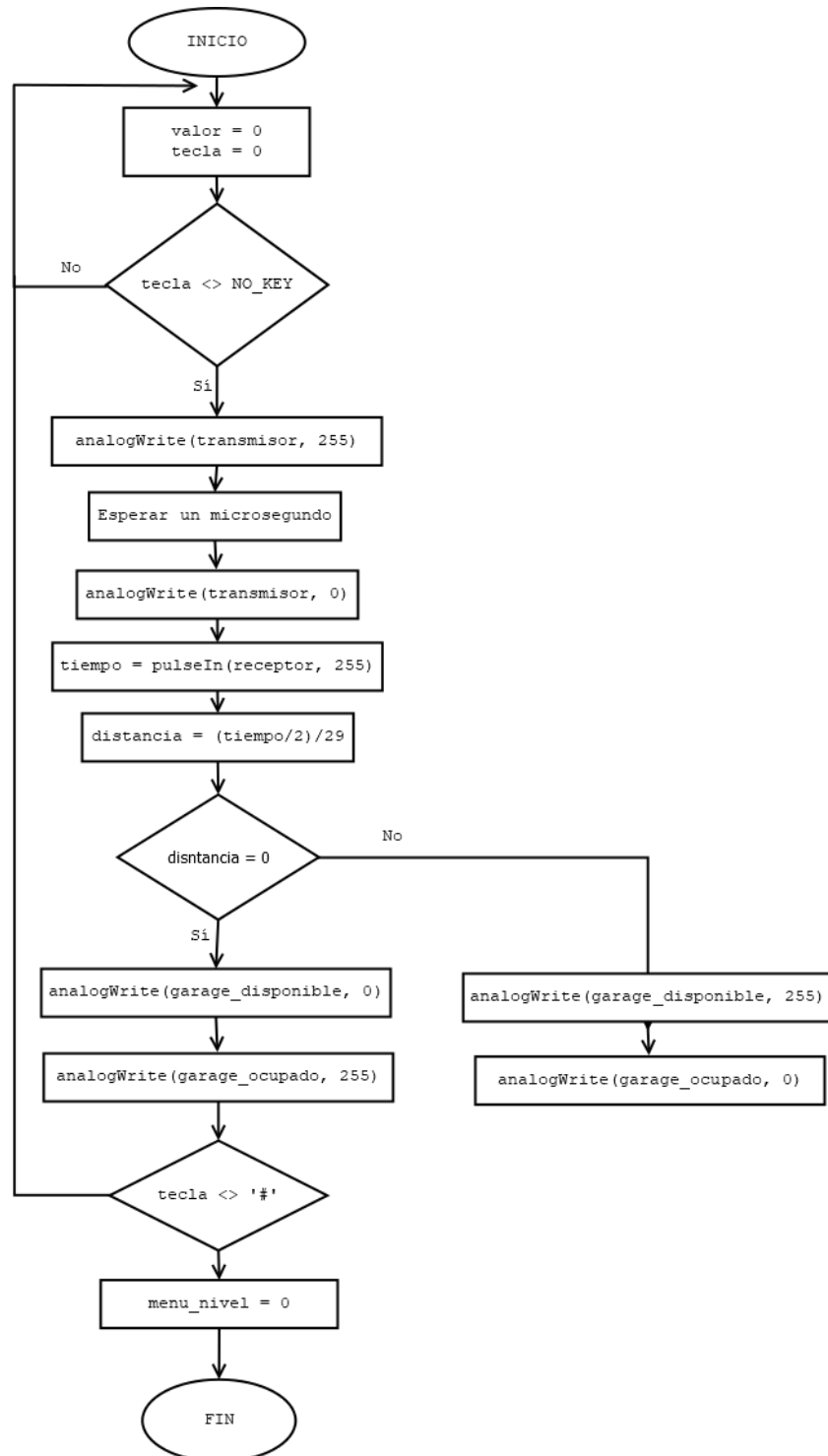
FIN EN CASO CONTRARIO

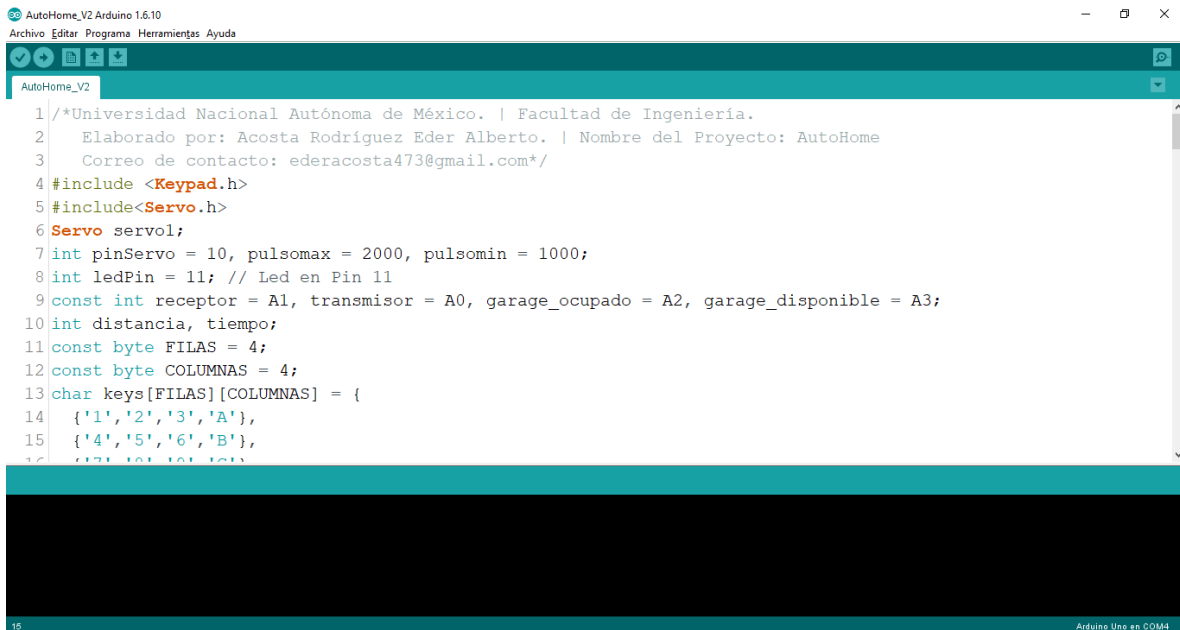
FIN MIENTRAS

HACER menú\_nivel = 0

FIN

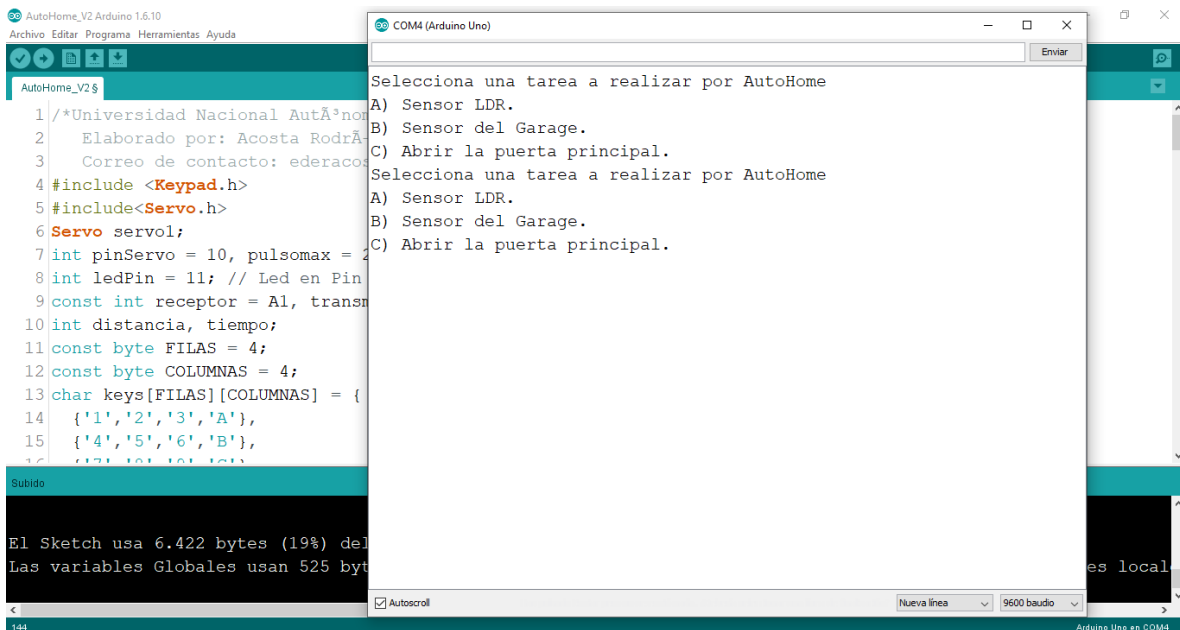
**Diagrama de flujo función B:**





```
1 /*Universidad Nacional Autónoma de México. | Facultad de Ingeniería.
2   Elaborado por: Acosta Rodríguez Eder Alberto. | Nombre del Proyecto: AutoHome
3   Correo de contacto: ederacosta473@gmail.com*/
4 #include <Keypad.h>
5 #include<Servo.h>
6 Servo servol;
7 int pinServo = 10, pulsomax = 2000, pulsomin = 1000;
8 int ledPin = 11; // Led en Pin 11
9 const int receptor = A1, transmisor = A0, garage_ocupado = A2, garage_disponible = A3;
10 int distancia, tiempo;
11 const byte FILAS = 4;
12 const byte COLUMNAS = 4;
13 char keys[FILAS][COLUMNAS] = {
14   {'1','2','3','A'},
15   {'4','5','6','B'},
16   {'7','8','9','C'},
17   {'*','0','#','.'}}
```

## Resultados:



```
Subido
El Sketch usa 6.422 bytes (19%) del
Las variables Globales usan 525 byt

COM4 (Arduino Uno)
Enviar
Selecciona una tarea a realizar por AutoHome
A) Sensor LDR.
B) Sensor del Garage.
C) Abrir la puerta principal.
Selecciona una tarea a realizar por AutoHome
A) Sensor LDR.
B) Sensor del Garage.
C) Abrir la puerta principal.
```

Link de video:

<https://github.com/Eder-Berno/AutoHome/blob/master/AutoHome%20Video%20Demostrativo.mp4>

**Conclusiones:**

Aunque al inicio fue complejo la unificación de todas las funciones, se pudo cumplir el objetivo de poder implementar las diversas estructuras de datos y así poder diferenciar entre ellas, para poder optimizar una codificación que solucionó una problemática.

**Referencias:**

- [https://www.youtube.com/watch?v=mlw3APOUt8U&ab\\_channel=BitwiseAr](https://www.youtube.com/watch?v=mlw3APOUt8U&ab_channel=BitwiseAr)
- [https://www.youtube.com/watch?v=6bPVZg17vKc&t=637s&ab\\_channel=BitwiseAr](https://www.youtube.com/watch?v=6bPVZg17vKc&t=637s&ab_channel=BitwiseAr)
- [https://www.youtube.com/watch?v=qKgWtImVilg&ab\\_channel=AgriculturaElectronica](https://www.youtube.com/watch?v=qKgWtImVilg&ab_channel=AgriculturaElectronica)
- Cuaderno de Apuntes de Fundamentos de Programación de la Facultad de Ingeniería, perteneciente al semestre 2020-1 y cuyo dueño es Acosta Rodríguez Eder Alberto.
- Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest y Clifford Stein; The MIT Press; 2009, tercera edición.
- Código de Teclado Matricial y Servo Motor proporcionado por un curso que tomó el alumno en ARACT UNAM en 2018.