



**Mi Universidad**

**EXAMEN**

*Sofia Pereyra Orantes*

*Examen Final*

*Programación Lógica*

*Ojeda Trujillo Juan José*

*Ingeniería en Sistemas Computacionales*

*Cuatrimestre 8*

*Comitan de Dominguez, Chiapas a 13 de Abril de 2025*

## Pregunta 1.

```
4 public class ServidorControl {
5     private static boolean carga1 = false;
6     private static boolean carga2 = false;
7     private static boolean carga3 = false;
8     private static int anguloServo = 0;
9
10    public static void main(String[] args) {
11        int puerto = 5000;
12
13        try {ServerSocket servidor = new ServerSocket(puerto) {
14            System.out.println("Servidor de control iniciado en el puerto " + puerto);
15
16            while (true) {
17                Socket cliente = servidor.accept();
18                BufferedReader entrada = new BufferedReader(new InputStreamReader(cliente.getInputStream()));
19                PrintWriter salida = new PrintWriter(cliente.getOutputStream(), true);
20
21                String mensaje = entrada.readLine();
22                String respuesta = procesarComando(mensaje);
23                salida.println(respuesta);
24
25                cliente.close();
26            }
27        } catch (IOException e) {
28            e.printStackTrace();
29        }
30    }
31
32    public static String procesarComando(String comando) {
33        if (comando == null) return "Comando vacío";
34
35        comando = comando.toUpperCase().trim();
36
37        if (comando.startsWith("ENCENDER")) {
38            int carga = Integer.parseInt(comando.split(" ")[1]);
39            return encenderCarga(carga);
40        } else if (comando.startsWith("APAGAR")) {
41            int carga = Integer.parseInt(comando.split(" ")[1]);
42            return apagarCarga(carga);
43        } else if (comando.startsWith("SERVO")) {
44            int angulo = Integer.parseInt(comando.split(" ")[1]);
45            return moverServo(angulo);
46        } else {
47            return "Comando no reconocido";
48        }
49    }
50 }
51
```

```
public static String encenderCarga(int num) {
    switch (num) {
        case 1 -> carga1 = true;
        case 2 -> carga2 = true;
        case 3 -> carga3 = true;
        default -> {
            return "Carga inválida";
        }
    }
    return "Carga " + num + " encendida.";
}

public static String apagarCarga(int num) {
    switch (num) {
        case 1 -> carga1 = false;
        case 2 -> carga2 = false;
        case 3 -> carga3 = false;
        default -> {
            return "Carga inválida";
        }
    }
    return "Carga " + num + " apagada.";
}

public static String moverServo(int angulo) {
    if (angulo < 0 || angulo > 180) {
        return "Ángulo fuera de rango (0-180)";
    }
    anguloServo = angulo;
    return "Servomotor movido a " + angulo + "°";
}
}
```

## Pregunta 2.

```
17 Socket cliente = servidor.accept();
18 BufferedReader entrada = new BufferedReader(new InputStreamReader(cliente.getInputStream()));
19 PrintWriter salida = new PrintWriter(cliente.getOutputStream(), true);
20
21 String mensaje = entrada.readLine();
22 String respuesta = procesarComando(mensaje);
23 salida.println(respuesta);
24
```

## Pregunta 3.

```
33 public static String procesarComando(String comando) {
34     if (comando == null) return "Comando vacío";
35
36     comando = comando.toUpperCase().trim();
37
38     if (comando.startsWith("ENCENDER")) {
39         int carga = Integer.parseInt(comando.split(" ")[1]);
40         return encenderCarga(carga);
41     } else if (comando.startsWith("APAGAR")) {
42         int carga = Integer.parseInt(comando.split(" ")[1]);
43         return apagarCarga(carga);
44     } else if (comando.startsWith("SERVO")) {
45         int angulo = Integer.parseInt(comando.split(" ")[1]);
46         return moverServo(angulo);
47     } else {
48         return "Comando no reconocido";
49     }
50 }
51
```

```
public static String encenderCarga(int num) {
    switch (num) {
        case 1 -> carga1 = true;
        case 2 -> carga2 = true;
        case 3 -> carga3 = true;
        default -> {
            return "Carga inválida";
        }
    }
    return "Carga " + num + " encendida.";
}
}
```

```
76 public static String moverServo(int angulo) {
77     if (angulo < 0 || angulo > 180) {
78         return "Ángulo fuera de rango (0-180)";
79     }
80     anguloServo = angulo;
81     return "Servomotor movido a " + angulo + "°";
82 }
83 }
84
```

## Pregunta 4.

**Comando "SERVO <angulo> ----entrada ---orden de movimiento**

**procesarComando()**----logica de decisión—detecta el comando y llama al controlador

**moverServo(angulo)**---control principal---valida y hace el movimiento

**anguloServo** ----guarda la posición actual del del servomotor

**System.out/return** ---salida---informa el resultado al cliente

Pregunta 5.

Las hélices ,junto con los motores eléctricos.

1. Los motores hacen girar las hélices muy rápido.
2. Al girar, las hélices empujan el aire hacia abajo.
3. Esto genera una fuerza hacia arriba, llamada sustentación o empuje, que levanta el dron del suelo.

Pregunta 6.

--procesarComando(String comando) >Lógica de decisión , determina que activar y como

--encenderCarga(),apagarCarga() >Modifican el estado de la carga

--moverServo(int angulo) >cambia posicion el servomotor

-variables carga1 y anguloServo > estado interno>guardan el estado actual de activación

Pregunta 7.

```
52 public static String encenderCarga(int num) {
53     switch (num) {
54         case 1 -> carga1 = true;
55         case 2 -> carga2 = true;
56         case 3 -> carga3 = true;
57         default -> {
58             return "Carga inválida";
59         }
60     }
61     return "Carga " + num + " encendida.";
62 }
63
64 public static String apagarCarga(int num) {
65     switch (num) {
66         case 1 -> carga1 = false;
67         case 2 -> carga2 = false;
68         case 3 -> carga3 = false;
69         default -> {
70             return "Carga inválida";
71         }
72     }
73     return "Carga " + num + " apagada.";
74 }
```

```
33 public static String procesarComando(String comando) { //invoco los metodos
34     if (comando == null) return "Comando vacio";
35
36     comando = comando.toUpperCase().trim();
37
38     if (comando.startsWith("ENCENDER")) {
39         int carga = Integer.parseInt(comando.split(" ")[1]);
40         return encenderCarga(carga);
41     } else if (comando.startsWith("APAGAR")) {
42         int carga = Integer.parseInt(comando.split(" ")[1]);
43         return apagarCarga(carga);
44     } else if (comando.startsWith("SERVO")) {
45         int angulo = Integer.parseInt(comando.split(" ")[1]);
46         return moverServo(angulo);
47     } else {
48         return "Comando no reconocido";
49     }
50 }
51 }
```

```
private static boolean carga1 = false;
private static boolean carga2 = false; //variables que representan las cargas
private static boolean carga3 = false;
private static int anguloServo = 0;
```

Pregunta 8.

Se podría controlar la automatización de una casa como por ejemplo las luces(cargas 1,2,3), las persianas o puertas automáticas (con el servomotor)