



## GUIA # 03

### ESTRUCTURAS REPETITIVAS

#### CONCEPTUALIZACION

##### ESTRUCTURAS REPETITIVAS

Hasta el momento, se ha visto la estructura permite tomar decisiones, pero en caso que se tenga que repetir alguna acción un determinado número de veces, si se realiza de forma lineal, sin embargo, en la mayoría de los casos, el número de instrucciones varia, además que el programa no sería óptimo. Para esto existe las estructuras repetitivas.

Las estructuras repetitivas se utilizan para realizar un determinado tipo de instrucciones en un número finito de veces. Se caracterizan por tener un punto inicial de partida, una condición la cual se encarga de ejecutar un número determinado de acciones hasta que esta condición no sea válida (no se cumpla). Las sentencias repetitivas son 3, for, while y do while.

##### **Sentencia For ( Desde/para)**

La sentencia for (para) es un número total de iteraciones que se deben ejecutar una o varias instrucciones. Esta sentencia se caracteriza por tener un contador, una condición que involucra este contador y una sentencia de incremento o decremento dependiendo el caso.

La estructura for es la siguiente:



**para (inicio de iteraciones; límite de iteraciones; incremento/decremento)**

acción 1

acción 2

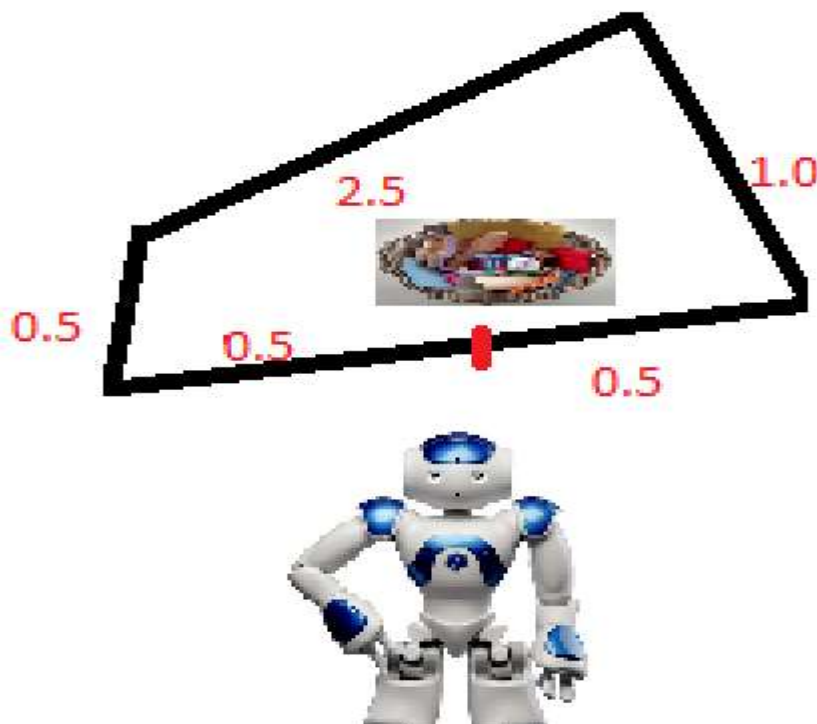
acción 3

....

acción n

**Fin\_para**

**Ejemplo:** *Un robot recorre un circuito de 5 m como se muestra en la figura, en cada recorrido, recoge un objeto al terminar una vuelta. Realice un programa para el cual el robot realice esta acción 10 veces .*



En este programa, el robot debe recorrer el circuito y recoger un objeto dentro de la canasta. Para realizar el programa, se debe tener en cuenta las distancia que se emplean el orden de las acciones serian:



#### INICIO:

Int objeto=0;  
avanzar 0.5 m;  
Girar a la derecha 80°;  
avanzar 0.5 m;  
girar a la derecha 40°  
avanzar 2.5 m;  
girar a la derecha 90°;  
avanzar 1 m;  
girar a la derecha 90°;  
recoge 1 objeto;  
objeto= objeto +1;// con esta contamos los objetos que recoge

#### FIN

Como se puede apreciar, este algoritmo resuelve el problema del recorrido del robot, sin embargo, solo lo hace para una sola vez, el problema nos pide para 10 giros, la solución sería copiar este mismo código un total de 9 veces, convirtiéndose en una tarea un poco tediosa y poco optima, en caso de futuros cambios.

Pero, existen las estructuras repetitivas, para este caso con un ciclo for la solución sería:



INICIO:

**Para (declare objeto=0;objeto >10; objeto +1)**

avanzar 0.5 m;

Girar a la derecha 80°;

avanzar 0.5 m;

girar a la derecha 40°

avanzar 2.5 m;

girar a la derecha 90°;

avanzar 1 m;

girar a la derecha 90°;

recoge 1 objeto;

**fin para**

FIN

Con este código garantizamos que se cumplan las 10 iteraciones por:

- ✓ En la primera parte del for se declara la variable de control, empezando en 0 en este caso ya que no tiene objetos.
- ✓ En la segunda parte, le decimos hasta donde va la sentencia, en este caso hasta que la variable objeto llegue al valor 10.
- ✓ En la ultima parte, hacemos el incremento.

*Actividad: realice una prueba de escritorio al código anterior.*

### **Sentencia While (Mientras que)**

Es otra estructura repetitiva, la cual realiza cierto número de acciones mientras se cumpla una condición. A diferencia del for, no necesariamente necesita un contador, sino que se cumpla una determinada condición ya sea, por ejemplo, que este en un intervalo, que algo sea mayor que otro etc.



### **Mientras que(condición)**

acción 1  
acción 2  
acción 3  
....  
acción n

### **Fin Mientras que**

**Ejemplo:** Retomemos el problema anterior (robot), para esta ocasión, lo haremos en un while, el cual quedaría así:

INICIO:

```
declare objeto =0;  
mientras que (objeto >=10)  
avanzar 0.5 m;  
Girar a la derecha 80°;  
avanzar 0.5 m;  
girar a la derecha 40°  
avanzar 2.5 m;  
girar a la derecha 90°;  
avanzar 1 m;  
girar a la derecha 90°;  
recoge 1 objeto;  
objeto=objeto+1;  
fin mientras que
```

FIN

Para esta sentencia, la variable de control, hay que darle un valor inicial, y la instrucción del incremento, se hace al final, caso adverso al for, el cual la hace al comienzo.

*Analice las dos estructuras repetitivas, explique cuál considera que es más eficiente y por qué.*



## Sentencia do-while (haga mientras que)

Es otra estructura repetitiva, que al igual que el mientras que, se rige por una condición, sin embargo, su comprobación se hace al final de la misma, lo cual garantiza que se ejecuta las acciones 1 vez. Suele utilizarse en menús y confirmación de información. Su estructura es la siguiente:

### Haga

acción 1  
acción 2  
acción 3  
....  
acción n

### Mientras que(condición)

### Fin\_haga Mientras

Retomando el ejemplo anterior del robot:

INICIO:

```
Declare objeto =0;  
haga  
avanzar 0.5 m;  
Girar a la derecha 80°;  
avanzar 0.5 m;  
girar a la derecha 40°  
avanzar 2.5 m;  
girar a la derecha 90°;  
avanzar 1 m;  
girar a la derecha 90°;  
recoge 1 objeto;  
objeto=objeto+1;  
}  
Mientras que (objeto >=10;)  
Fin mientras que
```

FIN



En esta ocasión, realiza la sentencia y por último comprueba, en caso de no cumplirse, sale de la estructura.

Cada una de estas estructuras, a pesar de servir para lo mismo, repetir determinado tipo de acciones, pueden tener un uso específico en cierto tipo de programas. Usted como programador debe saber elegir cual es la más acorde a la situación con la que se enfrente.

### Actividades

*Dado los siguientes problemas, mediante lenguaje natural realice los siguientes ejercicios, realice el procedimiento de resolución de problemas explicado en la guía anterior.*

- a. *La multiplicación se define como la suma de un número un número determinado de veces, por ejemplo  $3 \times 5$  es igual a decir  $3+3+3+3+3 = 15$ . Realice el algoritmo para hacer la multiplicación a través de la suma.*
- b. *Juanito tiene una cuenta de ahorros en la corporación SINFONDOS. Esta corporación otorga a sus ahorradores un interés mensual y un interés anual sobre el saldo del momento. Determinar a partir del saldo de la cuenta, del interés mensual y del interés anual, los ahorros que tendrá después de un número determinado de meses, si él no retira ni deposita dinero en ese tiempo.*

## ACTIVIDADES LÚDICAS

**Problema 01:** En los años 80, existió una serie muy famosa llamada El auto fantástico, el cual se caracterizaba por ser un Auto inteligente. En la parte frontal poseía una serie de luces que encendían y apagaban simultáneamente.



*Ilustración 1 El auto fantástico.*

Realice un prototipo electrónico que simule el procedimiento de estas luces. Realice cada uno de los procedimientos con su respectivo montaje

*(Analice la solución, identifique entradas y salidas, realice el respectivo montaje, guíese por el montaje de los LED hecho en la primera práctica, hágalo para mínimo 6 LED. No avance en la guía hasta no tener el montaje realizado.)*





Para realizar el código de arduino, guíese por este pseudocódigo en lenguaje natural, realice el respectivo código:

### INICIO

```
Para(led=1; led<=6; led +1){  
    Entrada = led;  
}
```

```
Para(led=1; led<=6;led+1){  
    Encienda led;  
    Espere 50  
    Apague led  
}
```

```
Para(led=6; led>=6;led-1){  
    Encienda led;  
    Espere 50  
    Apague led  
}
```

### FIN

NOTA: aquí se especificaron pines del 1 al 6 para ilustrar el ejemplo, usted debe guiarse por su montaje, en su caso, por ejemplo, si su primer led está en 4, ese sería su led 1.

Una vez terminado el código en arduino, enséñeselo a su tutor el código y pruebe su prototipo.

**Actividad:**

- a) *Realice un prototipo electrónico en el cual, se detecta un objeto a cierta distancia, se encienda un led, y al detectar una distancia menor, produzca un sonido. Realice el procedimiento para la resolución de problemas. Para el sonido utilice un pulsador, si posee conexión a internet, explore lo que es este dispositivo y explíquesele a su tutor, en caso contrario, será un compromiso adquirido para la próxima sesión.*



*Ilustración 2 Zumbador*

*(retome la practica 2 del ultrasonido, añada el zumbador e investigue en internet procedimiento para el funcionamiento del zumbador:*

- b. *Ajuste esta practica ahora con una estructura while. Muestre el código a su tutor y explique los cambios con el ejercicio anterior.*



## AUTOEVALUACION

Coloque una X en las siguientes preguntas

1. Con el desarrollo de esta guía reafirmó el concepto de estructuras repetitivas.  
SI \_\_\_\_ NO \_\_\_\_
2. Ha comprendido semejanzas y diferencias entre las diferentes estructuras repetitivas trabajadas en esta guía:  
SI \_\_\_\_ NO \_\_\_\_
3. Con las guías realizadas anteriormente, ha fortalecido los conceptos que posee de programación  
SI \_\_\_\_ NO \_\_\_\_
4. Con el desarrollo de esta guía ha comprendido aún más la programación de la plataforma arduino.  
SI \_\_\_\_ NO \_\_\_\_