

## LOGO – 2: Instrucciones para repeticiones

REPITE 5 [.....]

- Repite 5 veces las instrucciones del corchete
- Las instrucciones se repiten un número fijo de veces

MIENTRAS [condición] [.....]

- Mientras se cumpla la “condición” las instrucciones del corchete se repetirán indefinidamente.
- Cuando la “condición” sea falsa se continuara el programa en la siguiente línea

### Ejemplos con “mientras”

*Como uno siempre es igual a uno este programa se queda ‘colgado’ repitiendo de forma indefinida el contenido del corchete  
Para detener al programa pulsar la tecla ESCape..*

```
para bucleinfinito
  mientras [1=1] [ escribe[infinito] ]
  escribe[ Esto nunca lo veras]
fin
```

*Este programa pide una contraseña.*

*Mientras no sean iguales el texto que introduce el usuario y la contraseña (almacenada en la variable “c”) volverá a preguntar la contraseña una y otra vez, no permitiendo continuar con el programa..*

```
para protegido
  escribe[Dime la contraseña para seguir]
  haz "s leepalabra
  haz "c "contral23
  mientras [NO :s=:c][escribe[Contraseña incorrecta, repitela:] haz "s leepalabra]
  escribe[Solo leeras esto si metes la contraseña correcta]
fin
```

*Programa que preguntá un n° positivo y luego muestra su raíz cuadrada.*

*Observa que si te equivocas y metes un n° negativo el programa se detiene sin que lo hayamos previsto*

*Imagina lo malo que puede ser que se detenga de forma imprevista el programa que controla una central nuclear*

para raizdepositivo

```
  escribe[Dime un numero positivo]
  haz "n leepalabra
  haz "rc raizcuadrada(:n)
  (escribe "Tu "numero "es "el :n "y "su "raiz "es :rc)
Fin
```

*Programa que preguntá un n° positivo y luego muestra su raíz cuadrada.*

**PERO AHORA PROTEGEMOS EL PROGRAMA CONTRA NUMEROS NEGATIVOS**

*Si metes un n° negativo te echa la bronca y te lo preguntará de nuevo hasta que introduzcas uno positivo.*

para raizdepositivomejorado

```
  escribe[Dime un numero positivo]
  haz "n leepalabra
  mientras [:n<0][escribe[Eres un tonto, dime n° posit:] haz "n leepalabra]
  haz "rc raizcuadrada(:n)
  (escribe "Tu "numero "es "el :n "y "su "raiz "es :rc)
Fin
```

## Ahora eres tú el que debes decidir si usar un REPITE o un MIENTRAS

*Programa que pregunta al usuario cuántos números quiere sumar. Luego los pregunta y calcula su suma.*

```
para sumando
  escribe[¿cuántos n°s quieres sumar]
  haz "n leepalabra
  suma=0
  repite :n [ escribe[dime n°] haz "x leepalabra haz "suma :suma+:x]
  (escribe "la "suma "es :suma)
Fin
```

Aquí se usó el REPITE, ya que hay que repetir la pregunta exactamente n veces

*Programa que pregunta una y otra vez números. Se detiene al introducir uno negativo. Al final dice cuántos números has introducido.*

```
para contando
  escribe[Dime un n°(negativo para parar)=]
  haz "x leepalabra
  haz "contador 0
  mientras [NO :x<0] [haz "contador :contador+1 escribe[Dime un n°
    (negativo para parar)=] haz "x leepalabra]
  (escribe "Has "dicho :contador "numeros)
Fin
```

Aquí se usó el MIENTRAS, ya que hay que preguntar números, pero no sabemos cuántos, podrían ser 3, 1000 o ninguno.

Observa que para contar se utiliza una variable que inicialmente vale 0. Luego se va incrementando de 1 en 1.

*Programa que pregunta 5 números y cuenta y cuántos son positivos o cero y cuantos son negativos*

```
para positivosynegativos
  haz "p 0
  haz "n 0
  repite 5 [escribe[dime n°] haz "x leepalabra si :x<0 [haz "n :n+1] [haz "p :p+1]]
  (escribe "Haz :p "positivos "o "cero "y :n "negativos)
Fin
```

Observa dos cosas:

- Como hay que repetir algo exactamente 5 veces utiliza un REPITE 5 [...]
- Hay dos contadores, uno de n°s positivos y otro de negativos.
  - o Si el valor de x es menor que cero incrementa el contador de negativos
  - o y si no, incrementa el contador de positivos