

 <b>XALOC</b>	Nombre:
<b>4º ESO – Programación. Ejercicios con bucles (While)</b>	
Departamento de Tecnología	Fecha:

Seguimos trabajando con bucles, en esta ocasión con el **while**. Esta instrucción repite un bloque de código mientras se cumpla una condición. Las diferencias entre for y while no se limitan a la sintaxis, también a su “comportamiento” por ejemplo, el while no incluye una variable que se incremente, si conviene, la tendremos que crear. ¿Cuándo conviene utilizar for y cuándo while?. Es la pregunta que uno se hace cuándo empieza a utilizar bucles. La instrucción for se utiliza cuando conoces el final del recorrido, es decir, cuando conoces el valor de los rangos. Mientras que while, la condición puede ser indeterminada y es el usuario el que decide cuándo salir del bucle, por ejemplo, al pulsar una tecla.

Consulta los conceptos que se explican en el Sway de la asignatura correspondiente a bucles y repasa los ejercicios y ejemplos, te ayudarán a entender mejor el funcionamiento de bucles.

Como se ha indicado anteriormente, la sintaxis de la instrucción **while** utiliza una condición que determina cuántas veces se repetirá un bloque de código, recuerda los operadores vistos en la instrucción if.

Ejemplos:

while var<=3:

while var>5 and var <10:

Es frecuente que, al testear un programa, se produzca un bucle infinito. Esto es debido a que la condición indicada en el while está mal establecida. Pulsa **Ctrl + C** para detener el programa y comprueba con un breakpoint el funcionamiento del programa para encontrar dónde está el error.

En los siguientes ejercicios, hay actividades en que tendrás que utilizar únicamente while o simultáneamente con for.

**50. Realiza un programa que de los buenos días 3 veces. Con While**

	Buenos días
--	-------------

**51. A partir del programa anterior, modifica el código para que sea el usuario quién introduzca el número de veces que desea que repita la frase Buenos días. Con While**

2	buenos días buenos días
5	buenos días buenos días buenos días buenos días buenos días

**52. Realiza un programa que sume dos números enteros por teclado y presente el resultado por pantalla. El programa preguntará si deseas o no repetir la operación. Con While**

5	El resultado de la suma es: 11
6	Deseas repetir la operación s/n:
s	
7	El resultado de la suma es: 13
6	Deseas repetir la operación s/n:
s	
5	El resultado de la suma es: 10
5	Deseas repetir la operación s/n:
n	Mensaje: "Programa finalizado"

**53. A partir del código anterior, haz que aparezca al finalizar el programa por pantalla el total las sumas y el número de repeticiones. Con While**

5	El resultado de la suma es: 11
6	Deseas repetir la operación s/n:
s	
7	El resultado de la suma es: 13
6	Deseas repetir la operación s/n:
s	
5	El resultado de la suma es: 10
5	Deseas repetir la operación s/n:
n	Mensaje: "Resumen:"

**la suma total es: 34 y el número de repeticiones es: 3**

**54. Modifica el programa anterior y haz que se repita el ciclo automáticamente hasta que el total de todas las sumas sea superior a 50, será entonces cuando el programa finalice. No hará falta preguntar si deseas repetir la operación. En cada operación aparece por pantalla la suma de la operación y su acumulado. Para aquellos de vosotros que os fijáis en los detalles, controlar que el mensaje del acumulado es singular o plural.. 😊 . Con While**

5	El resultado de la suma es: 11
6	El total acumulado es: 11 y llevas 1 operación realizada
7	El resultado de la suma es: 13
6	El total acumulado es: 24 y llevas 2 operaciones realizadas
5	El resultado de la suma es: 10
5	El total acumulado es: 34 y llevas 3 operaciones realizadas
1	El resultado de la suma es: 3

2	El total acumulado es: 37 y llevas 4 operaciones realizadas
9	El resultado de la suma es: 17
8	El total acumulado es: 54 y llevas 5 operaciones realizadas Mensaje "Fin del programa"

**55. Última vez que reutilizamos el mismo código.. lo prometo 😊 . A partir del programa anterior haz que sea todo exactamente igual pero teniendo en cuenta que el programa se repita siempre y cuando la suma acumulada sea superior a 50 o la suma acumulada sea par. Con While**

5	El resultado de la suma es: 11
6	El total acumulado es: 11 y llevas 1 operación realizada
7	El resultado de la suma es: 13
6	El total acumulado es: 24 y llevas 2 operaciones realizadas
5	El resultado de la suma es: 10
5	El total acumulado es: 34 y llevas 3 operaciones realizadas
1	El resultado de la suma es: 3
2	El total acumulado es: 37 y llevas 4 operaciones realizadas
9	El resultado de la suma es: 17
8	El total acumulado es: 54 y llevas 5 operaciones realizadas
1	El resultado de la suma es: 3
2	El total acumulado es: 57 y llevas 6 operaciones realizadas Mensaje "Fin del programa"

**56. Realiza un programa que gestione un establecimiento de venta de bocadillos. Un pedido se compone de: **bocadillo, acompañamiento y bebida**. Un cliente puede pedir más de un pedido. El dependiente a partir del menú (ver imagen), se encarga de introducir los datos. El menú solo se visualiza una vez al ejecutar el programa. El programa debe preguntar al dependiente tras la realización de un pedido, si quiere gestionar otro.**

**El establecimiento contempla los siguientes descuentos:**

**Si el total a pagar es entre 20 y 30 euros, se aplica un descuento del 5%**

**Si el total a pagar es superior a 30 euros, se aplica un descuento del 15%**

**Una vez se finaliza la introducción de todos los pedidos de un cliente, debe aparecer por pantalla:**

- **El número de pedidos realizados**
- **Total a pagar.**
- **Total con IVA (10%)**
- **Total con el descuento aplicado.**

#### MENÚ

1. Bocadillo de calamares- 9 €"
2. Bocadillo de chistorra - 4.5 €
3. Bikini de jamón - 2.5 €"

#### ACOMPAÑAMIENTO

1. Patatas finas - 1.5 €

```

2. Patatas gruesas - 1.75 €
3. Patatas rústicas - 2 €

```

BEBIDAS

```

1. Coca cola - 2 €"
2. Acuario - 1.5 €
3. Agua - 1 €"

```

Ejemplo de 2 pedidos:

```

1
2
1
s
2
2
n

```

RESUMEN

```

Número de pedidos: 2
Total a pagar: 20.5
Total con iva: 22.55
Precio total con descuento del 5%: 21.4

```

Ejemplo de 3 pedidos:

```

2
1
1
s
1
1
1
s
3
2
1
s
3
3
1
n

```

RESUMEN

```

Número de pedidos: 4
Total a pagar: 33.25
Total con iva: 36.575
Precio total con descuento del 15%: 31.08

```

**56b.Opcional.** Haz alguna o todas las mejoras en el programa anterior que a continuación se indican:

- Cuando se pregunta "si desea realizar otro pedido", el encargado puede introducir s ó n en mayúscula o minúscula.
- Si el encargado introduce otro valor distinto a S o N, el programa debe repetir la pregunta e informar de que ha introducido un valor equivocado.
- El lugar de almacenar los precios en variables, utiliza una biblioteca (busca información) e investiga como moverte por los índices.
- Un pedido puede estar formado por 3, 2 o 1 componentes. Ej. Un usuario puede pedir únicamente una bebida.

**Ejercicios con Random.** Empezaremos con ejercicios sencillos para entender el funcionamiento de esta librería y combinaremos con bucles.

<b>57. Realiza un programa que permita adivinar un número comprendido entre 1 y 5. El programa debe controlar si el usuario introduce un número no comprendido entre 1 y 5</b>	
	Número acertado Número no acertado
<b>58. Modifica el programa anterior para que tengas 3 intentos. Utiliza while</b>	
<b>59. Diseña un programa que “piense” un numero aleatorio entre 0 y 1000 para que nos pida que intentemos adivinarlo. En cada intento, el programa nos dirá si el numero introducido es mayor o menor del correcto. No utilices break para salir del bucle. Cuando se acierte el número debe mostrarse por pantalla un mensaje y el número de intentos.</b>	
	Introduce un valor comprendido entre 1 y 1000: 500 El número que has introducido es menor Introduce un valor comprendido entre 1 y 1000: 600 El número que has introducido es mayor Introduce un valor comprendido entre 1 y 1000: 550 El número que has introducido es menor Introduce un valor comprendido entre 1 y 1000: 575 El número que has introducido es mayor Introduce un valor comprendido entre 1 y 1000: 560 El número que has introducido es mayor Introduce un valor comprendido entre 1 y 1000: 555 El número que has introducido es menor Introduce un valor comprendido entre 1 y 1000: 557 Acertaste, has realizado 7 intentos
<b>60. Diseña un programa que al introducir un número, realice su tabla de multiplicar del 1 al 10. Utiliza únicamente el while</b>	
5	5 10 15 20 25 30 35 40 45 50
2	2 4 6 8 10 12 14 16 18 20

**61. A partir del código anterior, haz que el programa finalice si el valor de la tabla de multiplicar es superior o igual a 40.**

5	5 10 15 20 25 30 35 40 Fin de programa
---	--

**62. Realiza un programa que pida dos números por teclado y presente por pantalla qué números hay pares e impares en ese rango. Utiliza for. Contempla si primer valor es superior al segundo.**

4  12	los números pares son 4-6-8-10 los números impares son 5-7-9-11
20  1	los números pares son 2-4-6-8-10-12-14-16-18 los números impares son 1-3-5-7-9-11-13-15-17-19

**63. Realiza un programa que permita tirar 100 veces un dado y nos presente por pantalla el número de veces que se repite cada número.**

	Uno: 11 Dos: 10 Tres: 27 Cuatro: 18 Cinco: 17 Seis: 16
	Uno: 16 Dos: 19 Tres: 19 Cuatro: 12 Cinco: 20 Seis: 13

**64. Programa que pida continuamente números por teclado hasta que el usuario introduzca el valor -99. Será entonces cuando por pantalla aparecerán las siguientes estadísticas:**

- a) total de pares**
- b) total de impares**
- c) total de números positivos**
- d) total de números negativos**
- e) total de ceros**
- f) total de la suma de todos los números introducidos**

Introduce un número: 5  Introduce un número: 4  Introduce un número: 5  Introduce un número: 6  Introduce un número: 7  Introduce un número: 1  Introduce un número: -6  Introduce un número: -2  Introduce un número: -99	RESUMEN  El número de pares es 4 El número de impares es 4 El número de positivos es 6 El número de negativos es 2 El total es 20
--	---

**65. Programa que pida continuamente números por teclado hasta que el usuario introduzca el valor -99. Por pantalla debe aparecer cuál de todos los números introducidos es el mayo y cuál el menor.**

Introduce un número: 5  
Introduce un número: 4  
Introduce un número: 5  
Introduce un número: 6  
Introduce un número: 7  
Introduce un número: 1  
Introduce un número: -6  
Introduce un número: -2  
Introduce un número: -99

RESUMEN  
El número de pares es 4  
El número de impares es 4  
El número de positivos es 6  
El número de negativos es 2  
El total es 20

**66. Repite el ejercicio 63. En lugar de 'tirar' 100 veces un dado, modifica el programa para ver cómo se comporta el dado en lanzamientos producidos durante aprox 3 segundos.**

RESUMEN  
Tiempo: 3.0004172325134277  
Uno: 995637  
Dos: 994685  
Tres: 998149  
Cuatro: 995805  
Cinco: 997652  
Seis: 1000241

3.000000476837158  
Uno: 981988  
Dos: 981767  
Tres: 979951  
Cuatro: 981394  
Cinco: 979586  
Seis: 982981

**67. Realiza de nuevo el programa de Password (fase 2). El password debe tener las siguientes consideraciones:**

**Debe tener una longitud entre 6 y 8 caracteres.**

**Debe contener como mínimo:**

**2 números mayores o iguales que 1 y menor o igual que 5**

**2 letras minúsculas**

**1 letra mayúscula**

**2 símbolos (\*, \_, @, &, /, #)**

**1 número mayor o igual que 6 y menor o igual que 5**

41A\*a8#3

password incorrecto

2Aa\*a

password incorrecto

8BA\*t7#4

password incorrecto

1bB\_a7/4

password correcto

1bB\_a7

password incorrecto

2Aa\*a5#8a

password correcto

**66. Añade al ejercicio anterior la posibilidad de que el programa pregunte si deseas o no continuar introduciendo passwords. Ej. "¿Deseas introducir otro password s/n?**