

VJ1203 – Programación I

Práctica 2: Bucles

(2 sesiones)

¡IMPORTANTE!:

1. Los ejercicios marcados con OBL son de carácter obligatorio y debes incluirlos necesariamente en la entrega.
2. Para los nombres de fichero utiliza la nomenclatura `a1XXXXXXprYYejZZ.py`, donde `a1XXXXXX` es tu usuario de la UJI, `prYY` el número de la práctica y `ejZZ` el de ejercicio. Por ejemplo, el nombre del fichero del *ejercicio 5* de la *práctica 1* del alumno *a1999999* debería ser `a1999999pr01ej05.py`.

Ejercicios con el módulo *turtle* y bucles *for*

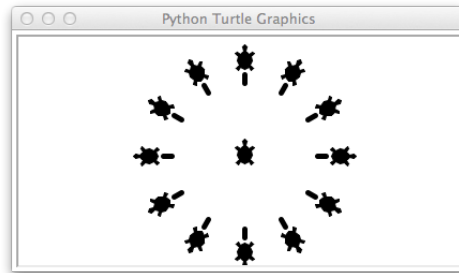
Ej. 1 — Visualizar una serie de puntos y calcular la distancia media al origen (OBL). Escribe un programa que pida un número entero n y después lea de teclado las coordenadas en el plano XY de una serie de n puntos. Usando el módulo *turtle* y un bucle *for* el programa deberá dibujar en la ventana gráfica cada uno de los puntos. Además, el programa deberá calcular la distancia media de los puntos al origen de coordenadas (0, 0) e imprimirla (con un `print`) por la salida estándar.

Ej. 2 — Visualizar una serie de puntos y un círculo centrado en el origen (OBL). *Ejercicio similar al anterior.* Escribe un programa que pida un número entero n y después lea de teclado las coordenadas de una serie de n puntos en el plano XY. Usando el módulo *turtle* y un bucle *for* el programa deberá dibujar en la ventana gráfica, por un lado, cada uno de los puntos y, por otro, el círculo de menor radio centrado en el origen de coordenadas (0, 0) que contenga todos los puntos de la serie. Además, el programa deberá imprimir por la salida estándar (con un `print`) el radio de dicho círculo.

Ej. 3 — Visualizar un cuadrado multicolor centrado en el origen (OBL). Escribe un programa que utilice el módulo *turtle* y un bucle *for* para dibujar en la ventana gráfica un cuadrado centrado en la posición (0, 0) y con cada uno de sus lados de un color. La longitud del lado del cuadrado se leerá de teclado en cada ejecución del programa.

Pista: Recuerda que el bucle *for* permite iterar sobre una lista de valores, que en este caso podrían ser colores. Puedes consultar la lista de posibles colores en la siguiente URL: <http://www.tcl.tk/man/tcl8.4/TkCmd/colors.htm>.

Ej. 4 — Visualizar la esfera de un reloj. Escribe un programa que utilice el módulo *turtle* y un bucle *for* para dibujar en la ventana gráfica la esfera de un reloj con marcas en forma de tortuga, tal y como se muestra en la figura a continuación. Pista: Para hacer las marcas puedes usar los métodos *shape* y *stamp*.

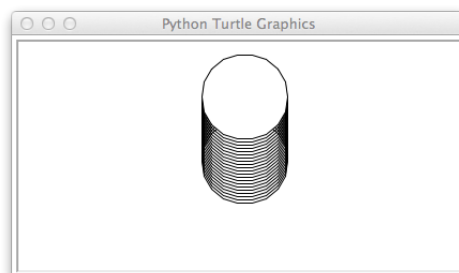


Ejercicios con el módulo *turtle* y bucles *while*

Ej. 5 — Visualizar la trayectoria de un “pirata borracho” (OBL). Un pirata borracho al caminar hace un giro aleatorio, avanza 50 pasos, vuelve a hacer un giro aleatorio, vuelve a avanzar 50 pasos, y así sucesivamente. Escribe un programa que utilice el módulo *turtle* y un bucle *while* para dibujar la trayectoria de un pirata borracho mientras esté dentro (o hasta que se salga fuera) de la ventana gráfica. Pista: Para generar un número aleatorio entre 1 y 360 usa la función *randint* del módulo *random*, de la siguiente forma: `giro = randint(1, 360)`.

Ej. 6 — Visualizar un cilindro (OBL). Un cilindro puede visualizarse dibujando y desplazando un círculo a lo largo de un eje con una cierta separación. Escribe un programa que utilice el módulo *turtle* y un bucle *while* para dibujar un cilindro con su base centrada en la posición (0, 0), tal y como se muestra en la figura a continuación. El radio del círculo y la altura del cilindro se leerán de teclado en cada ejecución del programa.

Pista: Para conseguir el efecto de la cara superior del cilindro puedes usar los métodos *fillcolor*, *begin_fill* y *end_fill*.



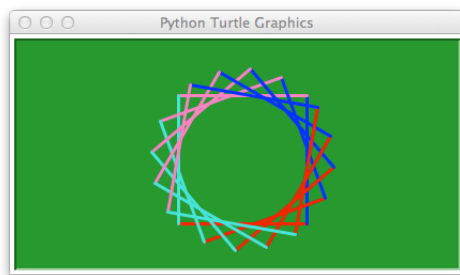
Ej. 7 — Visualizar la gráfica de la función seno. Escribe un programa que utilice el módulo *turtle* y un bucle *while* para visualizar el valor de la función seno para ángulos entre 0 y 360 grados según un incremento (en grados) dado, por ejemplo para 0, 10, 20, 30, etc si el incremento es de 10 grados. El incremento se leerá de teclado en cada ejecución del programa.

Pista: Ten en cuenta que la función `sin` del módulo `math` devuelve el seno de un ángulo dado en radianes.

Ejercicios con el módulo *turtle* y bucles *for* y *while*

Ej. 8 — Visualizar una serie de cuadrados concéntricos (OBL). Partiendo del programa del **Ej. 3**, escribe un programa que utilice el módulo `turtle` y bucles `for` y `while` para dibujar en la ventana gráfica una serie de cuadrados concéntricos centrados en la posición $(0, 0)$, ajustándose en la medida de lo posible al tamaño de la ventana. La longitud inicial del lado del cuadrado y la distancia entre los cuadrados se leerán de teclado en cada ejecución del programa.

Ej. 9 — Visualizar una serie de cuadrados rotados. Partiendo del programa del **Ej. 3**, escribe un programa que utilice el módulo `turtle` y bucles `for` y `while` para dibujar en la ventana gráfica una serie de cuadrados rotados con un cierto ángulo y centrados en la posición $(0, 0)$, tal y como se muestra en la figura a continuación. La longitud del lado del cuadrado y el ángulo de rotación se leerán de teclado en cada ejecución del programa.



Ejercicios con números y bucles

Ej. 10 — Adivinar un número (OBL). Escribe un programa para adivinar un número secreto que el usuario haya pensado. En primer lugar, el programa le pedirá al usuario que piense un número entre 1 y 100. A continuación el programa generará un número aleatorio, lo mostrará y preguntará al usuario si ha acertado. Si ha acertado el programa imprimirá un mensaje y terminará. Si no, el programa preguntará si el número secreto es mayor o menor que el número visualizado, y generará otro número aleatorio teniendo en cuenta esa información. Este proceso se repetirá hasta que el programa acierte el número.

Un ejemplo de ejecución del programa podría ser el siguiente:

```
Piensa un número del 1 al 100, ¡voy a intentar adivinarlo!
Pulsa intro cuando estés listo...
¿Es el 32 el número secreto? (s/n) no
Perdona, no te he entendido. ¿Es el 32 el número secreto? (s/n) n
¿Es el número secreto mayor o menor que el 32? (mayor/menor) menor
¿Es el 31 el número secreto? (s/n) n
¿Es el número secreto mayor o menor que el 31? (mayor/menor) menor
¿Es el 26 el número secreto? (s/n) n
¿Es el número secreto mayor o menor que el 26? (mayor/menor) menor
```

¿Es el 10 el número secreto? (s/n) n
 ¿Es el número secreto mayor o menor que el 10? (mayor/menor) mayor
 ¿Es el 15 el número secreto? (s/n) s
 Estoy de suerte, ¡HE ACERTADO!

Ej. 11 — Calcular el máximo común divisor de dos números (OBL). Escribe un programa que utilice un bucle *while* para calcular el máximo común divisor de dos números enteros positivos mediante el *algoritmo de Euclides*. La forma más simple del algoritmo de Euclides empieza con dos enteros positivos cualquiera y continua considerando el menor número y la resta del mayor menos el menor número. Por ejemplo, si los números son el 5 y el 7 los siguientes números a considerar serán el 5 y el 2. El proceso se repite hasta que los dos números son iguales, siendo ese número el máximo común divisor de los números originales.

Ej. 12 — Calcular el detalle de las cuotas de una hipoteca según el método alemán. Escribe un programa que calcule la cuantía de la amortización y los intereses correspondientes a los x primeros pagos mensuales de la cuota de una hipoteca según el método alemán. Además de los valores necesarios para el cálculo de la cuota de la hipoteca (que son: el capital h , el número de años n y el interés anual i), el valor de x se leerá de teclado en cada ejecución del programa.

Con el método alemán el valor de las cuotas está compuesto por una amortización constante del capital más los intereses (calculados sobre el capital pendiente). Para hallar la amortización dividiremos el capital inicial entre el número de cuotas (o sea, $h/(n \cdot 12)$). Para hallar los intereses, para el primer mes aplicaremos la tasa de interés mensual (o sea, $i/(100 \cdot 12)$) sobre el capital inicial del préstamo, y para los demás meses sobre el capital pendiente que vaya quedando. Por último, para hallar el capital pendiente restaremos la amortización al capital pendiente después del pago de la cuota anterior.

Por ejemplo, para amortizar una hipoteca de 150000€ en 15 años a un interés del 4,75 % anual, la amortización, los intereses y la cuota de las 6 primeras cuotas serán los siguientes:

| mes | amortización | intereses | cuota | capital pendiente |
|-----|--------------|-----------|----------|-------------------|
| 0 | — | — | — | 150000.00€ |
| 1 | 833.33€ | 593.75€ | 1427.08€ | 149166.67€ |
| 2 | 833.33€ | 590.45€ | 1423.78€ | 148333.33€ |
| 3 | 833.33€ | 587.15€ | 1420.49€ | 147500.00€ |
| 4 | 833.33€ | 583.85€ | 1417.19€ | 146666.67€ |
| 5 | 833.33€ | 580.56€ | 1413.89€ | 145833.33€ |
| 6 | 833.33€ | 577.26€ | 1410.59€ | 145000.00€ |