

Práctica 2

Práctica 2: Sentencias de control

UAH, Departamento de Automática, ATC-SOL
<http://atc1.aut.uah.es>

Semana de laboratorio 2

Objetivos:

- Desarrollar algoritmos simples
- Utilizar funciones
- Profundizar en el uso de sentencias de control de flujo

Comentarios iniciales

Los siguientes ejercicios están obtenidos de <https://github.com/zhiwehu/Python-programming-exercises>, http://www.ling.gu.se/~lager/python_exercises.html y “Programación estructurada en Pascal”, por José María Valls y David Camacho, Ed. Prentice-Hall.

Ejercicio 1

Defina una función `es_palindromo()` que devuelva `True` si la palabra pasada como argumento es palíndromo (se lee igual del derecho que del revés), o `False` en caso contrario.

Ejercicio 2

Defina una función `histograma()` que toma como argumento una lista de enteros e imprime su histograma por pantalla. por ejemplo, la llamada a `histogram([4, 9, 7])` debería imprimir lo siguiente por pantalla:

```
****
*****
*****
```

Ejercicio 3

Escribir una función que, dada una cadena pasada como argumento, obtenga el número de letras, vocales y palabras que contiene la frase. Utilice la ayuda e Internet en caso de que lo necesite.

Ejercicio 4

Se desea programar el juego de las tres en raya. ¿Qué estructura de datos es la más adecuada para almacenar la partida? Cree un programa que:

1. Obtenga por teclado el contenido del tablero (con fichas X y O)
2. Visualice por pantalla el contenido del tablero. Emplee para ello una función.
3. Indique si hay tres fechas del mismo tipo en raya.

Ejercicio 5

Cree una calculadora que muestre al usuario un menú con las operaciones básicas, lea por teclado dos valores, y muestre el resultado. Introduzca todas las operaciones básicas dentro de una función.

Ejercicio 6

Implemente el juego del buscaminas, para ello, siga los siguientes pasos.

1. Piense la estructura de datos en la que guardar el tablero. Tenga en cuenta que el mismo tiene un tamaño $N \times M$ de tamaño fijo, y cada casilla puede tener una mina, o no tenerla.
2. Implemente una función que inicialice el tablero introduciendo aleatoriamente las minas. Busque en Internet la forma de generar números aleatorios si es necesario.
3. Implemente una función que permita visualizar el tablero.
4. Implemente una función que, dada una posición (X, Y), devuelva el número de minas que hay alrededor de la posición. En caso de que exista una mina devuelva un -1 (o lance una excepción).
5. Implemente un menú que permita al jugador iteraccionar con el juego.

Ejercicio 7

Implemente el juego del ahorcado.