

# Programación I

## *Ejercicios de listas y strings (II)*

*Pablo Garaizar Sagarminaga  
Borja Sanz Urquijo  
M.<sup>a</sup> Luz Guenaga Gómez  
Jenny Fajardo Calderín*

Facultad de Ingeniería

# Menos de 5

- Prepara una lista con varias palabras y saca por pantalla otra lista que contenga solamente las palabras de menos de 5 letras.

```
python menosde5.py
```

```
["platano", "piña", "manzana", "luz"]
```

```
["piña", "luz"]
```

# Divisibles

- Prepara una lista de todos los números entre 1 y 100 que sean divisibles entre 2, 3 y 5 o 7.

```
python divisibles.py
```

```
[2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14,  
15, 16, 18, 20, 21, 22, 24, 25, 26,  
27, 28, 30, 32, 33, 34, 35, 36, 38,  
39, 40, 42, 44, 45, 46, 48, 49, 50,  
51, 52, 54, 55, 56, 57, 58, 60, 62,  
63, 64, 65, 66, 68, 69, 70, 72, 74,  
75, 76, 77, 78, 80, 81, 82, 84, 85,  
86, 87, 88, 90, 91, 92, 93, 94, 95,  
96, 98, 99, 100]
```

# Sin repetir

- Prepara una lista con 100 números aleatorios de 1 a 50 y saca por pantalla otra lista que contenga esos números pero sin repeticiones.

```
python sinrepetir.py
```

```
[18, 7, 21, 10, 2, 46, 20, 37, 34, 11,  
50, 36, 5, 6, 31, 49, 27, 29, 17, 41,  
25, 23, 44, 16, 13, 24, 15, 35, 22,  
30, 1, 28, 33, 32, 40, 39, 14, 47, 26,  
12, 48, 3, 42, 43, 19]
```

# Más larga

- Recibe un texto por teclado y muestra por pantalla la palabra más larga de ese texto.

`python maslarga.py`

**Texto: En un lugar de la Mancha**

**Palabra más larga: Mancha**

# Calificaciones

- Prepara una lista con los nombres de tus asignaturas para este curso y pide por teclado las calificaciones de cada una. Finalmente, muestra por pantalla la calificación media.

```
python calificaciones.py
```

```
Introduce tu calificación de Programación: 7
```

```
Introduce tu calificación de Álgebra: 5
```

```
Introduce tu calificación de Electrónica: 6
```

```
Introduce tu calificación de Cálculo: 5
```

```
Introduce tu calificación de Introducción a los Computadores: 9
```

```
Calificación media: 6.4
```

# Segundo más barato

- Prepara una lista con los precios de 10 productos y recorre la lista para encontrar el 2º más barato.

```
python barato.py
```

```
Precios: [123.50, 23.0, 12.99, 11.50, 67.40, 44.35, 111.0, 19.0,  
75.5, 99.99]
```

```
Segundo más barato: 12.99
```

# Sublista

- Prepara dos listas de números de diferente tamaño y muestra si una lista está dentro de otra lista.

```
python sublista.py
```

```
Lista a: [1, 5, 7, 12, 3, 56, 11]
```

```
Lista b: [12, 3, 56]
```

```
b es una sublista de a
```

```
Lista a: [1, 5, 7, 12, 3, 56, 11]
```

```
Lista b: [12, 56]
```

```
b NO es una sublista de a
```



# Morse

- Prepara una lista con todas las letras del alfabeto y otra lista con sus equivalentes en código Morse en el mismo orden. Recibe una frase por teclado y muéstrala en código Morse.

```
python morse.py
```

```
Frase: hola
```

```
Morse: .... --- .-.. -.- ....
```

A ● -	J ● - - -	S ● ● ●
B - ● ● ●	K - ● -	T -
C - ● - ●	L ● - ● ●	U ● ● -
D - ● ●	M - -	V ● ● ● -
E ●	N - ●	W ● - -
F ● ● - ●	O - - -	X - ● ● -
G - - ●	P ● - - ●	Y - ● - -
H ● ● ● ●	Q - - ● -	Z - - ● ●
I ● ●	R ● - ●	

# Ordenada

- Recibe números por teclado e introdúcelos en una lista. Cuando recibas un 0, deja de introducir números en la lista y muestra si la lista está ordenada o no.

```
python ordenada.py
```

```
Número: 7
```

```
Número: 11
```

```
Número: 2
```

```
Número: 0
```

```
[7, 11, 2] no está ordenada
```

# Insertar en orden

- Define una lista ordenada de números y pide un número por teclado. Calcula la posición en la que debería insertarse para que la lista siga ordenada e insértalo en la lista en esa posición.

```
python insertar.py
```

```
Lista: [1, 13, 23, 53, 98]
```

```
Número: 11
```

```
El número 11 hay que insertarlo en la  
posición 1
```

```
Lista: [1, 11, 13, 23, 53, 98]
```

# Dados

- Pide un número  $n$  por teclado y simula  $n$  tiradas de un dado de 6 caras. Almacena en una lista cuántas veces sale cada número y muestra sus porcentajes.

```
python dados.py
```

```
n: 50
```

```
Tiradas: [9, 11, 11, 12, 12, 7]
```

```
El 1 ha salido el 18% de las veces.
```

```
El 2 ha salido el 22% de las veces.
```

```
El 3 ha salido el 22% de las veces.
```

```
El 4 ha salido el 24% de las veces.
```

```
El 5 ha salido el 24% de las veces.
```

```
El 6 ha salido el 14% de las veces.
```

# Helados

- Define una lista con sabores de helado y prepara una lista con todas las permutaciones (el orden importa) con repetición de helados de 3 sabores.

```
python helados.py
```

```
Sabores: ["Chocolate", "Vainilla", "Fresa", "Nata", "Limón"]
```

```
Posibles helados: [
```

```
    ["Chocolate", "Chocolate", "Chocolate"],
```

```
    ["Chocolate", "Chocolate", "Vainilla"],
```

```
    ["Chocolate", "Chocolate", "Fresa"],
```

```
    ...
```

```
]
```

# Helados diferentes

- Define una lista con sabores de helado y prepara una lista con todas las permutaciones (el orden importa) sin repetición de helados de 3 sabores.

```
python heladosdiferentes.py
```

```
Sabores: ["Chocolate", "Vainilla", "Fresa", "Nata", "Limón"]
```

```
Posibles helados: [
```

```
    ["Chocolate", "Vainilla", "Fresa"],
```

```
    ["Chocolate", "Vainilla", "Nata"],
```

```
    ["Chocolate", "Vainilla", "Limón"],
```

```
    ...
```

```
]
```

# Sin usar

- Prepara una lista con varias palabras y saca por pantalla otra lista que contenga solamente las letras del alfabeto que no hayan sido utilizadas en esas palabras.

```
python sinusar.py
```

```
["platano", "piña", "manzana", "luz"]  
["b", "c", "d", "e", "f", "g", "h",  
"j", "k", "l", "q", "r", "s", "v",  
"w", "x", "y"]
```

# Ahorcado

- Define una palabra secreta en el código de tu programa y pide por teclado letras hasta completar la palabra secreta. Muestra en todo momento las letras correctas y las letras incorrectas, así como el número de intentos y la palabra que está siendo adivinada con – en las letras aún no encontradas.

```
python ahorcado.py
```

```
Letra: a
```

```
a sí está en la palabra
```

```
Correctas: ["a"]
```

```
Incorrectas: []
```

```
Intentos: 1
```

```
Palabra: --a-a--
```

```
...
```



# Fajardo S.A.

- El departamento de Recursos Humanos de una empresa cuenta con un listado de todos los trabajadores de la empresa. De cada trabajador se conoce su Nombre, Departamento, Sexo, Salario:

Nombre	Departamento	Sexo	Salario
Ana	Dept1	F	1500
Juan	Dept1	M	1500
Mikel	Dept2	M	1300
Iñigo	Dept3	M	1700
Laura	Dept3	F	1650
Idoia	Dept4	F	1700
Markel	Dept5	M	1500

# Fajardo S.A.

- De la información anterior necesitamos obtener diferentes reportes:
  - Dado un Departamento se necesita conocer el salario promedio del sexo “Femenino”.
  - Conocer todos los trabajadores que tienen un salario superior a 1500€.
  - Obtener un listado con todos los trabajadores de sexo “Masculino”.
  - Dado el nombre de un trabajador conocer el sexo, salario y departamento al que pertenece.