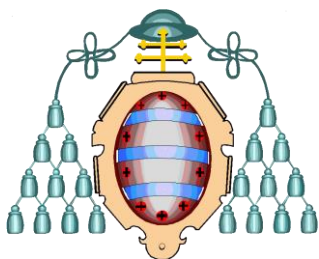


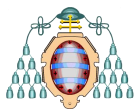
FUNDAMENTOS DE INFORMÁTICA

INTRODUCCIÓN A LA PROGRAMACIÓN

Funciones I



Departamento de Informática
Universidad de Oviedo



Ejercicio 1.- Definir dos funciones:

1. La función `lee_entero()` que retorna un número entero leído por teclado.
2. Una función que retorne el mayor de dos números enteros dados. A esta función la denotaremos como `mayor`.

Utilizar las funciones definidas previamente para escribir un programa que solicite dos números enteros y muestre en pantalla el mayor de ellos.

Ejercicio 2.- Mediante el uso de *docstrings* documenta las funciones del ejercicio anterior.

Nota. Además de los comentarios que se pueden incluir en cualquier parte del programa, las funciones se pueden documentar para especificar, textualmente, *qué necesitan* y *qué retornan*. Para ello se utilizan los *docstrings* que son cadenas de texto que se incluyen en el código fuente, al igual que un comentario.

En Python los *docstrings* comienzan y terminan con tres comillas dobles: `"""` y, lo habitual, es que se ubiquen tras la cabecera de la función. Ejemplo:

```
def producto (a, b):  
    """Dados dos números, a y b, retorna  
    el producto de ambos"""  
    return a * b
```

La función predefinida de ayuda (`help`) permite recuperar esta documentación pasando el nombre de la función. Así, la expresión `help(producto)` mostrará el siguiente resultado:

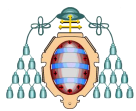
```
Help on function producto in module __main__:  
  
producto(a, b)  
    Dados dos números, a y b, retorna el producto de ambos
```

Ejercicio 3.- Utilizando las funciones del ejercicio 1 ¿podrías calcular el mayor de 3 números dados? Escribe un programa que así lo permita.

Ejercicio 4.- Define una función que retorne cierto si un año dado es bisiesto y falso en caso contrario. A continuación, escribe un programa que solicite por teclado un año (entero positivo) y muestre en pantalla si éste es o no bisiesto, haciendo uso de la función ya definida.

Se supondrá que el año se introduce correctamente y para describir la función se puede utilizar la definición de año bisiesto dada en la Wikipedia.

Ejemplo de resultado (para el año en curso) -> El año 2014 no es bisiesto.



Ejercicio 5.- Escribe una función que dada la nota numérica (con un decimal) de un alumno proporcione la calificación: *suspenso*, *aprobado*, *notable*, *sobresaliente* o *matrícula de honor*. Dónde:

$$\text{calificacion}(x) = \begin{cases} \text{suspenso} & \text{si } x \in [0, 5) \\ \text{aprobado} & \text{si } x \in [5, 7) \\ \text{notable} & \text{si } x \in [7, 9) \\ \text{sobresaliente} & \text{si } x \in [9, 10) \\ \text{matrícula de honor} & \text{si } x = 10 \end{cases}$$

Documenta la función realizada y escribe un programa que solicite por teclado la nota numérica del alumno y muestre en pantalla su calificación. Se supondrá que la nota se introduce correctamente.

Ejercicio 6.- Define una función que proporcione el *estado nutricional* de una persona. Esto es; la clasificación de su *índice de masa corporal* (IMC), según la tabla proporcionada. La función recibirá el peso (en kilos) y la talla de una persona (en metros), datos a partir de los cuales se puede calcular el IMC cómo:

$$\text{IMC} = \frac{\text{peso}}{\text{altura}^2}$$

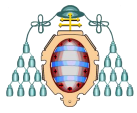
Clasificación	IMC (Kg/m ²)
<i>Bajo peso</i>	< 18.50
<i>Normal</i>	[18.50, 25.00)
<i>Sobrepeso</i>	≥ 25.00
<i>Obesidad</i>	≥ 30.00

Documenta la función realizada y escribe un programa que solicite por teclado el peso y la talla de una persona (ambos con dos decimales) y muestre en pantalla su estado nutricional. Se supondrá que los datos se introducen correctamente.

Ejercicio 7.- Define una función para obtener la distancia al origen de un punto de abscisa y ordenada dadas. Una vez definida la función, escribe un programa que obtenga por teclado las coordenadas cartesianas de un punto del plano y muestre en pantalla la distancia de éste al origen de coordenadas.

En la definición de la función, deberás importar las funciones necesarias desde el módulo `math`. Esta librería proporciona, entre otras, una función para obtener la raíz cuadrada de un número dado (`sqrt(x)`) y otra que calcula la potencia `y` de un número dado `x` (`pow(x, y)`).

Nota. Puedes consultar cómo utilizar las funciones de la librería `math` en la documentación en línea de Python: <https://docs.python.org/3/library/math.html>



Ejercicio 8.- Utilizando la función del ejercicio previo, define una función que retorne la longitud de una circunferencia cuyo centro es el origen de coordenadas y que pasa por un punto del plano de abscisa y ordenada dadas. Para determinar la longitud, se utilizará el valor de π importado desde el módulo `math` (`pi`).

Realiza un programa que solicite por teclado las coordenadas cartesianas de un punto del plano por el que pasa una circunferencia con centro en el origen de coordenadas y, utilizando la función realizada, muestra en pantalla su longitud.