

Funciones I

Informática - Grupo A1

Grados en Ing. Mecánica e Ing. en Tecnologías Industriales - 1^{er} curso

Alma Mallo - alma.mallo@udc.es

Material necesario

- Tema 7 de teoría: Introducción a Python 3 - Parte 1
- Diapositivas 43 a 48:
 - **Funciones**

Ejercicio 1

- Realizar un programa en Python que cuente y muestre por pantalla todos los números oblongos que se encuentren entre los números $n1$ y $n2$ (ambos inclusive).
 - Un número es oblongo si es igual al producto de dos números enteros consecutivos. Por ejemplo, son números oblongos el 2, 6 y 12, ya que $2 = 1 \times 2$, $6 = 2 \times 3$, $12 = 3 \times 4$.
 - Los números $n1$ y $n2$ deben pedirse al usuario teniendo en cuenta que $n1$ debe ser estrictamente mayor que 1 y $n2$ debe ser estrictamente mayor que $n1$.
- Funciones a implementar:
 - Petición de un número: se le pasa como argumento el límite inferior que debe cumplir y devuelve el número.
 - Función que comprueba si un número es oblongo: se le pasa un número y devuelve True si el número es oblongo y False si no lo es.

Ejercicio 1 (ejemplos)

=====

```
Introduzca un numero mayor que 1: 30
Introduzca un numero mayor que 30: 15
Error! El numero debe ser mayor que 30.
Introduzca un numero mayor que 30: 100
```

```
Los numeros oblongos entre 30 y 100 son: 30 42 56 72 90
En total son 5
```

=====

=====

```
Introduzca un numero mayor que 1: 1
Error! El numero debe ser mayor que 1.
Introduzca un numero mayor que 1: 111
Introduzca un numero mayor que 111: 120
```

```
Los numeros oblongos entre 111 y 120 son:
No se ha encontrado ningún numero oblongo
```

=====

Ejercicio 2

- Realizar un programa que pida al usuario por teclado dos años, año inicio y año fin, y muestre todos los años Jacobeos que existen entre ellos
 - Los años no pueden estar fuera del rango de años entre 1 y 9999
 - El año de fin debe ser mayor que el año de inicio.
- Se deberán realizar, como mínimo, las siguientes funciones:
 - pedir_anho: pide un año al usuario entre un límite mínimo y máximo.
 - es_jacobeo: indica si un año es jacobeo o no.

Ejercicio 2

Se denomina Año Santo Jacobeo al año en que el 25 de julio (festividad de Santiago Apóstol) cae en domingo. Para resolver el programa se usará la congruencia de Zeller, un algoritmo ideado por Julius Christian Johannes Zeller que permite obtener, a partir de una fecha, el día de la semana que le corresponde. Para el calendario gregoriano, y dados tres números *dia*, *mes* y *anho* que representan una fecha (día, mes y año), la congruencia de Zeller se expresa:

$$\text{dia_semana} = \left(\text{dia} + y + \frac{y}{4} - \frac{y}{100} + \frac{y}{400} + \frac{(31 * m)}{12} \right) \text{mod } 7$$

donde

$$m = \text{mes} + (12 * a) - 2$$

$$a = (14 - \text{mes}) / 12$$

$$y = \text{anho} - a$$

El resultado *dia_semana* tomará valores entre 0 y 6 (0 significa Domingo, 1 significa Lunes, etc.).

Importante: todas las divisiones son enteras

Ejercicio 2 (ejemplos)

```
=====
Introduzca año de comienzo (entre 1 y 9999): -2
Error de rango! Introduzca año de comienzo (entre 1 y 9999): 2030
Introduzca año de fin (entre 2030 y 9999): 1990

Los año xacobeos entre 1990 y 2030 son:
1993 1999 2004 2010 2021 2027
=====

=====
Introduzca año de comienzo (entre 1 y 9999): 1960
Introduzca año de fin (entre 1960 y 9999): 1950
Error de rango! Introduzca año de fin (entre 1960 y 9999): 2022

Los año xacobeos entre 1960 y 2022 son:
1965 1971 1976 1982 1993 1999 2004 2010 2021
=====
```

Ejercicio 3

- Desarrollar un programa en Python que calcule la suma de dos fracciones, cuyos denominadores y numeradores serán pedidos al usuario. Una vez sumadas las dos fracciones se simplificará la fracción resultante, que ha de ser mostrada por pantalla tal y como se indica en los ejemplos de la página siguiente.
- NOTAS IMPORTANTES:
 - Si el numerador o el denominador de alguna de las fracciones es negativo se pedirá de nuevo al usuario hasta que introduzca uno válido.
 - Se mostrará un aviso al usuario indicando si se ha podido simplificar el resultado. Además, si el denominador del resultado es igual a 1 no se mostrará como una fracción.

Ejercicio 3

- Se desarrollará una función por cada una de las siguientes operaciones:
 - Pedir por teclado al usuario el numerador y denominador de cada fracción.
 - Sumar dos fracciones.
 - Calcular el máximo común divisor de dos números enteros para simplificar las fracciones:
 - Para calcular el mcd se usará el método de Nicómaco. Este método se basa en restar el número menor al número mayor y reemplazar el número mayor por el valor de dicha resta. La operación se repite sucesivamente hasta que ambos números son iguales. Ese número final es el MCD de los dos números originales.
 - Ejemplo: dados los números 42 y 56, los pasos serían los siguientes:
 - Paso 1: $56-42=14$
 - Paso 2: $42-14=28$
 - Paso 3: $28-14=14$
 - Paso 4: 14 y 14 son iguales con lo que se detiene el proceso y el resultado final es 14
 - Simplificar una fracción.

Ejercicio 3 (ejemplos)

```
=====
Introduzca el numerador de la primera fracción: -5
El dato debe ser mayor que cero. Vuelva a intentarlo: 3
Introduzca el denominador de la primera fracción: 4
Introduzca el numerador de la segunda fracción: 0
El dato debe ser mayor que cero. Vuelva a intentarlo: 5
Introduzca el denominador de la segunda fracción: 8
```

Aviso: la suma se ha simplificado.

El resultado de sumar $3/4 + 5/8$ es $11/8$

```
=====
Introduzca el numerador de la primera fracción: 4
Introduzca el denominador de la primera fracción: 5
Introduzca el numerador de la segunda fracción: 12
Introduzca el denominador de la segunda fracción: 10
```

Aviso: la suma se ha simplificado.

El resultado de sumar $4/5 + 12/10$ es 2

```
=====
Introduzca el numerador de la primera fracción: 5
Introduzca el denominador de la primera fracción: 7
Introduzca el numerador de la segunda fracción: 8
Introduzca el denominador de la segunda fracción: 9
```

Aviso: la suma NO se ha podido simplificar.

El resultado de sumar $5/7 + 8/9$ es $101/63$

```
=====
```

Trabajo autónomo

- Realizar ejercicios propuestos al resto de grupos.
- Preparación clase siguiente:
 - Tema 7: Introducción a Python 3
 - Diapositivas 43 a 48:
 - **Funciones**