

Fundamentos de la Programación. Grado en Física - Curso 2017/2018

Examen extraordinario de teoría. Convocatoria de Julio (03/07/2018)

Instrucciones

1. Tienes **2 horas y 30 minutos** para realizar el examen. Lee tranquilamente el examen y decide por dónde empezar.
2. No puedes utilizar apuntes ni programas ya escritos por ti o por otros para hacer el examen. Tampoco puedes comunicarte con nadie (excepto con el profesor) durante el examen. Puedes consultar al profesor por instrucciones concretas de Python que no recuerdes.
3. **Importante:** Pon tu DNI, nombre y apellidos al principio de cada ejercicio que entregues.
4. Finalmente debes entregar tu examen con todos los ejercicios que hayas realizado en un archivo comprimido en `zip` o en `tar.gz` cuyo nombre debe ser tu DNI (`NUMERO_DNI.zip` o `NUMERO_DNI.tar.gz`).
5. Para entregar dicho archivo comprimido indícaselo al profesor.

Ejercicios

1. (ej1.py) (**2 puntos**) Deseamos implementar una función que elimine los elementos de una lista de enteros que tienen valor par. Se deben implementar dos funciones para dicha tarea:
 - `borraPares(lista)`: esta función no debe devolver nada, pero los valores pares de `lista` deben ser eliminados.
 - `eliminaPares(lista)`: esta función debe devolver una nueva lista con los valores pares eliminados (`lista` debe quedar inalterada).

Ejemplo: Si `lista=[1, 0, 2, 16, 11, 15, -2]`, las siguientes llamadas deben producir la salida indicada:

```
>>> print(eliminaPares(lista))
[1, 11, 15]
>>> print(lista)
[1, 0, 2, 16, 11, 15, -2]
>>> borraPares(lista)
>>> print(lista)
[1, 11, 15]
```

2. (ej2.py) (**2 puntos**) Como es bien sabido, el desarrollo en serie de potencias de la función seno es

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!}$$

Diseña una función que calcule el $\sin x$ utilizando únicamente sumas, restas, multiplicaciones y divisiones. El cálculo debe finalizar cuando se asegure un error menor que 10^{-12} .

NOTA: El programa debe calcular de forma eficiente tanto las potencias como los factoriales involucrados en cada término, de modo que no se vuelvan a calcular los factores calculados previamente. Para ello utiliza las propiedades $x^{2n+1} = x \cdot x^{2n}$ y $(2n+1)! = (2n+1)(2n)(2n-1)!$.

3. (ej3.py) (**2 puntos**) Implementa un programa al que se le pasan dos cadenas por línea de comandos, la primera cadena corresponde a un archivo de texto y la segunda a la cadena que hay que buscar en dicho archivo. El programa mostrará por pantalla todas las líneas en las que aparece la cadena, como palabra completa (no como subcadena), precedidas por el número de línea.

Ejemplo: Si el contenido del archivo `quijote.txt` es

```
En un lugar de la Mancha
de cuyo nombre no quiero acordarme,
no ha mucho tiempo que vivía un hidalgo
de los de lanza en astillero,
adarga antigua, rocín flaco y galgo ...
```

y ejecutamos la instrucción

```
$ python3 ej3.py quijote.txt de
```

La salida sería

```
1: En un lugar de la Mancha
2: de cuyo nombre no quiero acordarme,
4: de los de lanza en astillero,
```

Si hay algún error se debe emitir el mensaje **ERROR**.

4. (ej4.py) (**2 puntos**) Modifica el programa del ejercicio anterior para que la salida se guarde en un archivo con el mismo nombre con el sufijo `_cad` antes del primer punto en lugar de por pantalla (si no hubiera ningún punto se añadiría al final). El sufijo `cad` debe ser la cadena buscada en el archivo.

Ejemplo: La salida del ejemplo del ejercicio anterior se guardaría en el archivo `quijote_de.txt` y no se mostraría nada por pantalla (salvo el error en su caso).

5. (ej5.py) (**2 puntos**) Traduce a Python el siguiente programa en Fortran que define una función para calcular el máximo común divisor (MCD) de dos números y en el programa principal pide dos números al usuario y devuelve el MCD de dichos números.

```
PROGRAM EJEMPLO_8_3
IMPLICIT NONE
INTEGER :: I,J,Result
INTEGER :: MCD
EXTERNAL MCD
PRINT *, ' INTRODUCE DOS NUMEROS ENTEROS: '
READ(*,*) I,J
RESULT = MCD(I,J)
PRINT *, ' EL MAX COMUN DIV DE ',I,' Y ',J,' ES ',RESULT
END PROGRAM EJEMPLO_8_3
!
INTEGER FUNCTION MCD(A,B)
IMPLICIT NONE
INTEGER , INTENT(IN) :: A,B
INTEGER :: Temp
IF (A < B) THEN
    Temp=A
ELSE
    Temp=B
ENDIF
DO WHILE ((MOD(A,Temp) /= 0) .OR. (MOD(B,Temp) /=0))
    Temp=Temp-1
END DO
MCD=Temp
END FUNCTION MCD
```