

Fundamentos de la Programación - Grado en Física - Curso 2018/2019

Examen parcial grupo 1 (04/12/2018)

Instrucciones

- Tienes **2 horas** para realizar el examen. Lee tranquilamente el examen y decide por dónde empezar.
- No puedes comunicarte con nadie (excepto con el profesor) durante el examen.
- **Importante:** Pon tu DNI, nombre y apellidos al principio de cada ejercicio que entregues.
- Debes comprimir tu examen con todos los ejercicios que hayas realizado en un archivo **zip** o en **tar.gz** cuyo nombre debe ser tu DNI (**NUMERO_DNI.zip** o **NUMERO_DNI.tar.gz**).
- Debes entregar dicho archivo comprimido a través de la aplicación de UACloud → Evaluación → Entrega de Prácticas. Dicha aplicación se cerrará automáticamente a la hora fijada para la finalización del examen, de modo que debes tener cuidado en no retrasarte.

Ejercicios

1. (ej1.py) (**4 puntos**) Escribe una función en Python, **repeticiones(c, cad)**, que devuelva cuántas veces se repite un carácter **c** en una cadena **cad**. A continuación utiliza esa función para imprimir por pantalla el número de repeticiones de todos los caracteres de una cadena en otra cadena (ambas solicitadas por el programa) según el formato del siguiente ejemplo. Observa que se distinguen mayúsculas y minúsculas, que si una letra se repite 0 veces no hay que imprimirla y que sólo hay que tener en cuenta una vez las letras de la primera cadena.

Ejemplo:

Entrada	Salida
cadena Encuentra cuantas veces aparece	c: 4 a: 5 e: 5 n: 3

2. (ej2.py) (**3 puntos**) Puedes calcular recursivamente los números combinatorios sabiendo que, para $n \geq m$,

$$\binom{n}{m} = \binom{n-1}{m} + \binom{n-1}{m-1}$$

y que

$$\binom{n}{n} = \binom{n}{0} = 1$$

Diseña una función recursiva **combinaciones(n)** que devuelva, en forma de lista, $\binom{n}{m}$ para m entre 0 y n . Después diseña un programa que, a partir de un valor n leído del teclado, muestre el resultado por pantalla en el formato del siguiente ejemplo (para $n = 5$):

Las combinaciones de 5 elementos tomados de m en m son:

```
m = 0: 1
m = 1: 5
m = 2: 10
m = 3: 10
m = 4: 5
m = 5: 1
```

3. (ej3.py) (**3 puntos**) Queremos leer un conjunto de datos experimentales de un archivo. Los datos están organizados del siguiente modo:

- La primera línea contiene una descripción textual de las variables que se han medido en el experimento, por ejemplo:

```
hora_del_dia temperatura velocidad_del_viento presion_atmosferica humedad_relativa
```

- La segunda línea indica el número de medidas realizadas de cada una de las variables.

- El resto de líneas continen una serie de números reales, separados por espacios en blanco, que representan los valores obtenidos para las variables de la primera línea.

Un ejemplo de archivo podría ser:

```
hora_del_dia temperatura velocidad_del_viento presion_atmosferica humedad_relativa
5
10 15.1 23.2 1.01 73.2
11 15.5 23.1 1.02 81.6
12 17.2 26.2 0.99 97.3
13 17.5 25.2 0.98 100
14 18.3 20.1 1.00 99.3
```

Debes volcar estos datos en otro archivo ordenados por filas en lugar de por columnas. Es decir, el archivo de salida debe contener

```
hora_del_dia: 10 11 12 13 14
temperatura: 15.1 15.5 17.2 17.5 18.3
velocidad_del_viento: 23.2 23.1 26.2 25.2 20.1
presion_atmosferica: 1.01 1.02 0.99 0.98 1.00
humedad_relativa: 73.2 81.6 97.3 100 99.3
```

Observa que eliminamos el número de medidas y que añadimos ':' tras la descripción de la variable.

Los nombres de los archivos de entrada y salida se pasaran por línea de comandos. Si el programa se ejecuta sin alguno de los dos nombres se debe dar el aviso correspondiente y si el archivo de entrada no está disponible se debe emitir el error correspondiente sin realizar ninguna otra acción (ni siquiera creando el archivo de salida).