

## Programación I – 2020

### Trabajo práctico 1

#### Ejercicio 1

Los alumnos de un curso se han dividido en dos grupos A y B de acuerdo al sexo y el nombre. El grupo A esta formado por las mujeres con un nombre anterior a la M y los hombres con un nombre posterior a la N y el grupo B por el resto. Escribir un programa que pregunte al usuario su nombre y sexo, y muestre por pantalla el grupo que le corresponde.

#### Ejercicio 2

La pizzería Bella Napoli ofrece pizzas vegetarianas y no vegetarianas a sus clientes. Los ingredientes para cada tipo de pizza aparecen a continuación.

- Ingredientes vegetarianos: Pimiento y tofu.
- Ingredientes no vegetarianos: Peperoni, Jamón y Salmón.

Escribir un programa que pregunte al usuario si quiere una pizza vegetariana o no, y en función de su respuesta le muestre un menú con los ingredientes disponibles para que elija. Solo se puede elegir un ingrediente además de la mozzarella y el tomate que están en todas la pizzas. Al final se debe mostrar por pantalla si la pizza elegida es vegetariana o no y todos los ingredientes que lleva.

#### Ejercicio 3

Escribir un programa que pida al usuario un número entero positivo y muestre por pantalla todos los números impares desde 1 hasta ese número separados por comas.

#### Ejercicio 4

Escribir un programa que pida al usuario un número entero y muestre por pantalla un triángulo rectángulo como el de más abajo, de altura el número introducido.

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

### Ejercicio 5

Escribir un programa que pida al usuario un número entero y muestre por pantalla un triángulo rectángulo como el de más abajo.

```
1
3 1
5 3 1
7 5 3 1
9 7 5 3 1
```

### Ejercicio 6

Escribir un programa que pida al usuario un número entero y muestre por pantalla si es un número primo o no.

### Ejercicio 7

Escribir un programa en el que se pregunte al usuario por una frase y una letra, y muestre por pantalla el número de veces que aparece la letra en la frase.

### Ejercicio 8

Escribir un programa que almacene las asignaturas de un curso (por ejemplo Matemáticas, Física, Química, Historia y Lengua) en una lista, pregunte al usuario la nota que ha sacado en cada asignatura, y después las muestre por pantalla con el mensaje En <asignatura> has sacado <nota> donde <asignatura> es cada una des las asignaturas de la lista y <nota> cada una de las correspondientes notas introducidas por el usuario.

### Ejercicio 9

Escribir un programa que pregunte al usuario su nombre, edad, dirección y teléfono y lo guarde en un diccionario. Después debe mostrar por pantalla el mensaje <nombre> tiene <edad> años, vive en <dirección> y su número de teléfono es <teléfono>.

### Ejercicio 10

Escribir un programa que pregunte una fecha en formato dd/mm/aaaa y muestre por pantalla la misma fecha en formato dd de <mes> de aaaa donde <mes> es el nombre del mes.

### Ejercicio 11

Escribir un programa que almacene el diccionario con los créditos de las asignaturas de un curso {'Matemáticas': 6, 'Física': 4, 'Química': 5} y después muestre por pantalla los créditos de cada asignatura en el formato <asignatura> tiene <créditos> créditos, donde <asignatura> es cada una de las asignaturas del curso, y <créditos> son sus créditos. Al final debe mostrar también el número total de créditos del curso.

### Ejercicio 12

Generar números aleatorios del 1 al 10 y guardar el resultado en un diccionario. Al finalizar mostrar el resultado del diccionario ordenado de mayor a menor.

### Ejercicio 13

Generar 10 números aleatorios, mostrarlos por pantalla y mostrar el promedio.

### Ejercicio 14

Crear una variable con triple comidas que contenga un texto copiado de algún lugar de al menos 20 líneas y hacer un conteo de palabras, al finalizar el conteo mostrar las 20 palabras que más se repiten. Mostrar todas las palabras en orden alfabético y la cantidad que se repite.

### Ejercicio 15

Crear un programa que lea un archivo de texto que va a contener datos de una venta y mostrarlo por pantalla.

El formato será:

Nombre y apellido, monto de la venta, descripción, forma de pago (contado o tarjeta)

El archivo contendrá una línea por cada venta, y podrá contener múltiples ventas.

### Ejercicio 16

Modificar el ejercicio 15 y hacer que los datos se muestren con una etiqueta por delante, algo así como:

Nombre: xxxxxxxxxx, monto: xxxx.xx, descripción: xxxxxxxxxx, forma de pago: xxxxxxxx

### Ejercicio 17

Crear un programa que lea el archivo del ejercicio 15 y lo convierta a formato JSON con el siguiente formato:

```
{"nombre":"xxxxxxx", "monto":xxxx.xx, "descripcion":"xxxxxxx", "formapago":"xxxx"}
```

Debemos mostrarlo por pantalla y crear un archivo llamado ARCHIVO\_ORIGINAL.json donde guardaremos los objetos JSON convertidos

El nombre ARCHIVO\_ORIGINAL debe ser el que se definió para el archivo de datos.

## Ejercicio 18

Modificar el programa del ejercicio 17 agregando un control de datos vacíos.

Si por algún motivo alguno de los datos viene vacío o igual a "" debemos generar una excepción indicando que ocurrió un problema con la fuente de datos y notificar por pantalla.

La excepción debe ser una excepción propia.

## Ejercicio 19

Crear un programa similar al ejercicio 18 (incluyendo la funcionalidad del ejercicio 14) donde los datos del archivo de texto serán:

Nombre y apellido, número de tarjeta de crédito, código de verificación, tipo de tarjeta de crédito, monto de la venta, descripción de venta.

Los datos deben manejarse de la siguiente manera:

Crear una clase TarjetaCredito que almacene los datos de: Nombre y apellido, número de tarjeta de crédito, código de verificación, tipo de tarjeta de crédito.

Crear una clase Venta que almacene los datos de: monto de la venta, descripción de venta y un objeto de la clase TarjetaCredito.

El formato de los datos JSON deberá respetar el formato de Venta con los datos de monto y descripción y el objeto de la clase Venta.