



## Programación—Certamen 1 - Miércoles 12 de Abril de 2017

Nombre

Rol

Paralelo

2. [35 %] La constructora Pythonia tiene en su haber una gran cantidad de terrenos y su negocio es venderlos por lotes para que empresas y particulares construyan sus fábricas o viviendas. La manera actual de establecer los precios de esos lotes de terreno es preguntándole a un tasador. Debido a la gran cantidad de terrenos que la constructora está teniendo, se requiere un algoritmo para que establezca los precios puesto que el tasador ya no da abasto.

El precio por metro cuadrado es de 20 UF. Si el área del terreno es inferior a 100m<sup>2</sup> se aplica un descuento de 10 %, si el área está comprendida entre 100m<sup>2</sup> y 1000m<sup>2</sup>, se le aplica un incremento del 20 %. Finalmente, si es superior a 1000m<sup>2</sup> se suma un 50 %. Además los terrenos están en 3 sectores distintos.

Haga un diagrama de flujo que vaya preguntando por el área y sector de cada terreno a tasar. Cuando el área ingresada sea cero, el programa debe finalizar, pero antes debe imprimir por cada sector el total avaluado junto con el área y precio del terreno más caro en éste.

## Programación—Certamen 1 - Miércoles 12 de Abril de 2017

Nombre

Rol

Paralelo

3. [40 %] La comisión organizadora de la semana mechona creó un concurso on line, en el cual sus 3 alianzas deben participar. El concurso consta de una serie de turnos, en los cuales cada alianza debe ingresar una palabra. La palabra con mayor cantidad de vocales gana. Si dos o más alianzas igualan en el máximo de vocales, nadie gana esa ronda. El concurso termina cuando **una** alianza logre ganar  $x$  juegos (meta), la cual se debe definir al inicio del juego.

En base a lo anterior, la comisión le solicita a usted implementar lo siguiente:

- a) Escriba la función `ganador(c1, c2, c3, meta)` la cual recibe 4 parámetros, la cantidad de juegos ganados por la alianza 1, 2 y 3 y la meta a lograr en el juego. La función debe retornar el número de la alianza ganadora. En caso de no existir ganador, debe retornar el valor entero 0 (cero).

**Nota:** Tener en cuenta que no pueden existir empates en las alianzas al momento de alcanzar la meta.

```
>>> ganador(2, 5, 3, 5)
2
>>> ganador(1, 3, 3, 5)
0
```

- b) Escriba la función `contar(palabra)` que reciba un string `palabra`. La función debe retornar la cantidad de vocales existentes en la `palabra` recibida.

**Nota:** no puede utilizar el método `count` de procesamiento de texto.

```
>>> contar('paralelepipedo')
7
>>> contar('str')
0
```

- c) Desarrolle un programa que solicite la meta del Juego, y luego solicite las palabras de cada alianza por turno, hasta que exista una alianza que logre la meta. **Asuma que las palabras siempre serán ingresadas en minúsculas.** A continuación se presenta un ejemplo de cómo debiera lucir el programa.

```
Ingrese meta del Juego: 3
alianza 1: paralelepipedo
alianza 2: reuna
alianza 3: salida
alianza 1: tartamudo
alianza 2: mauricio
alianza 3: semana
alianza 1: murcielago
alianza 2: rescate
alianza 3: salvavida
alianza 1: temperatura
alianza 2: sala
alianza 3: certamen
La alianza ganadora es: 1
```