

## Funciones: Ejercicios

1. **Cargos de Estacionamiento:** Un estacionamiento cobra una tarifa mínima de \$2.00 por hasta tres horas y \$0.50 adicionales por hora por cada hora o fracción que exceda las tres horas. El cargo máximo para un período de 24 horas es de \$10.00. Suponga que ningún vehículo estaciona por más de 24 horas seguidas. Escriba un programa que calcule e imprima los cargos de estacionamiento de cada uno de los tres clientes que estacionaron sus autos en este estacionamiento ayer. Debe ingresar las horas de estacionamiento de cada cliente. Su programa debe imprimir los resultados en forma de tabla y calcular e imprimir el total de los recibos de ayer. El programa debe usar la función `calcularCargos` para determinar el cargo de cada cliente. Los resultados deben aparecer en el siguiente formato:

Vehículo	Horas	Cargo
1	1.5	2.00
2	4.0	2.50
3	24.0	10.00
TOTAL	29.5	14.50

2. **Redondeo de números:** Una aplicación de la función `floor` es redondear un valor al **entero más cercano**. La instrucción `y = floor(x + .5)`; redondea `x` al entero más cercano y asigna el resultado a `y`. Escriba un programa que lea varios números y redondee cada uno al entero más cercano. Para cada número procesado, imprime tanto el número original como el redondeado.
3. **Redondeo de números:** La función `floor` se puede usar para redondear un número a un **decimal específico**. Por ejemplo, la instrucción `y = floor(x * 10 + .5) / 10`; redondea `x` a la décima (la primera posición a la derecha del punto decimal).

La sentencia `y = floor(x * 100 + .5) / 100`; redondea `x` a las centésimas (la segunda posición a la derecha del punto decimal). Escriba un programa que defina funciones para redondear un número `x` de varias maneras:

- `redondeaaentero (número)`
- `redondearadecima (número)`
- `redondearacentesima (número)`
- `redondearamilesima (número)`

Para cada valor que introduzca el programa, muestre el valor original, el número redondeado al entero más cercano, el número redondeado a la décima más cercana, el número redondeado a la centésima más cercana y el número redondeado a la milésima más cercana.

4. Escriba declaraciones que asignen **números enteros aleatorios** a la variable **n** en los siguientes rangos:

a)  $1 \leq n \leq 2$

b)  $1 \leq n \leq 100$

c)  $0 \leq n \leq 9$

d)  $1000 \leq n \leq 1112$

e)  $-1 \leq n \leq 1$

f)  $-3 \leq n \leq 11$

5. Para cada uno de los siguientes conjuntos de números enteros, escriba una única declaración que **imprima un número al azar** del conjunto:

a) 2, 4, 6, 8, 10

b) 3, 5, 7, 9, 11

c) 6, 10, 14, 18, 22

6. **Cálculos de hipotenusa:** Defina una función llamada **hipotenusa** que calcule la hipotenusa de un triángulo rectángulo a partir de los valores de los otros dos lados. La función debe tomar **dos argumentos dobles** y devolver la hipotenusa como un **doble**. Prueba tu programa con los valores de los lados especificados en la siguiente tabla:

Lado 1	Lado 2
3.0	4.0
5.0	12.0
8.0	15.0

7. **Exponenciación:** Escribe la función **potencia\_entera**(base, exponente) que devuelva el valor de  $base^{exponente}$ . Por ejemplo, **potencia\_entera**(3, 4) = 3 \* 3 \* 3 \* 3. Suponga que el **exponente** es un entero positivo distinto de cero y que la **base** es un entero. La función **potencia\_entera** debe usar una instrucción **for** para controlar el cálculo. **No utilice ninguna función de la biblioteca matemática.**

8. **Múltiplos:** Escriba una función **esMultiple** que determine, **para un par de enteros**, si el segundo entero es múltiplo del primero. La función debe tomar dos argumentos enteros y devolver **1** (verdadero) si el segundo es múltiplo del primero y **0** (falso) en caso contrario. Use esta función en un programa que introduzca una serie de pares de enteros.

9. **Par o Impar:** Escriba un programa que introduzca una serie de enteros y los pase uno a uno a la función **esPar**, que utiliza el operador **resto** para determinar si un entero es par. La función debe tomar un argumento entero y devolver **1** si el entero es par y **0** en caso contrario.
10. **Cuadrado de Asteriscos:** Escriba una función que muestre un cuadrado sólido de asteriscos cuyo lado se especifique en el parámetro entero **lado**. Por ejemplo, si **lado** es **4**, la función muestra:

```
* * * *  
* * * *  
* * * *  
* * * *
```

11. **Mostrando un cuadrado de cualquier carácter:** Modifique la función del ejercicio anterior para formar el cuadrado con **cualquier carácter que se pasa como parámetro**. Por lo tanto, si **lado** es **5** y **caracter** es **"#"**, esta función debería imprimir:

```
#####  
#####  
#####  
#####  
#####
```

12. **Separación de dígitos:** Escriba segmentos de programa para lograr cada uno de los siguientes objetivos:

- Calcular la parte entera del **cociente** cuando se divide el entero **a** entre el entero **b**.
- Calcular el **resto** entero cuando se divide el entero **a** entre el entero **b**.
- Usar los fragmentos de programa desarrollados en (a) y (b) para escribir una función que ingrese un entero entre 1 y 32767, y lo imprima como una serie de dígitos, con dos espacios entre cada dígito. Por ejemplo, **4562** se debe imprimir como:

```
4  5  6  2
```