

- Ej. 1:** Programar una función que reciba como parámetros la longitud de los lados de un triángulo y que retorne el área del mismo. Utilizarla en un programa que solicite al usuario los datos, invoque a la función e informe el área calculada. Todos los datos y textos que se muestren por pantalla en este y en todos los ejercicios de la presente guía deben respetar estrictamente el formato de los ejemplos dados.

Ejemplo:

```
Ingrese lado 1: 2
Ingrese lado 2: 6
Ingrese lado 3: 7

El area del triangulo es = 5.56
```

Aclaración: El área de un triángulo se puede calcular como $\sqrt{p(p-a)(p-b)(p-c)}$ siendo $p = (a + b + c) / 2$ y donde a, b, c son los lados del triángulo.

- Ej. 2:** Programar la función **raiz** que recibe como parámetros un radicando **x** (*número real*) y un índice **n** (*número natural*). La función deberá retornar la raíz **n** (enésima) de **x**. Utilizarla en un programa que solicite al usuario el radicando real y el índice natural. Luego invocar a la función **raiz** e informar la raíz calculada. Ejemplo:

```
Ingrese el radicando (numero real): 14
Ingrese el índice (numero natural): 3

La raiz 3 de 14 es = 2.410142
```

Aclaración: Utilizar la función **pow**. Para ello importar la biblioteca **math** e invocar a **pow** de la siguiente forma: **math.pow(base, exponente)**

- Ej. 3:** El concepto de paridad es de fundamental importancia para la detección y corrección de errores en el almacenamiento y transmisión de datos en los sistemas informáticos. Para cada dato en binario se genera un bit adicional (bit de paridad) de manera tal que este será 1 (*uno*) si la suma de la cantidad de 1 (*unos*) presentes en el dato es impar y 0 (*cero*) si esta suma es par.

Programar la función booleana **paridad** que recibe como parámetro un número binario de hasta 8 bits y genera y retorna el bit de paridad correspondiente. Utilizarla en un programa que solicite al usuario el ingreso del número binario, invoque a la función e informe la paridad generada.

Ejemplo:

Ingrese un numero de binario de hasta 8 bits: 10111
Bit de paridad generado: 0

Ej. 4: Programar las funciones *areaCirc*, *areaCuad* y *areaNegra*.

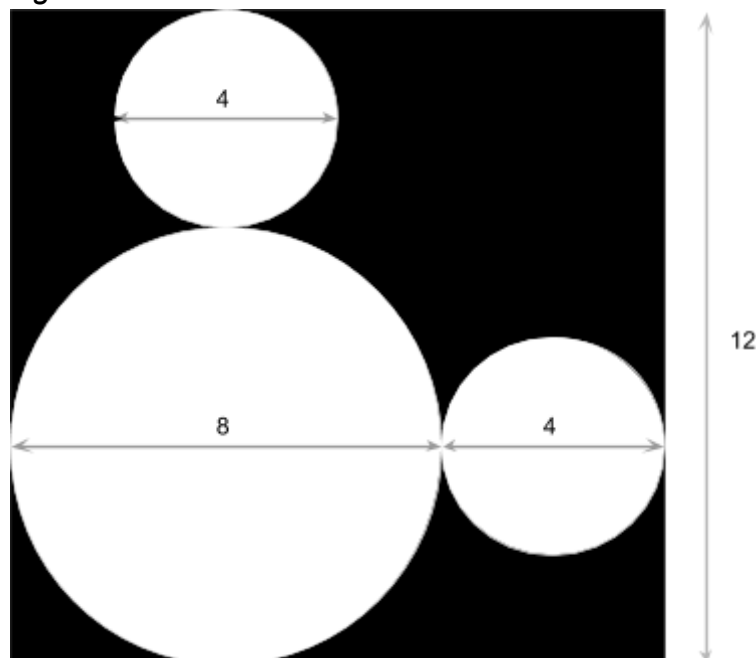
areaCirc recibe como parámetro el diámetro de un círculo y calcula y retorna el área del mismo.

areaCuad recibe como parámetro el lado de un cuadrado y calcula y retorna el área del mismo.

areaNegra recibe como parámetro el lado de un cuadrado de una figura (como la dada a continuación) y calcula y retorna el área negra resultante.

Luego utilizar las funciones en un programa que solicitará al usuario el lado del cuadrado y mostrará por pantalla el valor correspondiente para el área de color negra y además indicará el porcentaje que éste área representa con respecto al área total del cuadrado. *Notar que dado el lado del cuadrado, la proporción de tamaño entre este y los círculos siempre es la misma y esta proporción se debe deducir con los valores del ejemplo dado a continuación.*

Figura:



Ejemplo de salida:

Ingrese el lado: 12
El area negra es 68.60 y es un 47.64% del area total del cuadrado

Ej. 5: Programar una función que reciba como parámetros 2 números naturales y retorne un número natural al azar comprendido entre estos 2 números (inclusive). Debe asumirse que esta función será invocada de manera que el primer parámetro representa el límite inferior del intervalo y el segundo el límite superior.

Utilizarla en un programa que solicite al usuario los límites del intervalo e invoque a la función tres veces de la siguiente manera:

- 5.1. Invocarla con los extremos del intervalo ingresados por teclado y mostrar en pantalla el valor generado.
- 5.2. Invocarla como en el punto 5.1, pero usando como extremo inferior el valor generado en 5.1.
- 5.3. Invocarla como en el punto 5.2, pero usando como extremo superior el valor generado en 5.2.

Ejemplo:

```
Ingrese el limite inferior (numero natural): 14
Ingrese el limite superior (numero natural): 31

1-Limite inferior 14, limite superior 31. Numero generado: 25
2-Limite inferior 25, limite superior 31. Numero generado: 27
3-Limite inferior 25, limite superior 27. Numero generado: 25
```

Ej. 6: Programar una función booleana que retorne 0 o 1 al azar. Utilizarla en un programa en que se solicite al usuario dos alternativas para cada uno de los ítems de una cena: vestimenta, plato y bebida. Luego el programa muestra por pantalla la cena que resulta de elegir cada ítem al azar. Ejemplo:

```
Ingrese alternativa 1 para vestimenta: traje
Ingrese alternativa 2 para vestimenta: bermudas y musculosa
Ingrese alternativa 1 para plato: pancho con papas
Ingrese alternativa 2 para plato: langosta Thermidor
Ingrese alternativa 1 para bebida: vino
Ingrese alternativa 2 para bebida: agua gasificada

Cena al azar: bermudas y musculosa, langosta Thermidor y agua gasificada
```

Ayuda:

Recordar que un string multiplicado por 1 es el mismo string. Ej.:

`"hola"*1 -> "hola"`

Recordar que un string multiplicado por 0 es el string vacío. Ej.:

`"hola"*0 -> ""`

 PONTIFICIA UNIVERSIDAD CATÓLICA ARGENTINA	Facultad de Ingeniería y Ciencias Agrarias	INFORMÁTICA GENERAL
		Practica N° 2
	Funciones	

- Ej. 7:** Programar la función **justificado** que reciba una frase ***fra*** y un valor natural ***ancho*** como parámetros. La función debe retornar un string de un tamaño total ***ancho***, que comienza y termina con comillas simples y que contiene a la frase ***fra***, justificada a la derecha y rellena con espacios a la izquierda.
Utilizarla en un programa en que se solicite al usuario la frase y el ancho deseado, se invoque a la función y se imprima por pantalla el valor retornado.

Ejemplo:

```
Ingrese la frase: El dia es soleado
Ingrese el ancho total a ser usado: 30

    '          El dia es soleado'
```

Nota: las comillas se cuentan dentro del espacio total usado.

- Ej. 8:** Dado la función `main()`, completar el programa programando las funciones **`crearFila`** y **`crearRectangulo`** de modo que al ejecutar el programa dibuje un rectangulo de asteriscos. La función `main()` no debe ser modificado:

```
def main():
    ancho = int(input('Ingrese ancho: '))
    alto = int(input('Ingrese alto: '))
    print(crearRectangulo (ancho, alto))
```

`crearFila` recibe como parámetro un número natural ***ancho*** y retorna un string conformado por una cantidad ***ancho*** de asteriscos.

`crearRectangulo` recibe como parámetros dos números naturales ***ancho*** y ***alto***, y retorna un string que al ser mostrado por pantalla dibuja un rectángulo de ***ancho*** por ***alto*** asteriscos. Esta función invoca a **`crearFila`** para crear el string de cada fila.

Ejemplo de salida:

```
Ingrese ancho: 25
Ingrese alto: 7
*****
*****
*****
*****
*****
*****
*****
```