

# EJERCITACIÓN Nº 3: FUNCIONES Y DECISIONES

Cátedra Programación II

Septiembre 2016

## 1. Ejercitación

### 1.1. Probar en el Interprete

**EJERCICIO 1.** Usando el comando `help`, y posteriormente de haber cargado los programas desarrollados en los ejercicios 5, 6 y 7 de la práctica 2, mostrar en pantalla la documentación de las funciones desarrolladas en cada uno de ellos.

**EJERCICIO 2.** ¿Qué sucede si llamamos a una función y no realizamos nada con el resultado que esta produce— no lo asignamos a una variable, ni reutilizamos cómo parte de otra expresión?

**EJERCICIO 3.** ¿Qué sucede si utilizamos una función sin resultados cómo parte de una expresión?

### 1.2. Imprimir Vs. Devolver

**EJERCICIO 4.** Escribir una función `repite_hola` que reciba como parámetro un número entero  $n$  y escriba por pantalla el mensaje "*Hola*"  $n$  veces. Invocarla con distintos valores de  $n$ .

**EJERCICIO 5.** Escribir otra función `repite_hola` que reciba como parámetro un número entero  $n$  y devuelva la cadena formada por  $n$  concatenaciones de "*Hola*". Invocarla con distintos valores de  $n$ .

**EJERCICIO 6.** Escribir una función `repite_saludo` que reciba como parámetro un número entero  $n$  y una cadena *saludo* y escriba por pantalla el valor de *saludo*, un número  $n$  de veces. Invocarla con distintos valores de  $n$  y de *saludo*.

**EJERCICIO 7.** Escribir otra función `repite_saludo` que reciba como parámetro un número entero  $n$  y una cadena *saludo* devuelva el valor de  $n$  concatenaciones de *saludo* . Invocarla con distintos valores de  $n$  y de *saludo* .

### 1.3. Miscelaneas

**EJERCICIO 8.** Escribir dos funciones que resuelvan los siguientes problemas:

- a) Dado un número entero  $n$ , indicar si es par o no.
- b) Dado un número entero  $n$ , indicar si es primo o no.

**EJERCICIO 9.** Escribir una implementación propia de la función `abs`, que devuelva el valor absoluto de cualquier valor que reciba.

**EJERCICIO 10.** Escribir un programa, que utilizando funciones, calcule las raíces (reales o complejas) de un polinomio de segundo grado.

*Nota:* validar que las operaciones puedan efectuarse antes de realizarlas (no dividir por cero, ni calcular la raíz de un número negativo).

**EJERCICIO 11.** Escribir funciones que resuelvan los siguientes problemas:

a) Dado un año indicar si es bisiesto.

*Nota:* un año es bisiesto si es un número divisible por 4, pero no si es divisible por 100, excepto que también sea divisible por 400.

b) Dado un mes, devolver la cantidad de días correspondientes.

c) Dada una fecha (día, mes, año), indicar si es válida o no.

d) Dada una fecha, indicar los días que faltan hasta fin de mes.

e) Dada una fecha, indicar los días que faltan hasta fin de año.

f) Dada una fecha, indicar la cantidad de días transcurridos en ese año hasta esa fecha.

g) Dadas dos fechas (día1, mes1, año1), y (día2, mes2, año2), indicar el tiempo transcurrido entre ambas, en años, meses y días.

*Nota:* en todos los casos, invocar las funciones escritas previamente cuando sea posible.

**EJERCICIO 12.** Escribir dos funciones que permitan calcular:

a) La duración en segundos de un intervalo dado en horas, minutos, y segundos.

b) La duración en horas, minutos y segundos de un intervalo dado en segundos.

**EJERCICIO 13.** Usando las funciones del ejercicio anterior, escribir un programa que pida al usuario dos intervalos expresados en horas, minutos y segundos, sume sus duraciones, y muestre por pantalla la duración total en horas, minutos, y segundos.

**EJERCICIO 14.** Escribir una función que, dados cuatro números, devuelva el mayor producto de dos de ellos. Por ejemplo, si recibe los números 1, 5, (-2), (-4) debe devolver 8, que es el producto más grande que se puede obtener entre ellos ( $8 = (-2) \times (-4)$ ).

**EJERCICIO 15.** Resuelva cada uno de los siguientes items:

a) Escribir una función que dado un vector al origen (definido por sus coordenadas  $(x, y)$ ), devuelva la norma del vector, dada por  $\|(x, y)\| = \sqrt{x^2 + y^2}$

b) Utilizando las funciones anteriores, escribir una función que dados dos puntos en el plano  $(x_1, y_1)$  y  $(x_2, y_2)$ , devuelva la distancia entre ambos.

c) Escribir una función que calcule el área de un triángulo a partir de su base y su altura.

d) Utilizando las funciones anteriores escribir una función que reciba tres puntos en el plano  $(x_1, y_1)$ ,  $(x_2, y_2)$ , y  $(x_3, y_3)$ , y devuelva el área del triángulo correspondiente.

**EJERCICIO 16.** Resuelva cada uno de los siguientes items. Estos problemas ya fueron resueltos el cuatrimestre pasado utilizando el lenguaje *Dr. Racket*. Repensarlos en esta oportunidad utilizando el lenguaje *Python*. El desarrollo de cada programa deberá de constar de cada una de

las etapas vistas en clases para la resolución de problemas: análisis, especificación, -diseño, implementación, testing, y mantenimiento/modificaciones futuras.

- a) Defina la función `vol-cubo` que recibe la longitud de la arista de un cubo y calcula su volumen.
- b) Defina la función `area-cubo` que recibe la longitud de la arista de un cubo y calcula su área.
- c) Defina una función que según el valor que ingrese el usuario, se calcule el volumen, o el área de un cubo. Tener en consideración un valor por defecto, si el valor ingresado no fuera el adecuado, en dicho caso deberá calcular el área de un cubo. Reutilizar las funciones de los items anteriores.
- d) Modifique la función definida en item anterior para que, en caso de no recibir un valor apropiado nos muestre el siguiente mensaje en pantalla *"Valor no soportada por la función."*

**EJERCICIO 17.** Definir la función `observar_clima` la cual clasifica con una "etiqueta" ciertos intervalos numéricos asociados a temperaturas ambientes. La categorización de las temperaturas es la siguiente:

Intervalo	Denominación
menos de 0 grados	"Helado"
entre 0 y 15 grados	"Frío"
entre 15 y 25 grados	"Agradable"
más de 25 grados	"Caluroso"

**EJERCICIO 18.** Modificar el programa anterior para satisfacer los siguientes requerimientos:

- a) Se diferencia una categoría más, ahora si la temperatura esta entre 25 y 32 grados el mensaje deberá ser "Caluroso", pero si la temperatura excede los 32 grados deberá decir "Muy caluroso".
- b) Se establece un nuevo criterio de clasificación:

Intervalo	Denominación
menos de 0 grados	"Helado"
entre 0 y 10 grados	"Frío"
entre 10 y 15 grados	"Fresco"
entre 15 y 25 grados	"Agradable"
entre 15 y 25 grados	"Caluroso"
más de 32 grados	"Muy Caluroso"

Modificar el programa para que se adapte a este cambio.

- c) El programa ahora deberá avisarnos si es necesario utilizar protector solar o no. Para ello, se sabe que, si la temperatura es menor a los 15 no es necesario utilizar protector solar, pero por el contrario si la temperatura supera dicho valor, si se debe utilizar protector.

- d) ¿Todas las modificaciones que hicimos sobre nuestro programa fueron iguales? Existe alguna diferencia entre ellas. Explicar.

**EJERCICIO 19.** Suponiendo que el primer día del año fue lunes, escribir una función que reciba un número con el día del año (de 1 a 366) y devuelva el día de la semana que le toca. Por ejemplo: si recibe '3' debe devolver 'miércoles', si recibe '9' debe devolver 'martes'.

**EJERCICIO 20.** Programa de astrología: el usuario debe ingresar el día y mes de su cumpleaños y el programa le debe decir a qué signo corresponde.

Aries: 21 de marzo al 20 de abril.	Tauro: 21 de abril al 20 de mayo.
Geminis: 21 de mayo al 21 de junio.	Cancer: 22 de junio al 23 de julio.
Leo: 24 de julio al 23 de agosto.	Virgo: 24 de agosto al 23 de septiembre.
Libra: 24 de septiembre al 22 de octubre.	Escorpio: 23 de octubre al 22 de noviembre.
Sagitario: 23 de noviembre al 21 de diciembre.	Capricornio: 22 de diciembre al 20 de enero.
Acuario: 21 de enero al 19 de febrero.	Piscis: 20 de febrero al 20 de marzo.

**EJERCICIO 21.** Escribir un programa que reciba como entrada un entero representando un año (por ejemplo 751, 1999, o 2158), y muestre por pantalla el mismo año escrito en números romanos.

## Referencias

- [1] Think Python: How to Think Like a Computer Scientist, Allen B. Downey, 2nd Edition, Version 2.2.18.
- [2] Algoritmos y Programación I, Aprendiendo a programar usando Python como herramienta, Rosita Wachenchauzer et.al., 2016, (sin publicar).