

mcpp_taller2_Laura_Becerra

August 14, 2016

1 Taller 2

Métodos Computacionales para Políticas Públicas - URSario

Entrega: viernes 19-ago-2016 11:59 PM

[Laura Becerra] [l.becerra52@uniandes.edu.co]

1.1 Instrucciones:

- Guarde una copia de este *Jupyter Notebook* en su computador, idealmente en una carpeta destinada al material del curso.
- Modifique el nombre del archivo del *notebook*, agregando al final un guión inferior y su nombre y apellido, separados estos últimos por otro guión inferior. Por ejemplo, mi *notebook* se llamaría: mcpp_taller2_santiago_matallana
- Marque el *notebook* con su nombre y e-mail en el bloque verde arriba. Reemplace el texto "[Su nombre acá]" con su nombre y apellido. Similar para su e-mail.
- Desarrolle la totalidad del taller sobre este *notebook*, insertando las celdas que sea necesario debajo de cada pregunta. Haga buen uso de las celdas para código y de las celdas tipo *markdown* según el caso.
- Recuerde salvar periódicamente sus avances.
- Cuando termine el taller:
 1. Descárguelo en PDF.
 2. Suba los dos archivos (.pdf y .ipynb) a su repositorio en GitHub antes de la fecha y hora límites.

(El valor de cada ejercicio está en corchetes [] después del número de ejercicio.)

1.2 1. [1]

[Pensar como un computador] Considere el siguiente código: `if x > 2: if y > 2: z = x + y print("z es", z) else: print("x es", x)` ¿Cuál es el resultado si

- a) $x = 2, y = 5$?
- b) $x = 3, y = 1$?
- c) $x = 1, y = 1$?

d) $x = 4, y = 3$?

- a) primero defino $x=2$ y $y=5$, dado que x no es mayor a 2 porque es igual a 2 el computador evalua else. Por tanto el resultado es x es 2.
 - b) primero defino $x=3$ y $y=1$, dado que y no es mayor a 2 porque es igual a 1 el computador no da ningun resultado porque else tiene print x y no y .
 - c) primero defino $x=1$ y $y=1$, dado que x y y no son mayores a 2 porque son iguales a 1 entonces el computador evalua else. Por tanto el resultado es x es 1.
 - d) primero defino $x=4$ y $y=3$, dado que x y y son mayores a 2 entonces el computador hace la suma y el resultado es z es 7
-

1.3 2. [1][Pensar como un computador] ¿Cuál es el resultado del siguiente código y cuántas veces se recorre el loop?

$i = 0$ while $i < 10$: $i = i + 1$ if i print(i) primero valida el cero y como este es menor a 10 va a tomar los i menores a 10 y para cada i le va a sumar 1, después va a buscar los resultados que cuyo residuo dividiendolos en dos sea cero y los va a imprimir en la pantalla. Al final el resultado es una impresión de los números pares (2, 4, 6, 8, 10)

1.4 3. [1][Pensar como un computador] ¿Cuál es el resultado del siguiente código y cuántas veces se recorre el loop?

$i = 0$ while $i > 10$: $i = i + 1$ if i print(i) Primero valida el cero y como este es no es mayor a 10 no va a correr el if por ende el computador no imprime ningun resultado.

1.5 4. [2]

Escriba un programa que pida al usuario ingresar un número entero, y que imprima “par” si el número es par e “impar” si el número es impar. Agregue a su programa un código que genere una advertencia en caso de que el usuario ingrese algo diferente a un número entero: “Error. El usuario debe ingresar un número entero.” (Investigue por su cuenta cómo lograr dicha validación y la generación del mensaje.)

```
In [4]: num = eval(input("Ingrese un numero entero: "))
        if (num % 2) == 0:
            print("{0} es par".format(num))
        elif (num % 2) == 1:
            print("{0} es impar".format(num))
        else:
            print("error")
```

```
Ingrese un numero entero: 2.5
error
```

1.6 5. [2]

Escriba un for loop que imprima todos los múltiplos de 3 desde 40 hasta 0 en orden decreciente. Esto es, 39, 36, 33, ..., 3, 0.

```
In [15]: for n in reversed(range (0, 41)):  
        if n % 3 == 0:  
            print (n)
```

```
39  
36  
33  
30  
27  
24  
21  
18  
15  
12  
9  
6  
3  
0
```

1.7 6. [2]

Escriba un loop que imprima todos los números entre 6 y 30 que no son divisibles por 2, 3 o 5.

```
In [28]: for n in range (6, 31):  
        for i in (2,3,5):  
            if n % i != 0:  
                print (n,"no es divisible por", i)
```

```
6 no es divisible por 5  
7 no es divisible por 2  
7 no es divisible por 3  
7 no es divisible por 5  
8 no es divisible por 3  
8 no es divisible por 5  
9 no es divisible por 2  
9 no es divisible por 5  
10 no es divisible por 3  
11 no es divisible por 2  
11 no es divisible por 3  
11 no es divisible por 5
```

```
12 no es divisible por 5
13 no es divisible por 2
13 no es divisible por 3
13 no es divisible por 5
14 no es divisible por 3
14 no es divisible por 5
15 no es divisible por 2
16 no es divisible por 3
16 no es divisible por 5
17 no es divisible por 2
17 no es divisible por 3
17 no es divisible por 5
18 no es divisible por 5
19 no es divisible por 2
19 no es divisible por 3
19 no es divisible por 5
20 no es divisible por 3
21 no es divisible por 2
21 no es divisible por 5
22 no es divisible por 3
22 no es divisible por 5
23 no es divisible por 2
23 no es divisible por 3
23 no es divisible por 5
24 no es divisible por 5
25 no es divisible por 2
25 no es divisible por 3
26 no es divisible por 3
26 no es divisible por 5
27 no es divisible por 2
27 no es divisible por 5
28 no es divisible por 3
28 no es divisible por 5
29 no es divisible por 2
29 no es divisible por 3
29 no es divisible por 5
```

1.8 7. [4]

Escriba un programa llamado “Adivine ni número”. El computador generará aleatoriamente un entero entre 1 y 100. El usuario digita un número y el computador responde “Menor” si el número aleatorio es menor que el escogido por el usuario, “Mayor” si el número aleatorio es mayor, y “¡Correcto!” si el usuario adivina el número. El jugador puede continuar ingresando números hasta que adivine correctamente.

Ejemplo: - El número aleatorio es 79. - El computador muestra el texto “Adivine el número

entre 1 y 100:" y espera a que el usuario lo digite. - El usuario digita el número que está abajo en *itálicas*. - El computador devuelve uno de tres textos, según el caso: "Mayor", "Menor", o "¡Correcto!".

```
In [ ]: import random
```

```
print("\tBienvenido a 'Adivine mi numero'!")  
  
print("\nEstoy pensando un número entre 0 y 100.")
```

```
# set the initial values
```

```
the_number = random.randint(1, 100)
```

```
guess = int(input("Adivina: "))
```

```
# guessing loop
```

```
while guess != the_number:
```

```
    if guess > the_number:
```

```
        print("Menor...")
```

```
    else:
```

```
        print("Mayor...")
```

```
    guess = int(input("Adivina: "))
```

```
if guess == the_number:
```

```
    print("Correcto", the_number)
```

```
input("\n\nPress the enter key to exit.")
```

```
Bienvenido a 'Adivine mi numero'!
```

```
Estoy pensando un número entre 0 y 100.
```

```
Adivina: 50
Mayor...
Adivina: 70
Mayor...
Adivina: 80
Mayor...
Adivina: 90
Menor...
Adivina: 85
Menor...
Adivina: 84
Menor...
Adivina: 83
Menor...
Adivina: 82
Correcto 82
```

In []:

```
Adivine el número entre 1 y 100: 40 Mayor
Adivine el número entre 1 y 100: 70 Mayor
Adivine el número entre 1 y 100: 80 Menor
Adivine el número entre 1 y 100: 77 Mayor
Adivine el número entre 1 y 100: 79 ¡Correcto!
```

¿Cómo generar números aleatorios en Python?

- Al comienzo de su programa escriba: `import random`
- Para generar un número aleatorio entre 1 y 100 escriba: `random.randint(1, 100)`

Pistas:

- Piense en qué estructuras de control le sirven para resolver el problema.
 - ¿Cómo determina si el número es mayor, menor o correcto?
 - ¿Cómo le da turnos adicionales al usuario para adivinar, dependiendo de si en el turno anterior adivinó o no?
-