

mcpp_taller2_julian_ramirez

August 15, 2019

1 Taller 2

Métodos Computacionales para Políticas Públicas - UROSARIO

Entrega: viernes 16-ago-2019 11:59 PM

Julián Santiago Ramírez julians.ramirez@urosario.edu.co

1.1 Instrucciones:

- Guarde una copia de este *Jupyter Notebook* en su computador.
- 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) $x = 4, y = 3$?

1.2.1 a)

Si $x=2$ y $y=5$:

RTA: x es 2

1.2.2 b)

Si $x=3$ y $y=1$:

RTA: No imprime nada

1.2.3 c)

Si $x=1$ y $y=1$:

RTA: x es 1

1.2.4 d)

Si $x=4$ y $y=3$:

RTA: 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)

1.3.1 Rta:

imprime todos los numeros desde 1 a 10 que sean multiplos de 2 , por tanto el resultado es: 2,4,6,8,10. Como el valor de "i" cumple la condicion al tomar el valor de 10 entonces el loop se recorre 9 veces.

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)

1.4.1 Rta:

No imprime nada, no se recorre ni una vez dado que la condición esta mal especificada. Por tanto en estos casos toca parar el kernel dado que el computador se queda en esa condición.

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 [12]: #### Pedimos al usuario que ingrese un numero
         ingreso=input('Ingrese un número entero: ')
         try:
             numero=int(ingreso)
             print("Numero correctamente ingresado")
             ## revisamos si es par o no
             if numero==0:
                 print("#####")
                 print("número ingresado es cero")
             elif numero%2==0:
                 print("#####")
                 print("número ingresado es par")
             else:
                 print("#####")
                 print("número ingresado es impar")
         except:
             print("Error. El usuario debe ingresar un número entero")
```

```
Ingrese un número entero: 4
Numero correctamente ingresado
#####
número ingresado es par
```

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 [4]: ### Hacemos un for recorriendo de forma decreciente, el parentesis se le de la sig man.
         ### (Desde donde empezamos el "for", Hasta donde llegamos en el "for" sin incluir el n
         for i in range(40,-1,-1):
             if (i%3==0):
                 print(i)
```

```
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 [7]: ### Mostrare los números que no son divisibles por 2 o por 3 o por 5, es necesario que  
### divisibles por ninguno de esos tres números  
for i in range(6,31):  
    if i%2!=0 and i%3!=0 and i%5!=0:  
        print (i)
```

7
11
13
17
19
23
29

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 *italicas*. - El computador devuelve uno de tres textos, según el caso: "Mayor", "Menor", o "¡Correcto!".

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!

```

In [ ]: import random as r

numero_ale=r.randint(1,100)

ingreso=input('Adivine el número entre 1 y 100:')
try:
    numero=int(ingreso)
    print("Numero correctamente ingresado")
    ## revisamos si es par o no

    while numero_ale!=numero:
        if numero_ale>numero:
            print("Mayor")
            ingreso=input('Adivine el número entre 1 y 100:')
            try:
                numero=int(ingreso)
                print("Numero correctamente ingresado")
            except:
                print("Error. El usuario debe ingresar un número entero")
        else:
            print("Menor")
            ingreso=input('Adivine el número entre 1 y 100:')
            try:
                numero=int(ingreso)
                print("Numero correctamente ingresado")
            except:
                print("Error. El usuario debe ingresar un número entero")

    print("¡Correcto!")
    print('\n')
    print("El número aleatorio era "+ str(numero_ale))

except:
    print("Error. El usuario debe ingresar un número entero")

```

```

Adivine el número entre 1 y 100:5
Numero correctamente ingresado
Mayor
Adivine el número entre 1 y 100:op
Error. El usuario debe ingresar un número entero
Mayor

```

¿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?
-