



## TP N.º 7 - Computación

### Ejercicio 1

Calcular la suma de la diferencia entre pares consecutivos en un array de orden descendiente.

**Ejemplo:**

```
sum_of_differences([1, 2, 10])
```

Return 9

Descending order: [10, 2, 1]

Sum:  $(10 - 2) + (2 - 1) = 8 + 1 = 9$

Si el array esta vacio o tiene solo un elemento debe retornar 0.

**Testing:**

```
test.assert_equals(sum_of_differences([1, 2, 10]), 9)
```

```
test.assert_equals(sum_of_differences([-3, -2, -1]), 2);
```

```
test.assert_equals(sum_of_differences([-17, 17]), 34);
```

### Ejercicio 2

Una maquina ATM permite 4 o 6 dígitos PIN esos PIN no pueden contener nada excepto 4 o 6 dígitos.

La función devuelve True si es un PIN valido, de otra forma devuelve False.

**Ejemplo:**

"1234" → True

"12345" → False

"a234" → False

**Testing:**

```
test.assert_equals(validate_pin("12345"),False")
```

```
test.assert_equals(validate_pin("9765"),True")
```

```
test.assert_equals(validate_pin("a345"),False")
```



## UNIVERSIDAD DE MENDOZA

### FACULTAD DE INGENIERIA

#### Ejercicio 3

Si témenos un array desordenado que contiene números enteros del 0 al 100 inclusive. Sin embargo hay un solo numero que falta. Escribir una función que me devuelva el numero que falta.

##### Testing:

```
nums = list(range(0,101))  
  
nums.remove(5)  
  
test.assert_equals(missing_no(nums), 5)  
  
nums = list(reversed(range(0,101)))  
  
nums.remove(10)  
  
test.assert_equals(missing_no(nums), 10)
```

#### Ejercicio 4:

Para el juego de BINGO, recibirás un array de 10 números de 1 a 26 como entrada. Puede haber números duplicados.

Cada numero corresponde a una letra del abecedario A=1, B=2, etc. Escribir una función que me retorne "WIN" si los números que contienen el array pueden formar la palabra BINGO, no importa el orden en que estén. De otra forma si no se puede formar la palabra BINGO me tiene que devolver FALSE.

```
test.assert_equals(bingo([1, 2, 3, 4, 5, 6, 7, 8, 9, 10]), "LOSE")  
  
test.assert_equals(bingo([20, 12, 23, 14, 6, 22, 12, 17, 2, 26]), "LOSE")  
  
test.assert_equals(bingo([1, 2, 3, 7, 5, 14, 7, 15, 9, 10]), "WIN")  
  
test.assert_equals(bingo([5, 2, 13, 7, 5, 14, 17, 15, 9, 10]), "WIN")
```

#### Ejercicio 5

Dada una lista de enteros únicos, y dos enteros a y b. Crear una función que se fije si a y b aparecen consecutivos en el array, en ese caso devolver True, en caso contrario devolver False.

##### Testing:

```
test.assert_equals(consecutive([1, 3, 5, 7], 3, 7), False)  
  
test.assert_equals(consecutive([1, 3, 5, 7], 3, 1), True)  
  
test.assert_equals(consecutive([1, 6, 9, -3, 4, -78, 0], -3, 4), True)
```