

Instituto Tecnológico de Costa Rica

Escuela de Computación

Introducción a la Programación / Taller de programación.

Profesor: Jeff Schmidt Peralta



EJERCICIOS INICIALES SOBRE RECURSIVIDAD.

I SEMESTRE 2018

1. Escriba una función recursiva de pila `suma_impar(num)` que reciba un número entero y sume los dígitos que sean impares.

```
>>> suma_impar(482401)    >>> suma_impar(3579)
1                          24
```

2. Escriba una función recursiva de pila `cuenta_par(num)` que reciba un número entero y cuente los dígitos que sean pares.

```
>>> cuenta_par(482401)    >>> cuenta_par(3579)
5                          0
```

3. Escriba una función booleana `iguales(num)` que recibe un número entero y verifica si el primer y el último dígito del número son iguales. La función debe comportarse de la siguiente forma:

```
>>> iguales(80642)
False
>>> iguales(351733)
True
```

4. Escriba una función booleana `suma10(num)` que recibe un número entero y verifica si la suma de sus dígitos es mayor o igual a 10. La función debe comportarse de la siguiente forma:

```
>>> suma10(80642)
True
>>> suma10(200412)
False
```

5. Escriba una función booleana `cuenta_dig(num, dig)` que recibe un número entero y un dígito válido y cuenta las veces que aparece el dígito en el número. La función debe comportarse de la siguiente forma:

```
>>> cuenta_dig(93235, 3)
2
>>> cuenta_dig(93735, 2)
0
```

6. Escriba una función `revise_num` que reciba un número (que debe ser entero) y retorne una tupla que tenga la siguiente forma:

(cantidad-dígitos-entre 0 y 4, cantidad-dígitos-entre 5 y 9)

La función debe retornar su resultado en forma similar al siguiente ejemplo:

```
>>> revise_num(482401)
(5, 1)
>>> revise_num(4)
(1, 0)
```

7. Escriba una función booleana `todos_pares(num)` que recibe un número entero y verifica si el número dado tiene todos sus dígitos pares. La función debe comportarse de la siguiente forma:

```
>>> todos_pares(80642)
True
>>> todos_pares(201462)
False
```

8. Escriba una función booleana `hay_par(num)` que recibe un número entero y verifica si el número dado tiene al menos un dígito par. La función debe comportarse de la siguiente forma:

```
>>> hay_par(80642)
True
>>> hay_par(351733)
False
```