

Instituto Tecnológico de Costa Rica

Escuela de Computación

Introducción a la Programación / Taller de programación.

Profesor: Jeff Schmidt Peralta



EJERCICIOS SOBRE RECURSIVIDAD. I SEMESTRE 2018

1. Escriba una función booleana `mas_nueve(num)` que recibe un número entero y verifica si el número dado tiene más de nueve dígitos. La función debe comportarse de la siguiente forma:

```
>>> mas_nueve(9003500919)
True
>>> mas_nueve(8373)
False
```

2. Escriba una función booleana `sumax(num1, num2)` que recibe un número entero y verifica si la suma de sus dígitos es mayor o igual al otro número. La función debe comportarse de la siguiente forma:

```
>>> sumax(80642, 6)
True
>>> sumax(200412, 10)
False
```

3. Un número es palíndromo si puede leerse igual de izquierda a derecha que de derecha a izquierda. Escribir una función llamada `palindromo(num)` que reciba un número y retorne `True` si el número recibido es un palíndromo y `False` si no. Por ejemplo:

```
>>> palindromo(38583)
True
>>> palindromo(4)
True
>>> palindromo(2442)
True
>>> palindromo(1010010)
False
```

4. Hacer una función llamada `invertir(num)`, que reciba un número entero e invierta el orden de sus elementos (sin utilizar funciones de conversión de datos):

```
>>> invertir(1234)
4321
```

5. Escriba una función recursiva en pila en Python `exactos(num)` que determine la cantidad de divisores exactos que tiene un número entero mayor que cero.

```
>>> exactos(28)
6
>>> exactos(13)
2
```

6. Escriba una función llamada `todos_div` que recibe un número y un dígito y retorne `True` si todos los dígitos del número son divisores exactos del dígito recibido y `False` en caso contrario.

```
>>> todos_div(46248, 2)
True
>>> todos_div(16248, 2)
False
```

7. Escriba una función recursiva de pila `cuenta(dig, num)` que reciba un dígito y un número entero y obtenga dos contadores, el primero compuesto por la cantidad de dígitos que son divididos en forma exacta por el dígito dado y el segundo compuesto por la cantidad de dígitos que no son divididos en forma exacta al dígito dado. Se sugiere implementar utilizando funciones `lambda`. La función debe comportarse como los siguientes ejemplos:

```
>>> cuenta(2, 1029099)      >>> cuenta(3, 396366)
(1, 6)                      (6, 0)
>>> cuenta(5, 15)           >>> cuenta(7, 8)
(1, 1)                      (0, 1)
```

8. Escribir una función en recursividad de pila `cambia`, que reciba un número entero y cambie los dígitos que sean un divisor de 2, o sea, 2, 4, 6, 8 por un cero, retornando un número entero.

```
>>> cambia(1488)      >>> cambia(73571)
1000                  73571
>>> cambia(4275)
75
```

9. Escriba una función `restel` que reciba un número (que debe ser entero) y retorne el número con un 1 restado a cada uno de sus dígitos, exceptuando el caso que el dígito sea cero.

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

```
>>> restel(482401)
371300
>>> restel(4)
3
```