

EJERCICIOS PARA II EXAMEN PARCIAL.

1. Escriba una función por medio de iteración `digitos` que reciba un número (que debe ser entero) y retorne una tupla que tenga la siguiente forma:

(cantidad-dígitos-mayores-que-5, cantidad-dígitos-menores-o-iguales-que-5)

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

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

2. Hacer una función iterativa `forma_par(num)` que recibe un número entero y forma otro número con los dígitos pares del número de entrada.

```
>>> forma_par(93235)      >>> forma_par(423114)
2                          424
>>> forma_par(4444)      >>> forma_par(113377)
4444                      0
```

3. Escriba una función iterativa `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
```

4. Implementar la versión iterativa del algoritmo que calcula el factorial para un número `n`.

```
>>> factorial(6)
720
```

5. Implementar la versión iterativa del algoritmo que dado un valor para `n` encuentra el valor dentro de la sucesión de Fibonacci para ese `n`.

```
>>> fib(7)                >>> fib(35)
13                        24157816
```

6. Un número es palíndromo si puede leerse igual de izquierda a derecha que de derecha a izquierda. Escribir una función usando iteración 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)      >>> palindromo(2442)
True                        True
>>> palindromo(4)          >>> palindromo(1010010)
True                        False
```

7. Escriba una función iterativa `elimine(dig, num)` que reciba un dígito y una lista y elimine la primera aparición del elemento en la lista (dígito menos significativo).

```
>>> elimine (4, 45348)
4538
```