

Conversão de Bases em Python 3

Gabriel Veiga

Função

`quadrado(n) => n*n`

Overview

- `to_base(numero_texto, base_original, base_final)`
 - a. `__get_decimal_value(char)`
 - `'F' => 15`
 - b. `__get_symbol(n)`
 - `15 => 'F'`
 - c. Converte número em texto pra base 10
 - d. Converte base 10 pra base_final

Como

$$x = n_k \cdot b^k + n_{k-1} \cdot b^{k-1} + \dots + n_0 \cdot b^0$$

```
# converte valor original para base "10"
```

```
decval = 0
```

```
origbase_exp = len(val) - 1
```

```
for char in val:
```

```
    # decimal_value corresponde a um unico digito
```

```
    decimal_value = __get_decimal_value(char)
```

```
    if decimal_value < origbase:
```

```
        decval += decimal_value * ( origbase ** origbase_exp )
```

```
        origbase_exp -= 1
```

```
    else:
```

```
        raise ValueError('Algarismo {} nao faz sentido para origbase {}'.format(char, origbase))
```

Como

```
# converte valor base 10 para nova base
```

```
if newbase == 10:  
    return decval, 10
```

```
numarr = []
```

```
aux = decval  
while aux >= newbase:  
    numarr.append(aux % newbase)  
    aux //= newbase
```

```
numarr.append(aux)
```

```
returnstr = ''.join([__get_symbol(n) for n in numarr[::-1]])
```

$$3864 = 7 \cdot 8^3 + 4 \cdot 8^2 + 3 \cdot 8^1 + 0 \cdot 8^0 = 7430_8$$

$$7 \cdot 8^2 + 4 \cdot 8^1 + 3 \cdot 8^0 = \frac{7430_8}{8} = 483, \text{ resto } 0$$

$$7 \cdot 8^1 + 4 \cdot 8^0 = \frac{7430_8}{8^2} = 60, \text{ resto } 3$$

$$7 \cdot 8^0 = \frac{7430_8}{8^3} = 7, \text{ resto } 4. \quad 7 < 8.$$

$$0, 3, 4, 7 \rightarrow 7430$$

Entrada, saída

```
if __name__ == '__main__':  
  
    number = input('Numero, por favor: ').upper()  
    base = int(input('Base original: '))  
    print()  
  
    print('Numero {}({}).'.format(number, base))  
    print()  
  
    newbase = int(input('Para qual base agora? '))  
    print()  
    print('Numero {}({}).'.format(*to_base(number, base, newbase)))
```

