

Exercícios referentes ao capítulo 11:

### Exercicio\_11.01:

```
fin = open('/root/words.txt')
def load_dic():
    aux = dict()
    for line in fin:
        aux[line.strip()] = ''
    return aux
word = str(input('Insira a palavra a ser buscada: '))
aux = load_dic()

if word in aux:
    print('Palavra esta no dicionario')
else:
    print('Palavra não esta no dicionario')
```

### Exercicio\_11.02:

```
def invert_dic(aux):
    cont = dict()
    for key in aux:
        val = aux[key]
        cont.setdefault(val, [])
        cont[val].append(key)
    return r
aux = {'a' : 1, 'p' : 1, 'r' : 2, 't' : 1, 'o' : 1}
cont = invert_dic(aux)
print(cont)
```

### Exercicio\_11.03:

```
aux = {}
def ack(m, n):
    if m == 0:
        return n+1
    if n == 0:
        return ack(m-1, 1)
    if (m, n) in aux:
        return aux[m, n]
    else:
        aux[m, n] = ack(m-1, ack(m, n-1))
    return aux[m, n]
print(ack(3, 4))
print(ack(3, 6))
print(aux)
```

### Exercicio\_11.04:

```
def has_duplicates(key):
    aux = dict()
    for cont in key:
        if cont in aux:
            return True
        aux.setdefault(cont)
    return False
x = [1, 2, 3, 4, 2]
y = ['a', 'c', 'a', 'j']
z = [3, 5, 1, 6, 7]
print(has_duplicates(x))
print(has_duplicates(y))
print(has_duplicates(z))
```

### Exercicio\_11.05:

```
fin = open('/root/words.txt')
def load_dic():
    aux = dict()
    for line in fin:
        word = line.strip()
        aux.setdefault(word)
    return aux
def rotate_word(word, n):
    x = ''
    for cont in word:
        x += chr(ord(cont) + n)
    return x
words = load_dic()
for word in words:
    for i in range(1, 25):
        x = rotate_word(word, i)
        if x in words:
            print(x + ' + ' + word)
```