

CORSO DI FONDAMENTI DI INFORMATICA

ESERCITAZIONE 8

PROF. GABRIELE DI STEFANO - DR. FRANCESCO GALLO

Blocco 0 - III Piano - francesco.gallo@univaq.it

Web: <http://people.disim.univaq.it/~francesco.gallo/fondamenti.html>

Dizionari

Un dizionario contiene una raccolta di indici, chiamati **chiavi**, e una raccolta di valori. Ciascuna chiave è associata ad un unico valore. L'associazione tra una chiave e un valore è detta **coppia chiave-valore** o anche **elemento**.

```
# Costruiamo un dizionario che trasforma le parole dall'inglese  
# all'italiano
```

Dizionari

Un dizionario contiene una raccolta di indici, chiamati **chiavi**, e una raccolta di valori. Ciascuna chiave è associata ad un unico valore. L'associazione tra una chiave e un valore è detta **coppia chiave-valore** o anche **elemento**.

```
# Costruiamo un dizionario che trasforma le parole dall'inglese  
# all'italiano
```

```
def diz_eng_it():  
  
    eng2it = dict()  
  
    eng2it = {'one': 'uno', 'two': 'due', 'three': 'tre'}  
  
    return eng2it
```

Dizionari

Un dizionario contiene una raccolta di indici, chiamati **chiavi**, e una raccolta di valori. Ciascuna chiave è associata ad un unico valore. L'associazione tra una chiave e un valore è detta **coppia chiave-valore** o anche **elemento**.

```
# Costruiamo un dizionario che trasforma le parole dall'inglese  
# all'italiano
```

```
def diz_int_it_ver2():  
  
    eng2it = dict()  
  
    eng2it = { 1: 'one', 2: 'two', 3: 'three'}  
  
    return eng2it
```

Dizionari

Un dizionario contiene una raccolta di indici, chiamati **chiavi**, e una raccolta di valori. Ciascuna chiave è associata ad un unico valore. L'associazione tra una chiave e un valore è detta **coppia chiave-valore** o anche **elemento**.

Lunghezza di un dizionario

Dizionari

Un dizionario contiene una raccolta di indici, chiamati **chiavi**, e una raccolta di valori. Ciascuna chiave è associata ad un unico valore. L'associazione tra una chiave e un valore è detta **coppia chiave-valore** o anche **elemento**.

Lunghezza di un dizionario

```
def lunghezza_diz(d):  
    return len(d)
```

Dizionari

Uso dell'operatore **in**: informa se qualcosa compare come **chiave** nel dizionario

Dizionari

Uso dell'operatore **in**: informa se qualcosa compare come **chiave** nel dizionario

```
def is_in(d, key):  
    return key in d
```

Dizionari

Uso dell'operatore **in**: informa se qualcosa compare come **chiave** nel dizionario

```
def is_in(d, key):  
    return key in d
```

E se volessi conoscere i **valori** contenuti nel dizionario?

Dizionari

Uso dell'operatore **in**: informa se qualcosa compare come **chiave** nel dizionario

```
def is_in(d, key):  
    return key in d
```

E se volessi conoscere i **valori** contenuti nel dizionario?

```
def values_is_in(d):  
    return d.values()
```

Esercizio1

Data una stringa S , contare quante vi compare ciascuna lettera.

SOL1. Utilizzare le liste e la funzione **ord**

SOL2. Utilizzare i dizionari

Esercizio1

Data una stringa S , contare quante vi compare ciascuna lettera.

SOL1. Utilizzare le liste e la funzione **ord**

SOL2. Utilizzare i dizionari

```
def conta_caratteri(s):  
    d = dict()  
    for c in s:  
        if c not in d:  
            d[c] = 1  
        else:  
            d[c] += 1  
    return d
```

Esercizio

Dato il file `words.rtf` (salvatelo in formato `.txt`), scrivere delle procedure che forniscano il seguente output:

- per ogni parola, restituire il carattere dell'alfabeto con maggiore frequenza (qualora ci fossero frequenze uguali di valore massimo, prendere l'ultimo carattere di valore massimo)
- restituire la parola di lunghezza massima (parola più lunga che inizia con a, parola più lunga che inizia con b, etc)

Attenzione: la soluzione proposta deve applicare in modo corretto il concetto di modularizzazione