

anno 2014-15
Introduzione all'Algoritmica per i Licei

9 – Analisi di testi

Elio Giovannetti
Dipartimento di Informatica
Università di Torino

1 marzo 2015



Quest'opera è distribuita con [Licenza Creative Commons](http://creativecommons.org/licenses/by-nc-sa/3.0/it/legalcode)
[Attribuzione - Non commerciale - Condividi allo stesso modo 3.0 Italia.](http://creativecommons.org/licenses/by-nc-sa/3.0/it/legalcode)
<http://creativecommons.org/licenses/by-nc-sa/3.0/it/legalcode>

Una funzione per estrarre le parole di un testo

```
def estraiParole(testo):  
    n = len(testo)  
    i = 0  
    parole = []    la lista di parole, inizialmente vuota  
    while i < n:  scandisco tutto il testo, un caratt. per volta  
        while i < n and not testo[i].isalpha():  
            i = i+1    cerco il primo caratt. di una nuova parola  
        parola = ""  inizio una nuova parola  
        while i < n and testo[i].isalpha():  
            parola = parola + testo[i]  
            i = i+1    costruisco la parola  
        parole.append(parola)  la aggiungo alla lista di parole  
    return parole
```

Un dizionario delle frequenze dei personaggi

```
personaggi =  
{ "Renzo":0, "Lorenzo":0, "Lucia":0, "Abbondio":0,  
  "Rodrigo":0, "Agnese":0, "Perpetua":0,  
  "Cristoforo":0, "Federigo":0, "innominato":0}  
  
inputFile = open('iPromessiSposi.txt', 'r')  
text = inputFile.read()  
  
for parola in estraiParole(text):  
    if parola in personaggi:  
        personaggi[parola] = personaggi[parola] + 1  
  
print(personaggi)  
  
Se li vogliamo scritti in ordine decrescente di frequenza:  
print(sorted(personaggi.items(),  
            key=operator.itemgetter(1), reverse=True))
```

In realtà in Python è già predefinita una funzione **count**, che conta quante volte una sottostringa compare in una stringa

Usando tale potente funzione, non c'è bisogno di definire la funzione **estraiParole**. Basta scrivere:

```
for parola in personaggi:  
    personaggi[parola] = text.count(parola)
```