

Laurea Triennale in Ingegneria Gestionale

Corso di Fondamenti di Informatica A.A. 2016/2017

DIPARTIMENTO DI INGEGNERIA INFORMATICA
AUTOMATICA E GESTIONALE ANTONIO RUBERTI



SAPIENZA
UNIVERSITÀ DI ROMA

Files

Argomenti della lezione

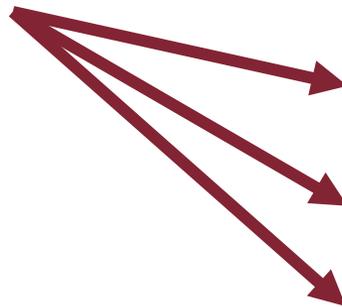
- Gestione del carattere di escape
- Rimozione caratteri speciali
- Lettura di singole parole da file
- Lettura di singoli caratteri da file
- Scrittura su file

Gestione del carattere di escape '\n'

- senza rimozione

```
f = open("input.txt", "r")
for riga in f:
    print(riga)
f.close()
```

Ogni riga del file viene stampata, **incluso** il carattere di escape '\n'



Mario Gianni

Anno Accademico 2016-2017

Fondamenti di Informatica

Gestione del carattere di escape '\n'

- con rimozione

```
f = open("input.txt", "r")
for riga in f:
    nuovaRiga = riga.rstrip()
    print(riga)
f.close()
```

Ogni riga del file viene stampata, **escluso** il carattere di escape '\n'



```
Mario Gianni
Anno Accademico 2016-2017
Fondamenti di Informatica
```

La funzione `rstrip()` crea una nuova stringa in cui tutti gli spazi bianchi *alla fine della riga*, **incluso** il carattere di escape '\n', sono rimossi.

Rimozione caratteri speciali

```
Mario Gianni! ,?  
Anno Accademico 2016-2017..!?  
Fondamenti di Informatica!!!!?
```

```
f = open("input.txt", "r")  
for riga in f:  
    nuovaRiga = riga.rstrip("! . ?")  
    print(riga)  
f.close()
```

Ogni riga del file viene stampata, **escluso** il carattere di escape '\n' ed **esclusi** tutti i caratteri passati in ingresso alla funzione `rstrip()`



```
Mario Gianni  
Anno Accademico 2016-2017  
Fondamenti di Informatica
```

Rimozione caratteri speciali

- rimozione alla fine della riga

```
nuovaRiga = riga.rstrip()
```

- rimozione all'inizio della riga

```
nuovaRiga = riga.lstrip()
```

- rimozione all'inizio e alla fine della riga

```
nuovaRiga = riga.strip()
```

Invocate con argomenti, le funzioni `rstrip()`, `rstrip()` e `strip()` creano una nuova stringa in cui tutti i caratteri speciali, specificati negli argomenti, all'inizio, alla fine, all'inizio ed alla fine, rispettivamente, sono stati rimossi.

Letture di singole parole da file

- Carattere separatore: spazio

```
Mario Gianni  
Anno Accademico 2016-2017  
Fondamenti di Informatica
```

```
f = open("input.txt", "r")  
for riga in f:  
    lista = riga.split()  
    print(lista)  
f.close()
```

```
['Mario', 'Gianni']  
['Anno', 'Accademico', '2016-2017']  
['Fondamenti', 'di', 'Informatica']
```

Lettura di singole parole da file

- Carattere separatore: **virgola**

```
Mario, Gianni  
Anno, Accademico, 2016-2017  
Fondamenti, di, Informatica
```

```
f = open("input.txt", "r")  
for riga in f:  
    lista = riga.split(",")  
    print(lista)  
f.close()
```

Attenzione: l'ultimo
elemento di ogni lista
contiene il carattere di
escape '\n'

```
['Mario', 'Gianni\n']  
['Anno', 'Accademico', '2016-2017\n']  
['Fondamenti', 'di', 'Informatica\n']
```

Lettura di singole parole da file

- Carattere separatore: **virgola**

```
Mario, Gianni  
Anno, Accademico, 2016-2017  
Fondamenti, di, Informatica
```

Prima **eliminiamo** il carattere di escape '\n' e **dopo** creiamo la lista contenente le parole di ogni singola riga

```
f = open("input.txt", "r")  
for riga in f:  
    nuovaRiga = riga.strip()  
    lista = nuovaRiga.split(",")  
    print(lista)  
f.close()  
  
['Mario', 'Gianni']  
['Anno', 'Accademico', '2016-2017']  
['Fondamenti', 'di', 'Informatica']
```

Esempio: Leggere una sequenza di interi, memorizzati in un file, separati da una virgola e trovare il massimo

3,7,9,15
19,41,15,16
1,2,91

→

91

```
massimo = 0
f = open('input.txt', 'r')
for riga in f:
    nuovaRiga = riga.strip()
    l = nuovaRiga.split(',')
    for i in l:
        n = int(i)
        if(n >= massimo):
            massimo = n
f.close()
print(massimo)
```

Letture di singoli caratteri da file

```
f = open("input.txt", "r")
carattere = f.read(1)
while carattere != "":
    # Analizza il carattere corrente
    carattere = f.read(1)
f.close()
```

Esempio: Contare il numero di volte che ciascuna lettera dell'alfabeto inglese compare in un testo letto da file

```
n = [0]*26
f = open("input.txt", "r")
c = f.read(1)
while c != "":
    c = c.upper()
    if c >= "A" and c <= "Z":
        cod = ord(c) - ord("A")
        n[cod] = n[cod] + 1
    c = f.read(1)
f.close()
print(n)
```

Scrittura su file

```
f = open("input.txt", "w")
```

- senza carattere di escape '\n'

```
f.write("Hello world!")
```

- con carattere di escape '\n'

```
f.write("Hello world!\n")
```

- senza carattere di escape '\n'

```
print("Hello world!", end="", file=f)
```

- con carattere di escape '\n'

```
print("Hello world!", file=f)
```

```
f.close()
```