

```
#include <stdlib.h>
#include <string.h>
#include <ctype.h>

#define MAXPAROLA 30
#define MAXRIGA 80

int main(int argc, char *argv[])
{
    int freq[MAXPAROLA]; /* vettore di contatori
delle frequenze delle lunghezze delle parole */
    char riga[MAXRIGA];
    int i, inizio, lunghezza;
    FILE *f;

    for(i=0; i<MAXPAROLA; i++)
        freq[i]=0;

    if(argc != 2)
    {
        fprintf(stderr, "ERRORE: serve un parametro con il nome del file\n");
        exit(1);
    }
    f = fopen(argv[1], "r");
    if(f==NULL)
    {
        fprintf(stderr, "ERRORE: impossibile aprire il file %s\n", argv[1]);
        exit(1);
    }

    while( fgets( riga, MAXRIGA, f ) != NULL )
```

## L'ambiente UNIX/Linux

# Comandi UNIX & Linux (Parte B)

Stefano Quer

Dipartimento di Automatica e Informatica  
Politecnico di Torino

## Gestione archivi

- ❖ L'archiviazione e la compressione di dati può essere gestita mediante il comando **tar**
  - tar = an archiving utility
  - Archiviazione e compressione del direttorio dir nel file file di nome file e estensione tgz

```
tar -czvf <file>.tgz <dir>
```

- Estrazione del contenuto dell'archivio file

```
tar -xzvf <file>.tgz <dir>
```

## Gestione archivi

### ❖ Opzioni utili

Opzioni			
Formato		Significato	Effetto
Compatto	Esteso		
-c			Crea l'archivio
-x			Estra l'archivio
-z, -j, -J			Comprime (gzip, bzip2, 7z)
-f			Specifica il nome dell'archivio
-v			Verbose (stampa i messaggi)

## Gestione archivi

### ❖ Comandi alternativi

- gzip, gunzip
- zip, unzip
- rar, unrar
- compress

## Occupazione spazio su disco

```
df [opzioni] disco
```

❖ Per controllare l'occupazione dei dischi è possibile utilizzare il comando `df`

File system disk  
space usage

### ➤ Opzioni

- `--block-size=x, -B x`
  - `x` scala la dimensione del file del fattore `x`; `x` è una unità opzionale, e.g., 1K, 10K, 1M, 1G, 1T, etc.
    - `-BM` stampa il risultato in unità di 1048576 byte (1MB)
  - `-k` corrisponde a `--block-size=1k`
  - Occupazione in kB

## Esempio

Av.=Available,  
Cap.=Capacity,  
Mnt=Mount point

➤ df

```
$ Filesystem      1K-blocks    Used Available  Use% Mounted on
udev              8183252         0   8183252    0% /dev
tmpfs             1642600     9248   1633352    1% /run
/dev/sda1        49808620 14095784  33159648   30% /
tmpfs             8212992       220   8212772    1% /dev/shm
tmpfs              5120          4     5116     1% /run/lock
tmpfs             8212992         0   8212992    0% /sys/fs/
F_DRIVE          600948732 260043768 340904964   44% /media/D
G_DRIVE          976760828 897641752  79119076   92% /media/G
tmpfs             1642600        44   1642556    1% /run/user/
```

## Occupazione spazio su disco

```
du [opzioni] direttorio
```

- ❖ Per ottenere lo spazio occupato da una directory e tutte le sue sottodirectory è possibile utilizzare il comando du

- **Opzioni**

- --all, -a
  - Occupazione di ciascun file
- --summarize, -s
  - Solo il totale complessivo
- --block-size=1K, -k
  - Occupazione in kB

Estimante file  
space usage  
(disk usage)

## Esempio

Spazio occupato dai  
vari file

```
➤ du
3516  ./graphISOMORPHISM/graphMulti-1.0/trimbleExp
180   ./graphISOMORPHISM/graphMulti-1.0/trimbleStQ
108   ./graphISOMORPHISM/graphMulti-1.0/trimbleStatic
32    ./graphISOMORPHISM/graphMulti-1.0/prova
3840  ./graphISOMORPHISM/graphMulti-1.0
56    ./graphISOMORPHISM/graph-3.0/Yara
64    ./graphISOMORPHISM/graph-3.0/exp
152   ./graphISOMORPHISM/graph-3.0/src
56    ./graphISOMORPHISM/graph-3.0/expYara
332   ./graphISOMORPHISM/graph-3.0
...
1365856 .
```



## Spell checker

```
aspell [opzioni] -c file
```

- ❖ Check sullo spelling dei vocaboli con successiva lista di suggerimenti
  - **Aspell = Interactive spell checker**
  - **Opzioni**
    - `--check name, -c name`
      - Spell check il file di nome name
    - `--master=name, -d name`
      - Dizionario da utilizzare (it=italiano, en=inglese, etc.)

## Spell checker

### ➤ Esempi

- `aspell -c <nomeFile>`
- `aspell -d en -c <nomeFile>`
- `aspell -d it -c <nomeFile>`