



informatica open source

Tutorial n.5
Linguaggio C/C++
Funzioni che ritorna una Struct
Puntatori in C++
I files di testo esempi e casi d'uso

Sito Web <https://www.informaticaopensource.it>

Che cos'è un file

- Un insieme di informazioni memorizzati in un supporto di memoria di massa (ad es. Disco Rigido)
- In ogni Sistema Operativo i files sono identificati da un nome e da un'estensione
- Il nome ad esempio in Windows è CASE Insensitive in Linux/Mac OS no
- L'estensione generalmente indica l'applicazione che ha generato il file non è obbligatoria ma è consigliabile

Argomenti

- I puntatori in C++
- Array con i Puntatori in C++
- File di testo in C++
- File di testo con struct in C++
- File binari in C++

Tipologia di File

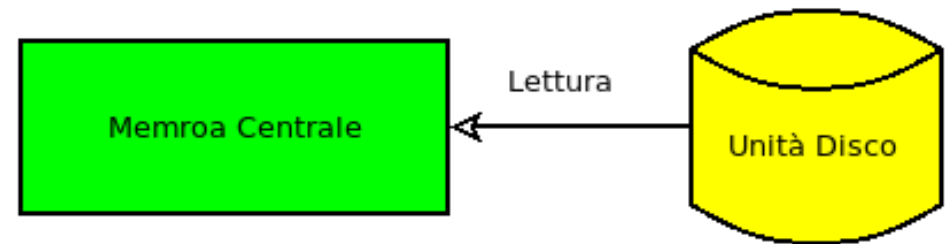
- In tutti i linguaggi di programmazione i files si possono suddividere in:
 - File di testo
 - senza formato (le informazioni sono solo testuali)
 - Con formato (le informazioni sono da interpretare)
 - File binari
 - File costituiti da sequenza di byte
 - La lettura e la scrittura prevede sempre l'interpretazione attraverso il programma ne fa uso

Modalità di Accesso

- Un'altra distinzione forte è la differenza nella modalità di accesso che può essere:
 - Sequenziale (i dati sono letti e scritti in modo consecutivo)
 - Casuale e/o Diretto (la lettura e la scrittura avviene mediante il posizionamento all'indirizzo desiderato)
 - Relative o Calcolato (l'indirizzo è calcolato sulla base di un criterio relativo alle informazioni presenti nel file)

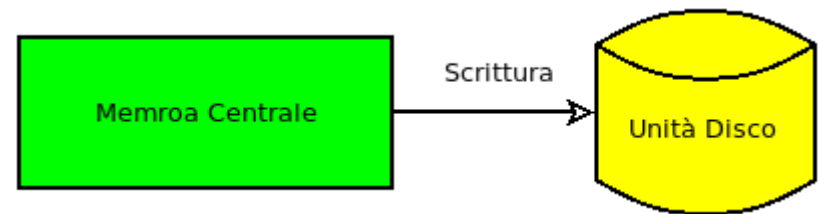
Scrittura e Lettura

La lettura di un file prevede il trasferimento dalla memoria di massa alla memoria centrale

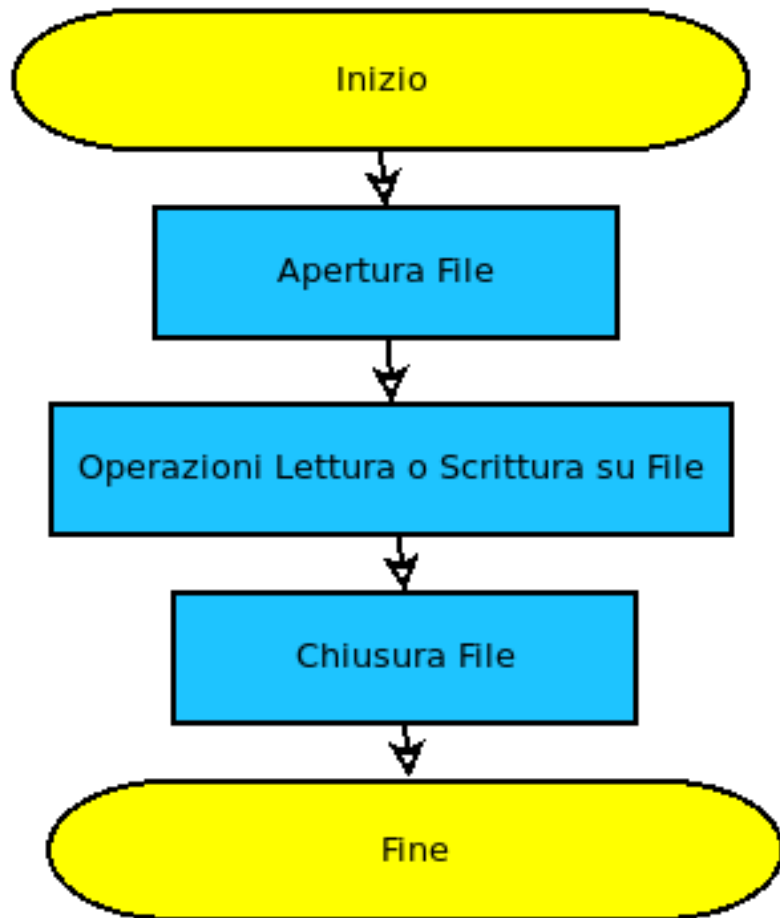


Scrittura e Lettura

La scrittura di un file prevede il trasferimento dalla memoria centrale alla memoria di massa



Sequenza Operativa per i file



- Apertura file collegare il file fisico (presente sul supporto) al descrittore di file nel programma (identificativo file logico)
- Effettuare mediante dei cicli di lettura o scrittura le operazioni
- Chiusura del file e de-allocazione risorsa

Come operare nel C++

- Occorre utilizzare la libreria `fstream`
- I file come descrittori possono essere solo di input solo di output o entrambi con le seguenti dichiarazioni:
 `ifstream nome_descrittore_file;`
 `ofstream nome_descrittore_file;`
 `fstream nome_descrittore;`

Lettura e Scrittura

- La scrittura avviene con la sintassi
`descrittore_file << varibile1<<...<<variailen;`
- E' opportuno separare i dati da separatori come
“\n” se sono dati con formato
- La lettura avviene con l'istruzione
`descrittore_file>>variabile1>>...>>variabilen`

Apertura del file in C++

- `nome_file.open(nome_file_fisico,ios:accesso);`
ios:in sola lettura
ios:out sola scrittura
ios:app aggiunta
ios:binary apertura del file in binario
- E' possibile combinare più flag con l'operatore
"|"
- Ad esempio `nome_file.open("dati.txt",ios:out);`

Caso di studio

```
#include <iostream>
#include <fstream>
#include <string>
using namespace std;
fstream fp;
int n,k;
int age;
string cognome, nome;
int main()
{
    fp.open("dati.txt",ios::out);
    cout<< "Quanti alunni\n"<<endl;
    cin >> n;
    for (k=0;k<n;k++)
    { cout <<"\n Inserisci nome alunno:";
      getchar();
      getline(cin,nome);
      cout << "\n Inserisci cognome alunno:";
      getline(cin,cognome);
      cout << "\n Inserisci l'età:";
      cin >> age;
      fp << nome << "\n" << cognome << "\n" << age;}
    fp.close();
    fp.open("dati.txt",ios::in);
    cout << "Dati alunni:\n";
    while (!fp.eof())
    {
        fp >> nome >> cognome >> age;
        cout << "\nNome Alunno:"<<nome;
        cout << "\nCognome Alunno:"<<cognome;
        cout << "\nEtà alunno:"<<age;
    }
    cout << "\nGrazie per aver utilizzato il programma\n";
    return 0;
}
```

Questo esercizio chiede all'utente di inserire dei dati all'interno di un file di testo. Una volta inseriti i dati occorre ristamparli. Nell'esempio posto è essenziale l'uso del separatore “\n” per appunto distinguere all'interno del file i dati.

Commenti

- E' utilizzata la funzione `getline()` per prevedere l'input di stringhe con spazi
- Il ciclo di lettura deve essere sempre prevedere una struttura del tipo
`leggi_dati_da_file`
`mentre(non finito il file)`
`leggi_dati_da_file`
per evitare che il ciclo un loop infinito vale in generale

Fine Video Lezione

Grazie per l'attenzione

Team informatica Opensource

Sito web <https://www.informaticaopensource.it>