

Esempio di utilizzo dei **File di testo** (come nelle Selezioni Territoriali OII)

Sommare una serie di numeri interi predisposti in un **file** di input che contiene come primo intero un valore che indica quanti sono i numeri da sommare (N) e nelle successive N righe gli N numeri da sommare.

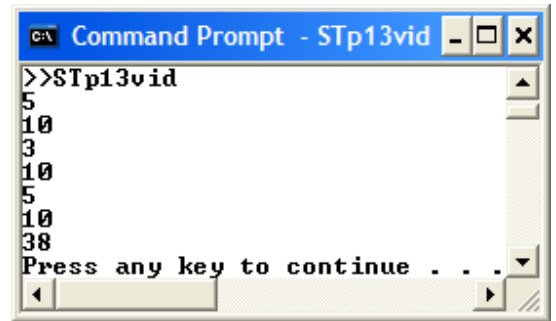
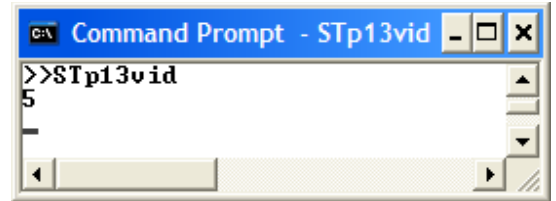
Il programma analogo (STp13vid.cpp) che **acquisisce i dati da tastiera** ed **espone a video il risultato**, senza messaggi di richiesta dei dati di input, è il seguente:

```
//--programma STp13vid.cpp--
#include <iostream>
using namespace std;
main() {
    int N, i, temp, somma = 0;
    cin >> N;
    for (i=0; i< N ; i++)
    {
        cin >> temp;
        somma += temp;
    }
    cout << somma << endl;
    system("pause");
}
```

Letto N il programma resta in attesa degli N valori da sommare (in questo caso di prova digitiamo 5)

digitati i 5 valori (10, 3, 10, 5, 10)

il programma espone di seguito il risultato (38) e termina.



Il programma STp13.cpp con uso di **FILE di testo** differisce di poco dal precedente:

```
//--- programma STp13.cpp ---
#include <iostream>
using namespace std;
#include <fstream>
main() {
    int N, i, temp, somma = 0;
    ifstream in("STp13in.txt");
    ofstream out("STp13out.txt");
    in >> N;
    for (i=0; i< N ; i++)
    {
        in >> temp;
        somma += temp;
    }
    out << somma << endl;
    system("pause");
}
```

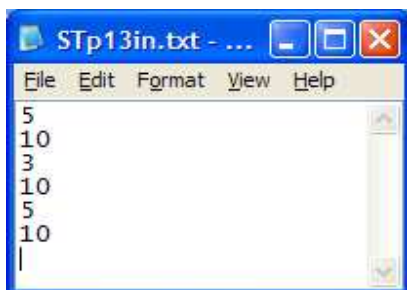
va incluso il modulo **<fstream>** altrimenti il compilatore non riconosce le istruzioni **ifstream** e **ofstream** che consentono di **dichiarare** i nomi **logici usati** all'interno del programma per il file di input (**in**) e per quello di output (**out**) e di **abbinare** i nomi logici (**in** e **out**) ai **nomi fisici** dei file: "STp13**in**.txt" e "STp13**out**.txt"rispettivamente (i file di testo di input vanno creati con Blocco Note in Windows, quelli di output, creati dai programmi, possono essere visualizzati con Blocco Note)

- ifstream in**("STp13**in**.txt"); ← dichiarazione del file di **input in** (istruzione **ifstream**) e **apertura in lettura** (implicita) del file "STp13**in**.txt"
- ofstream out**("STp13**out**.txt"); ← dichiarazione del file di **output out** (istruzione **ofstream**) e **apertura in scrittura** (implicita) del file "STp13**out**.txt"
- in >> N;** ← lettura dal file **in** del primo numero intero (nella variabile N)
- in >> temp;** ← ciclo per leggere nella variabile temp, di volta in volta, gli N numeri che seguono N nel file **in** e aggiungerli in somma
- out << somma << endl;** ← scrittura del risultato finale (somma) sul file **out**

NOTA: N indica quanti numeri andranno letti nella variabile **temp** (dal file di input) e sommati in **somma**

Salvato e compilato il programma con Dev-C++ (nome del File Sorgente: STp13.cpp), con **Blocco Note** va preparato il file di prova digitando sulla prima riga il valore di N (in questo caso il numero 5) e sulle 5 righe successive i 5 valori da sommare (es: 10, 3, 10, 5, 10).

NOTA: è opportuno usare il tasto invio dopo ogni numero, come se fornissimo i dati di input da tastiera.

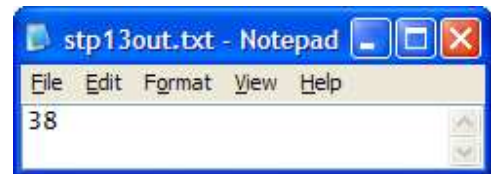
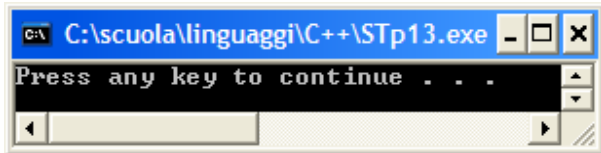


Il file di input va salvato indicando:
 - un **nome** (in questo esempio *STp13in*)
 - e il **tipo di file** (*Documento di Testo (.txt)*)

Attenzione:
 l'estensione **.txt** viene aggiunta da BloccoNote.

Mandando in esecuzione il programma, l'istruzione `system("pause");` presenta la videata con il solo messaggio <<Premere un tasto per continuare . . .>> (*Press any key to continue* in inglese).

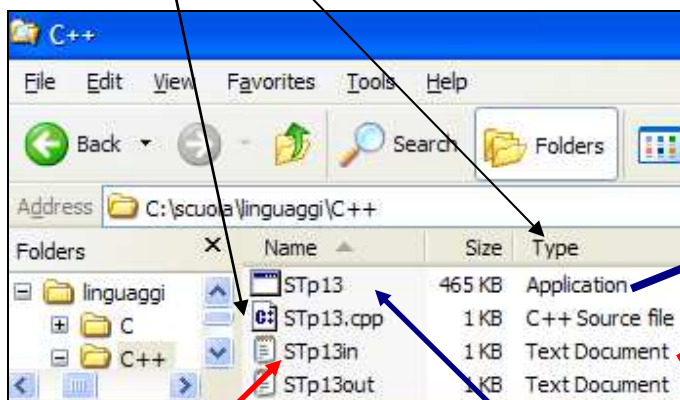
Il risultato dell'elaborazione va controllato con Blocco Note (*Notepad* in inglese) aprendo il file *STp13out.txt* creato dal programma STp13.cpp.



ATTENZIONE!!!

Se l'installazione di Windows è configurata per **NON mostrare le estensioni dei file conosciuti** sia i file di testo (.txt) che i file eseguibili (.exe) sono esposti solo con il **nome**, anche se ne viene indicato correttamente il tipo (*Type* in inglese).

file di testo contenente un sorgente C++

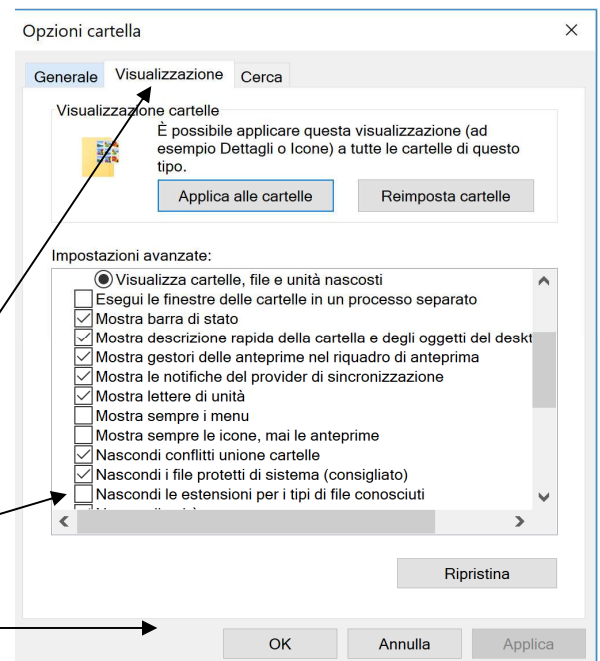


file di testo

file eseguibile

in una configurazione che mostra tutti i file con la propria estensione si vedrebbe, invece:

Name	Size	Type
STp13.cpp	1 KB	C++ Source
STp13.exe	465 KB	Application
STp13in.txt	1 KB	Text Document
STp13out.txt	1 KB	Text Document



Per **poter vedere le estensioni di tutti i file** presenti nelle cartelle, da:

- Sistema Windows > Esplora File
- dalla scheda File,

selezionare

- Modifica opzioni cartelle e ricerca (o Opzioni)
- dalla scheda Visualizzazione,
- eliminare il segno di spunta nella casella "Nascondi le estensioni per i tipi di file conosciuti",
- premere il tasto *Applica* (*Apply*).

utilizzo dei **File di testo** con **Ripetizione Pre-condizionale (while)**
e controllo errori nelle operazioni sui file - metodi: fail() e eof()

Lo stesso problema può essere risolto senza che si registri sul file di input il numero N di valori da leggere successivamente (cioè senza sapere quanti numeri sono registrati sul file di input e quindi leggendo numeri fino al raggiungimento della fine del file – *end of file* – *eof*).

Il programma (STp13eof.cpp) diventa in questo caso:

```
//--- programma STp13eof.cpp ---
```

```
#include <iostream>
using namespace std;
#include <fstream>
int main(){
    fstream in;
    fstream out;
    int temp, somma = 0;
```

In questo esempio si usa l'*apertura* e *chiusura esplicita* dei file: metodi **open()** e **close()**.
 NOTA: i file vengono comunque chiusi a fine programma (chiusura implicita)

dichiarazione del file di **input** (istruzione **fstream**)
 dichiarazione del file di **output** (istruzione **fstream**)

```
    in.open("stp13in.txt", ios::in);
    if ( in.fail() ){
        cout<< "\nATTENZIONE! il file non esiste\n";
        return 100;
    }
    in >> temp;
    while ( ! in.eof() ) {
        somma += temp;
        in >> temp;
    }
    in.close();

    out.open("stp13ou.txt", ios::out);
    out << somma << endl;
    out.close();

    cout<< "\nelaborazione completata\n";
}
```

apertura in lettura (parametro ios::**in**) del file "STp13**in**.txt"
controllo esito apertura (metodo **in.fail()**)
 se il file non esiste il programma segnala l'errore e termina

lettura dal file **in** del primo numero (nella variabile temp)

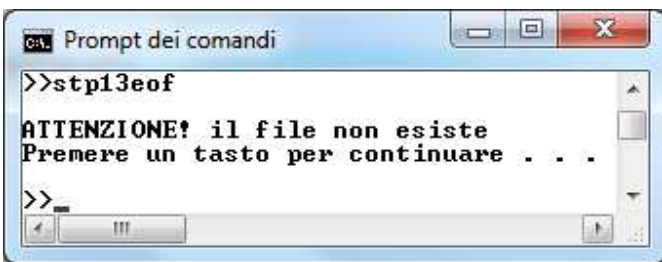
Mentre NON è raggiunta la fine del file **in** il ciclo aggiunge in **somma** il numero letto in **temp** e tenta una lettura successiva

chiusura esplicita del file **in**

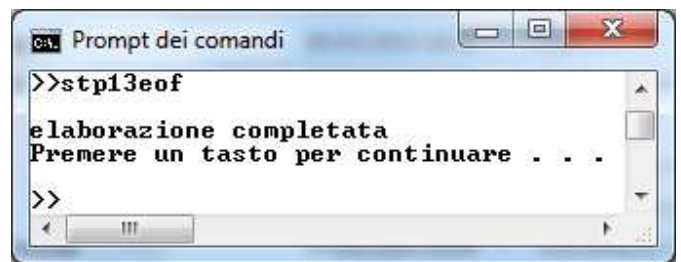
apertura in scrittura (parametro ios::**out**) del file "STp13**ou**.txt"

scrittura del risultato finale (somma) sul file **out**
 chiusura esplicita del file **out**

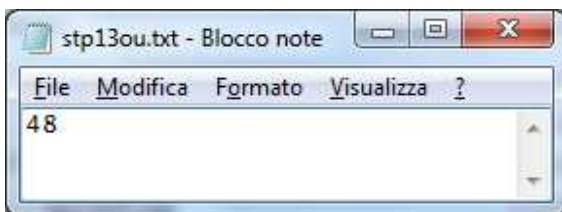
Risposta a video se il file di input NON esiste



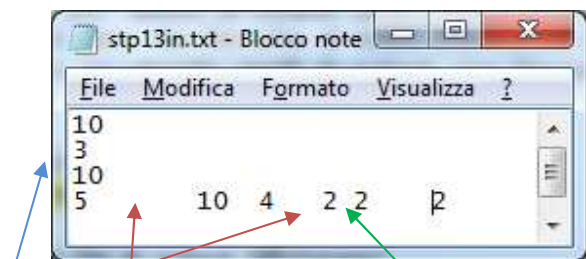
Risposta a video se il file di input è pronto



Contenuto del file di **output** dopo l'esecuzione



Contenuto del file di **input**



NOTA: nel file di input i dati devono essere separati da **invio**, da **tabulazione** o da **uno spazio**