Conversione stringhe/numeri in C++

Alessandro Bugatti

22 gennaio 2009

Sommario

Esistono molti modi per convertire delle stringhe in numeri e viceversa, sia in C che in C++. Ognuno di questi modi presenta i suoi pro e i suoi contro, quello proposto in questo articolo utilizza le classi sstream per fare la conversione e ha il vantaggio, rispetto ai metodi standard del C, che non pone limiti alle dimensioni delle cose da convertire, appoggiandosi su classi che gestiscono la memoria internamente. Oltre alla conversione in questo articolo verrà poi spiegato come passare delle stringhe stile C++ (classe string) a delle funzioni che si aspettano come parametri delle stringhe stile C (puntatori a carattere).

La classe stringstream

La classe *stringstream* ha lo scopo di fornire ai programmatori un'interfaccia comune ai normali stream di I/O, con due sottoclassi specializzate per l'input (*istringstream*) e l'output (*ostringstream*). Per utilizzarle è necessario includere l'header standard *sstream* nel modo seguente:

```
#include <sstream>

using namespace std;

int main()
{
        ostringstream out;
        istringstream in;
        ....
}
```

Come vedemo poi negli esempi seguenti il modo di utilizzarle è del tutto simile a quanto si fa con i flussi predefiniti *cin* e *cout*.

Conversione da stringa a numero

La necessità di convertire una stringa in un numero (sia esso intero o floating point) è spesso sentita nei programmi dotati di una GUI, nei quali l'input dell'utente è sempre disponibile sotto forma di stringa. L'esempio che segue mostra la conversione di una stringa C++, che si suppone contenere un intero (per brevità non solo inclusi gli header).

```
int main()
{
        istringstream in;
        string s = "123";
        int numero;
        //Inserisco la stringa nel flusso di input
        in.str(s);
        //Effettuo la conversione
        in >> numero;
        cout << numero; //stampa 123
        return 0;
}
```

Lo stesso meccanismo può essere utilizzato nel caso dovessimo convertire una stringa stile C, sfruttando il costruttore di default che convertirà la stringa C in un oggetto di classe string.

```
int main()
{
        istringstream in;
        char s[] = "123";
        int numero;
        //Inserisco la stringa nel flusso di input
        in.str(s);
        //Effettuo la conversione
        in >> numero;
        cout << numero; //stampa 123
        return 0;
}
```

Ovviamente nel caso s non contenga una stringa che rappresenta un numero la conversione non sarà possibile. Per gli altri tipi di dati numerici (float e double) il procedimento è esattamente il medesimo.

Conversione da numero a stringa

Sempre nel contesto di un'applicazione GUI una volta calcolato il valore numerico di una qualche espressione c'è l'esigenza di mostrarlo a video e anche in questo caso i metodi per farlo richiedono di trasformarlo in una stringa (stile C o C++ a seconda della libreria). In questo caso viene usata la classe *ostringstream* nel seguente modo.

```
int main()
{
        ostringstream out;
        int numero = 123;
        string s;
        //Inserisco il numero nel flusso di output
        out << numero;
        //La assegno a una stringa
        s = out.str();
        cout << s; //stampa 123
```

```
        return 0;
}
```

Se ci fosse bisogno di convertirla in una stringa C basterebbe utilizzare il metodo c_str() della classe string.

```
void funzioneQualunque(const char *s);

int main()
{
        ostringstream out;
        int numero = 123;
        string s;
        //Inserisco il numero nel flusso di output
        out << numero;
        //La passo a funzioneQualunque come
        //vettore di caratteri
        funzioneQualunque(out.str().c_str());
        return 0;
}
```