

Nota che il programma potrebbe anche essere ridotto a una selezione a una sola via, cioè senza la clausola `else`, scrivendo dopo la riga 8 le due istruzioni:

```
if (numero<0) { numero = -numero; }
cout<<"\n il valore assoluto del numero e' "<<numero;
```

Questo codice funzionerebbe altrettanto bene, però modificherebbe il valore della variabile `numero` se l'utente ha inserito un dato negativo; perciò il dato di input originario non sarebbe più disponibile, nel caso si volessero aggiungere ulteriori elaborazioni.

Molti algoritmi richiedono l'uso di più **selezioni annidate** una nell'altra.

Scarica per esempio dal sito web del libro il codice `multiplo.cpp` della **figura 1.3** e testane il funzionamento.

Questo programma chiede di inserire due numeri interi positivi e poi controlla se il maggiore tra i due numeri sia o meno multiplo del minore.

Per il controllo è ideale usare l'operatore aritmetico modulo `%`, che restituisce 0 quando la divisione di un intero per un altro non dà resto.

Bisogna tuttavia prestare attenzione al fatto che l'operatore funziona correttamente soltanto quando i due numeri interi sono entrambi positivi, e inoltre non sappiamo quale sarà il maggiore tra i due dati di input inseriti dall'utente.

```

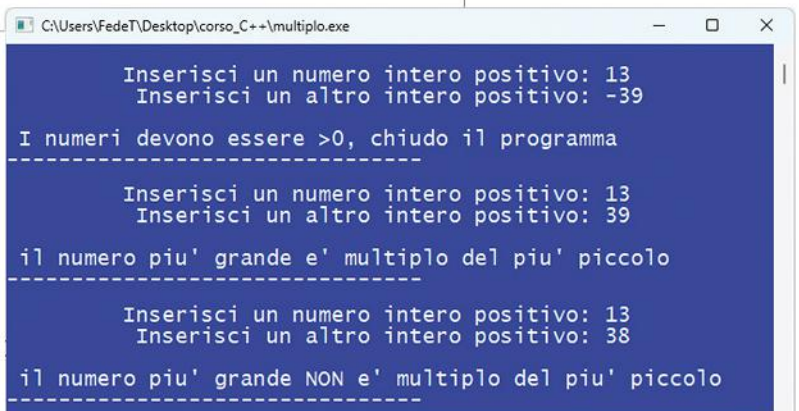
multiplo.cpp
1  #include <iostream>
2  using namespace std;
3
4  int main(void)
5  {
6      int numA, numB;
7      cout<<"\n\t Inserisci un numero intero positivo: ";
8      cin>>numA;
9      cout<<"\t Inserisci un altro intero positivo: ";
10     cin>>numB;
11     // stabiliamo anzitutto che nessuno dei due numeri sia zero o negativo
12     if (numA <= 0 || numB <= 0) {
13         cout<<"\n I numeri devono essere >0, chiudo il programma "; }
14     else { // scopriamo se il numero più grande è multiplo di quello più piccolo
15         if (numA >= numB) {
16             if (numA%numB==0)
17                 {cout<<"\n il numero piu' grande e' multiplo del piu' piccolo";}
18             else
19                 {cout<<"\n il numero piu' grande NON e' multiplo del piu' piccolo";} }
20         else {
21             if (numB%numA==0)
22                 {cout<<"\n il numero piu' grande e' multiplo del piu' piccolo";}
23             else
24                 {cout<<"\n il numero piu' grande NON e' multiplo del piu' piccolo";} } }
25     }
26

```

Figura 1.3
Un programma con strutture di selezione annidate.

condizione composta sui due dati di input

nota come l'**indentazione** dei blocchi di istruzioni aiuti a identificare visivamente il progressivo annidamento delle strutture condizionali



Il codice della **figura 1.3** contiene selezioni annidate una nell'altra su tre livelli:

- ▶ **riga 12:** il primo `if` controlla, con una condizione composta che usa l'operatore logico OR, se uno o l'altro dei numeri di input sia negativo: se è così, il programma invia il messaggio della riga 13 e poi termina;
- ▶ **riga 15:** se entrambi i numeri sono positivi, questo `if` annidato controlla se il primo numero sia maggiore del secondo: se è così, l'ulteriore selezione annidata delle **righe 16-19** determina se il primo numero è multiplo del secondo e stampa a schermo il responso;
- ▶ **riga 20:** il percorso `else` dell'`if` annidato viene seguito se il primo numero è *minore* del secondo: in tal caso l'ulteriore selezione annidata delle **righe 21-24** determina se il secondo numero è multiplo del primo e stampa a schermo il responso.