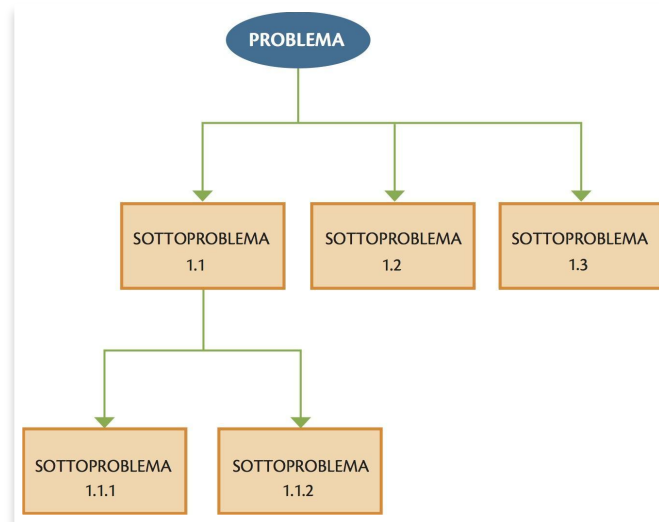


**WEB DEVELOPER
BASI DEL CODING
Massimo PAPA
Primi elementi di programmazione**

Progettare una soluzione

Il **metodo top-down** è la metodologia che permette di affrontare un problema in modo discendente, *dall'alto verso il basso*.



Un'altra metodologia di analisi è offerta dal **metodo bottom-up**, *dal basso verso l'alto*: si risolvono prima i dettagli più caratteristici del problema.

Dichiarazione dei dati in C++

Nella realizzazione dei programmi è importante la progettazione delle strutture dati da usare e la loro memorizzazione in apposite variabili.

Con riferimento al **linguaggio di programmazione C++** illustriamo nella tabella le istruzioni per dichiarare variabili con il loro significato.

Dichiarazione	Tipo	Occupazione
short int i;	numero intero compreso tra -32768 e +32767	2 byte
long int popolazione;	numero intero compreso tra -2.147.483.648 e +2.147.483.648	4 byte
int n;	può coincidere con short int o con long int	2-4 byte
bool risposta;	può assumere valore true (vero) o false (falso)	1 byte
char categoria;	carattere ASCII	1 byte
float area;	numero decimale in precisione singola	4 byte
double distanza;	numero decimale in precisione doppia	8 byte

Le istruzioni di I/O

L'**interfaccia utente** è il mezzo con cui l'utente può comunicare con l'elaboratore.

Tramite le apposite **istruzioni** si forniscono in **input** i dati e si ricevono in **output** i risultati.



Le istruzioni di I/O in C++

L'interfaccia può essere di tipo **testuale**, come nell'ambiente **C++**:
input e output (**I/O**) sono gestiti a livello di «linea» e in modo sequenziale.

Un programma che vuole effettuare operazioni di input/output deve prevedere l'uso di apposite librerie.

È necessario inserire all'inizio del programma le seguenti direttive:

```
#include <iostream>  
using namespace std;
```

In questo modo è possibile usare le istruzioni **cin** e **cout** che permettono rispettivamente di acquisire dati dalla tastiera (console input) e di visualizzare risultati sul video.

Gestire l'input e l'output in C++

Realizziamo per esempio un programma che usa le istruzioni di **output** per scrivere a video il messaggio «Buongiorno».

L'algoritmo



La codifica in C++

```
#include <iostream>
using namespace std;

int main (){
    cout << "Buongiorno" << endl;
    system ("pause");
    return(0);
}
```

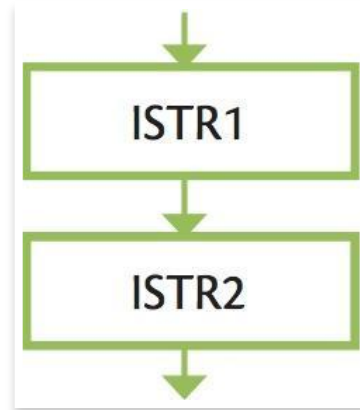
Istruzione di scrittura.

L'esecuzione

```
Buongiorno
Premere un tasto per continuare . . . _
```

La sequenza e l'assegnazione

Il costrutto di **sequenza** rappresenta la situazione in cui si hanno diverse azioni che sono eseguite una dopo l'altra.



L'istruzione tipica che compare negli algoritmi è l'**assegnazione**, che permette di conservare in una variabile un determinato valore.

La sequenza e l'assegnazione: un esempio – problema e analisi

Il problema

Per esempio, dati due numeri, vogliamo calcolare il prodotto.

L'analisi

Per potere effettuare il prodotto di due numeri occorre:

- riceverli in input;
- memorizzarli in variabili opportune;
- dopo avere effettuato la moltiplicazione, fornire il risultato in output.

La sequenza e l'assegnazione: un esempio – le variabili

Le variabili

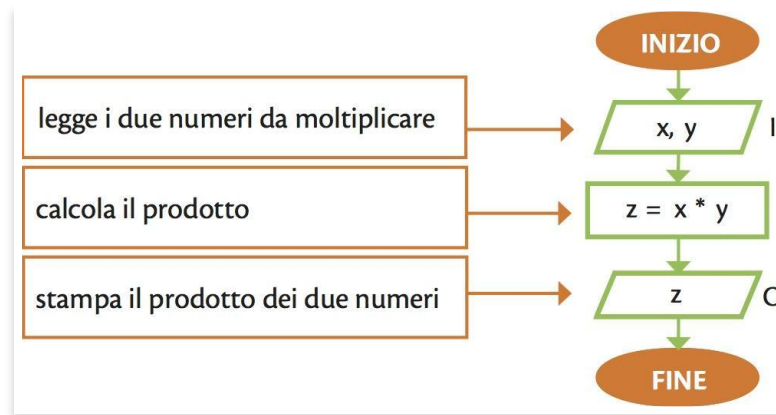
Iniziamo scrivendo le variabili che andremo a usare.

Per ciascuna di esse assegniamo un **nome**, individuiamo il **tipo** di variabile, valutiamo se si tratta di una variabile di input o di output o prodotta dal programma (cioè l'**utilizzo**) e inseriamo una **descrizione**.

Nome	Tipo	Utilizzo	Descrizione
x	intero	input	primo fattore
y	intero	input	secondo fattore
z	intero	output	risultato del prodotto

La sequenza e l'assegnazione: un esempio – l'algoritmo

L'algoritmo



Il test e la simulazione

Possiamo prevedere due casi prova: uno con numeri naturali e uno con numeri interi negativi.

$$x \leftarrow 3 \quad y \leftarrow 5 \quad Z \leftarrow 3 * 5 = 15$$

$$x \leftarrow 2 \quad y \leftarrow -4 \quad Z \leftarrow 2 * (-4) = -8$$

La sequenza e l'assegnazione: un esempio – le codifiche

La codifica in C++

```
#include <iostream>
using namespace std;
int main (){
    int x;
    int y;
    int z;
    cout<<"inserire il primo fattore"<<endl;
    cin>>x;
    cout<<"inserire il secondo fattore"<<endl;
    cin>>y;
    z = x * y;
    cout <<"prodotto risultante = \t"<<z<<endl;
    system ("pause");
    return (0);
}
```

La sequenza e l'assegnazione: un esempio – le esecuzioni

L'esecuzione del programma in C++

```
inserire il primo fattore  
5  
inserire il secondo fattore  
3  
prodotto risultante = 15  
Premere un tasto per continuare . . .
```