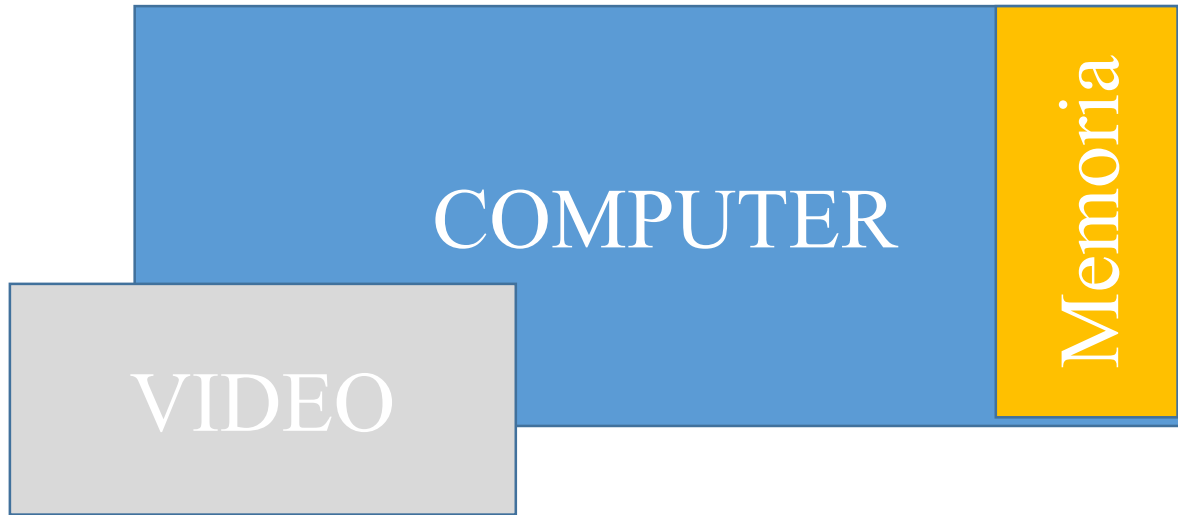


Programmazione

e programmiamo



scriviamo programmi
che il computer
esegue

per il momento non
ci chiediamo come
questo succede,
ma succede

un programma usa variabili:

```
int x, pippo, pluto; // variabili intere
```

possiamo leggere i loro valori da tastiera

```
cin >> x >> pippo >> pluto;
```

se scriviamo da tastiera

10

5

30

memoria

x

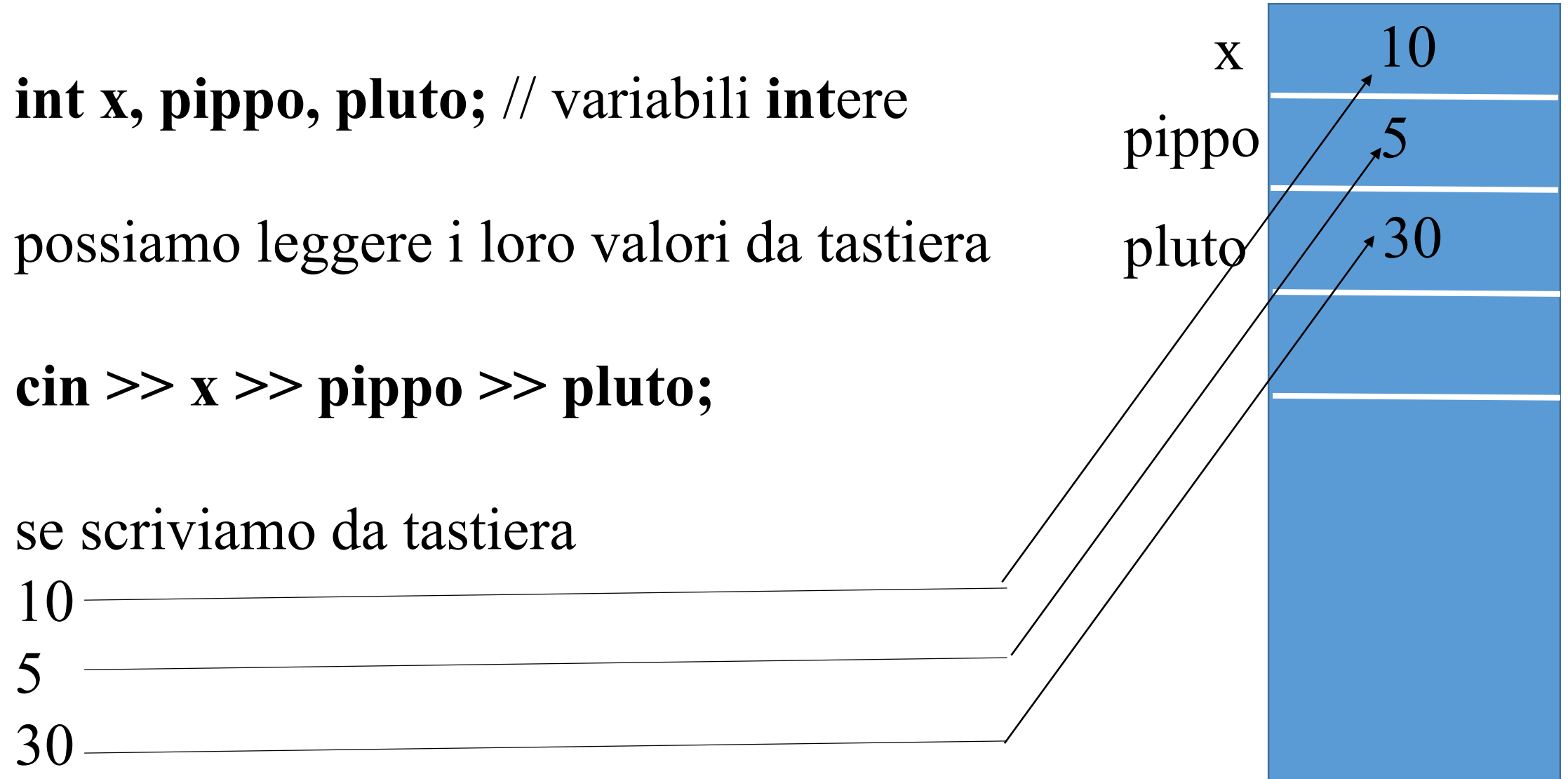
10

pippo

5

pluto

30



vogliamo calcolare il minimo tra x, pippo e pluto:

x < pippo e x < pluto → minimo è x

pippo < x e pippo < pluto → minimo è pippo

pluto < x e pluto < pippo → minimo è pluto

if(x < pippo && x < pluto)

cout << x;

else

if(pippo < x && pippo < pluto)

cout << pippo;

else

if(pluto < x && pluto < pippo)

cout << pluto;

questo è C++
if-else
&&
;
<
<<

facciamo da 2 a 6 confronti !

Possiamo farne di meno !

```
if (x < pippo)  
    if (x < pluto)  
        cout << x;  
    else  
        cout << pluto;  
else  
    if (pippo < pluto)  
        cout << pippo;  
    else  
        cout << pluto;
```

sempre 2 confronti !

abbiamo imparato diverse cose:

- meglio non fare operazioni inutili
- nomi o variabili hanno un posto in memoria in cui c'è il valore
- i linguaggi di programmazione obbediscono a regole:
 - if-then
 - le istruzioni finiscono con ;
 - << e >> per l'input/output per dare valori alle variabili e mostrare il risultato
 - < e > per confrontare valori

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

```
main()
```

```
{ int x,pippo,pluto;
```

```
    cin >> x >> pippo >> pluto;
```

```
    if (x < pippo)
```

```
        if (x < pluto)
```

```
            cout << x;
```

```
        else
```

```
            cout << pluto;
```

```
    else
```

```
        if (pippo < pluto)
```

```
            cout << pippo;
```

```
        else
```

```
            cout << pluto;
```

```
}
```

serve a includere le librerie di i/o

questo è un programma C++

se salviamo quel programma nel file: esercizio.cpp otteniamo un programma sorgente e se scriviamo in un terminale:

```
g++ -o esercizio esercizio.cpp
```

lo traduciamo nel programma oggetto esercizio

che il computer esegue

```
./esercizio // ha bisogno di 3 interi e poi restituisce il minimo
```


vediamo che succede se lo facciamo:

il programma esegue, ma resta sospeso in attesa dei 3 interi
è brutto !

```
cout << « scrivi 3 interi\n»
```

```
cin >> x >> pippo >> pluto;
```

se volessimo trovare il minimo di 4, 5, 6 ecc. variabili?

ci serve un modo di dichiarare molte variabili in un colpo solo: gli array

int A[100];

100 variabili: A[0], A[1], A[2], ..., A[99]

possiamo leggere 100 interi senza scrivere cin >> A[0] >> A[1] >> 100 volte ?

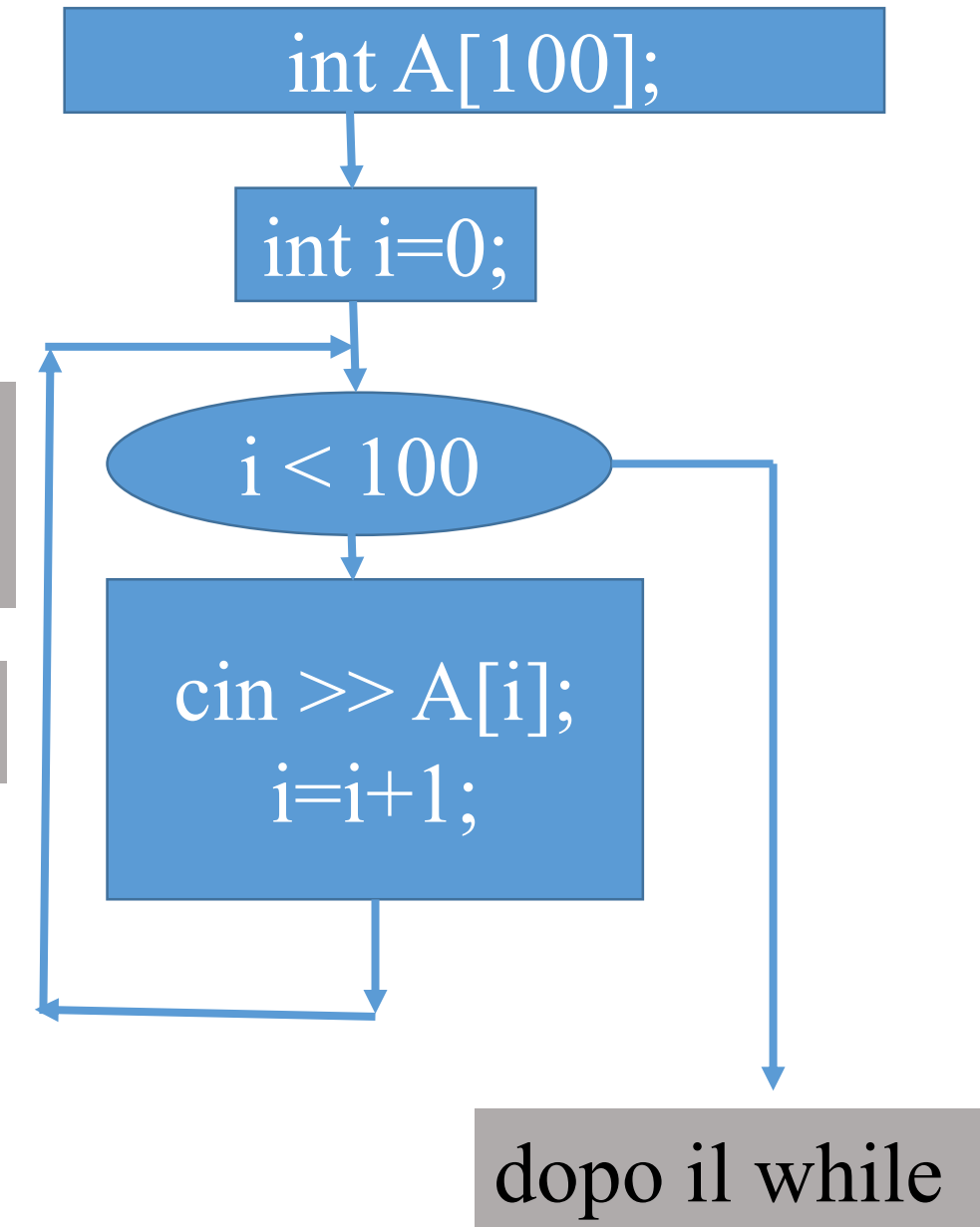
è facile:

```
int A[100], i=0;
```

```
while (i < 100)  
{  
    cin >> A[i];  
    i= i+1;  
}
```

condizione di
permanenza

corpo



sarà anche facile trovare il minimo dei 100 valori letti:

```
int min =A[0], i=1;  
while(i <100)  
{  
    if (A[i] < min)  
        min = A[i];  
    i=i+1;  
}  
  
cout << min;
```