

Informatica 1

Corso di Laurea Triennale in Matematica

Gianluca Rossi

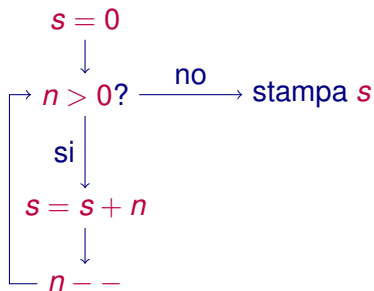
`gianluca.rossi@uniroma2.it`

Dipartimento di Matematica
Università di Roma "Tor Vergata"

4: Controllo del Flusso



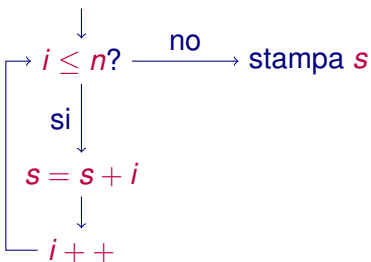
$$s = \sum_{i=0}^n i$$



Flusso del codice e decisioni (2)

$$s = \sum_{i=0}^n i$$

$s = 0; i = 0$



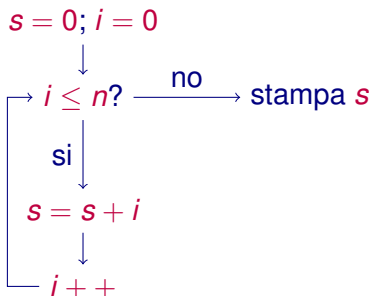
Istruzione **while**

while(*condizione*) *istruzione*

Esegue *istruzione* fino a quando *condizione* è vera (*condizione* \neq 0).
Altrimenti passa all'istruzione successiva.



Istruzioni per il controllo del flusso: **while** ... (2)

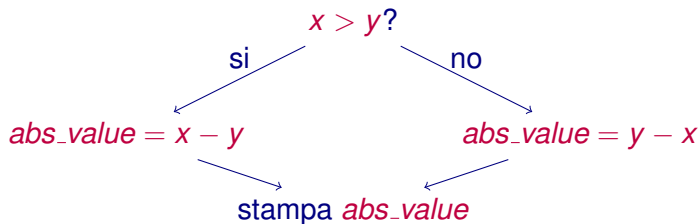


```
main(){  
    int n,i,s;  
  
    n=10;  
    s = 0; i = 0;  
  
    while(i <= n){  
        s = s + i;  
        i++;  
    }  
}
```



Istruzioni per il controllo del flusso: **if ... else ...**

Calcolo di $|x - y|$:



Istruzioni per il controllo del flusso: **if ... else ...** (2)

Istruzione **if ... else**

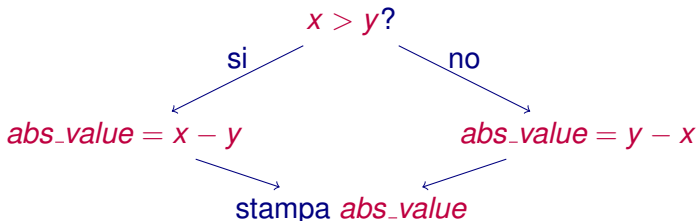
if(*condizione*) *istruzione*₁ **else** *istruzione*₂

Viene eseguita *istruzione*₁ se *condizione* è vera. Altrimenti si esegue *istruzione*₂.

Il ramo **else** è facoltativo.



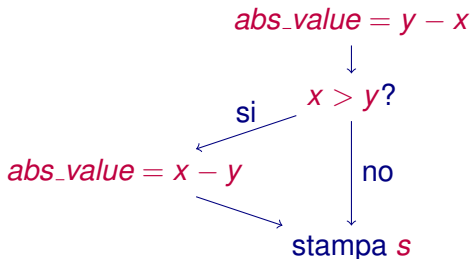
Istruzioni per il controllo del flusso: **if ... else ...** (2)



```
main(){  
    int x,y;  
    unsigned int abs_value;  
  
    if(x > y)  
        abs_value = x - y;  
    else  
        abs_value = y - x;  
}
```



Istruzioni per il controllo del flusso: **if** ...



```
main(){  
    int x,y;  
    int abs_value = y - x;  
  
    if(x > y)  
        abs_value = x - y;  
}
```



if ... else ... è inutile?

if(*condizione*) *istruzione*₁ **else** *istruzione*₂
... **equivale a...**

vero = *condizione*;
falso = 1;

```
while(vero) {  
    istruzione1;  
    vero = 0;  
    falso = 0;  
}
```

```
while(falso) {  
    istruzione2;  
    falso = 0;  
}
```



Istruzioni per il controllo del flusso: **do ... while**

Istruzione **do ... while**

do *istruzione* **while**(*condizione*);

Esegue *istruzione* almeno una volta. Viene rieseguita fino a quando *condizione* è vera (*condizione* \neq 0). Altrimenti passa all'istruzione successiva.



Istruzioni per il controllo del flusso: **for**

Istruzione **for**

for(*inizializzazione*; *condizione*; *iterazione*) *istruzione* ;

Esegue equivale a

```
inizializzazione;  
while(condizione) {  
    istruzione;  
    iterazione;  
}
```



Istruzioni per il controllo del flusso: **for** (2)

$$s = \sum_{i=0}^n i$$

```
main(){  
    int n,i,s;
```

```
    n=10;  
    s=0;
```

```
    for(i = 0; i <= n; i++)  
        s = s + i;  
}
```

oppure

```
main(){  
    int n,i,s;
```

```
    n=10;
```

```
    for(i = 0, s = 0 ; i <= n; i++)  
        s = s + i;  
}
```

