

# Fondamenti di Programmazione (A)

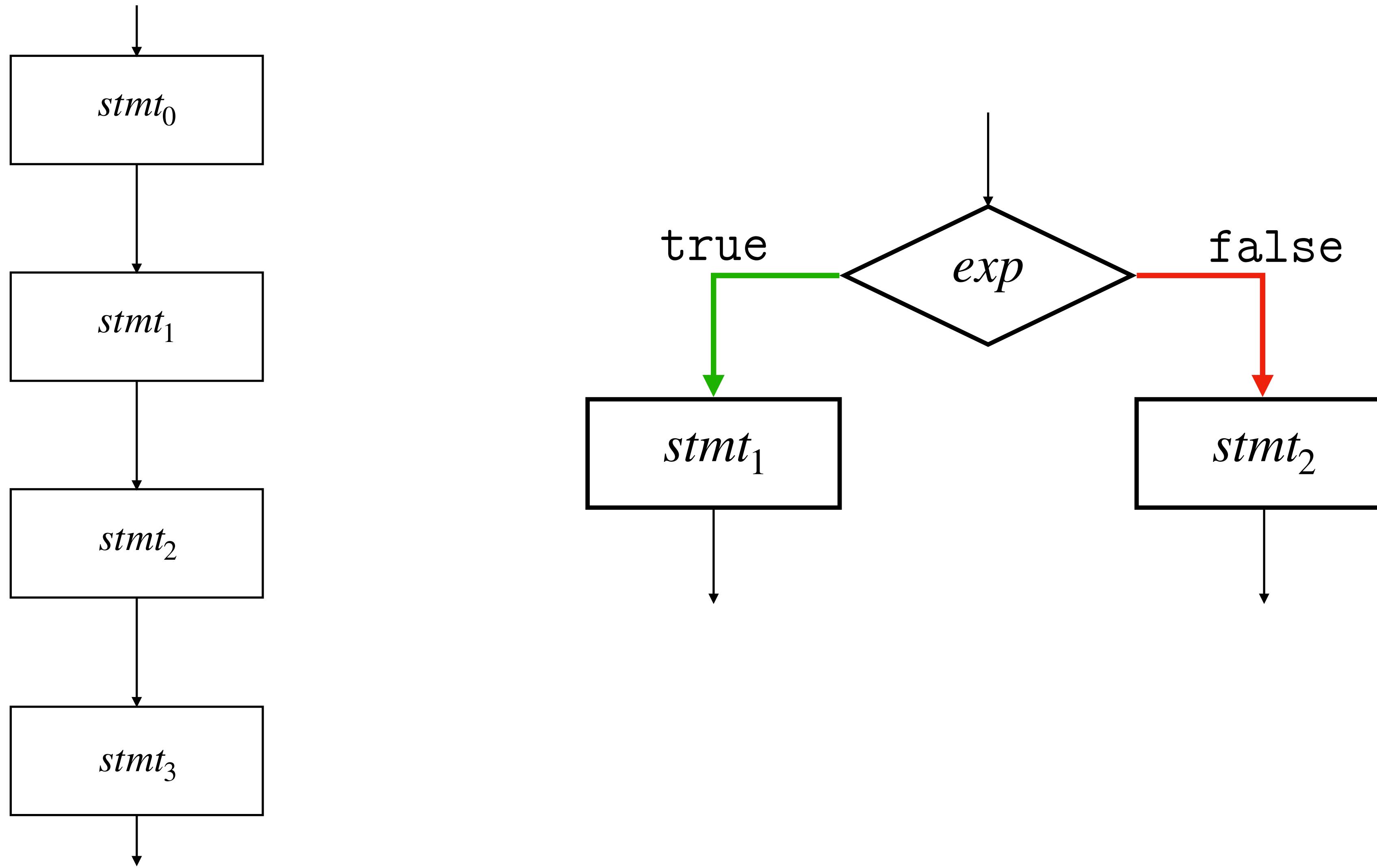
Lecture 10 - Costrutti di controllo del flusso - Ciclo while

Vincenzo Arceri - Università di Parma - [vincenzo.arceri@unipr.it](mailto:vincenzo.arceri@unipr.it)

# Puntate precedenti

- Assegnamenti
- Espressioni
- Statement if

# Programmi sequenziali e branching



# Esempio

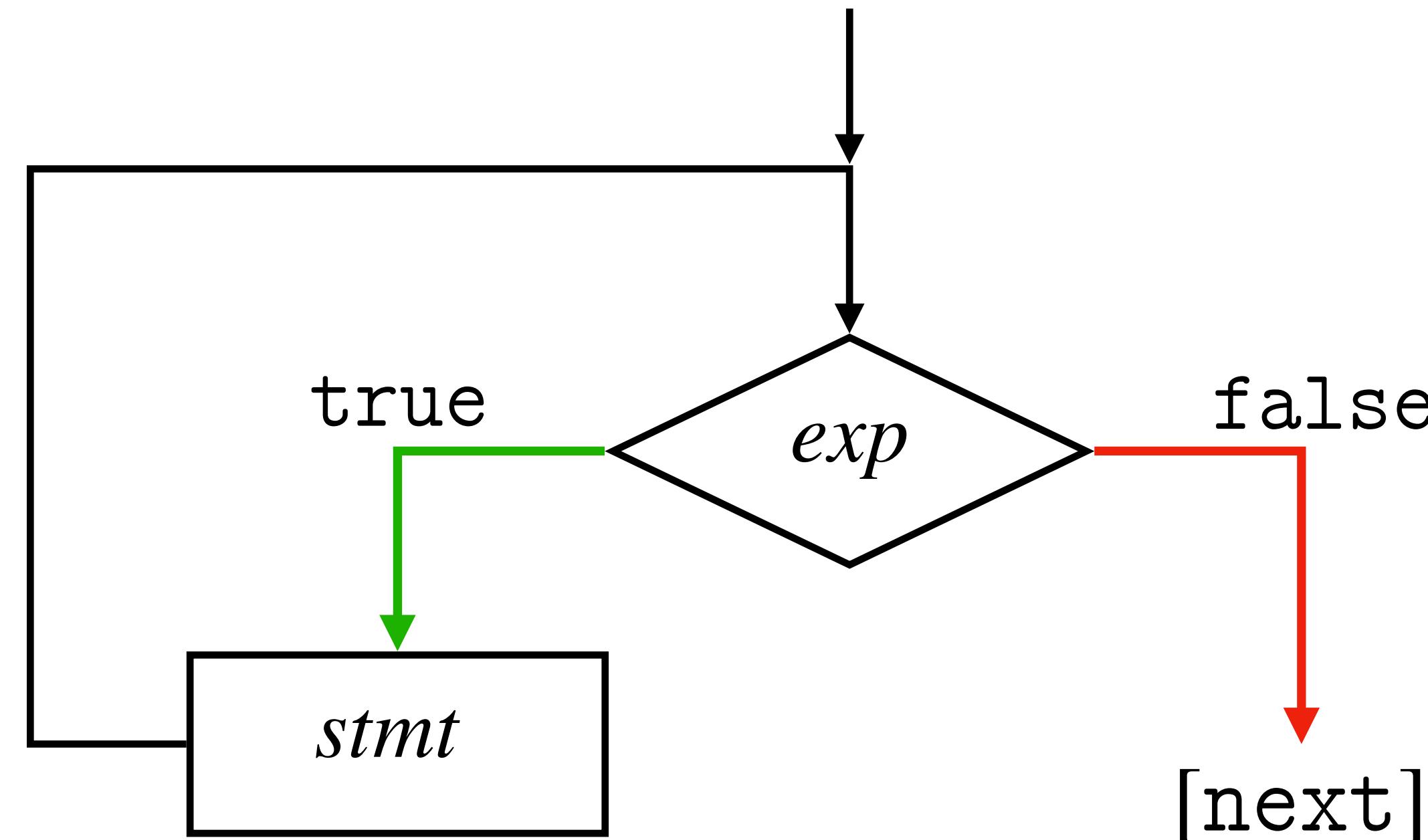
- **Problema:** stampare 10 volte \*\*\*\*

\*\*\*\*  
\*\*\*\*  
\*\*\*\*  
\*\*\*\*  
\*\*\*\*  
\*\*\*\*  
\*\*\*\*  
\*\*\*\*  
\*\*\*\*  
\*\*\*\*

# Statement while

- Informalmente: finché  $exp$  è vera, esegui  $stmt$
- Più precisamente: calcola il valore di  $exp$ , se è vera esegui  $stmt$  e ripeti, altrimenti termina l'esecuzione dello statement while (passa a [next])

```
while (exp)
      stmt
      [next]
```



# Statement while

## Esercizio

- **Problema:** stampare 10 volte \*\*\*\*

```
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****
```

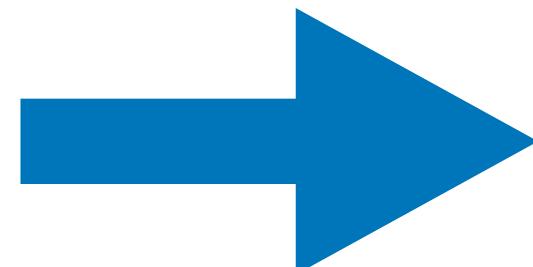
# Statement while

## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

Memoria

Output

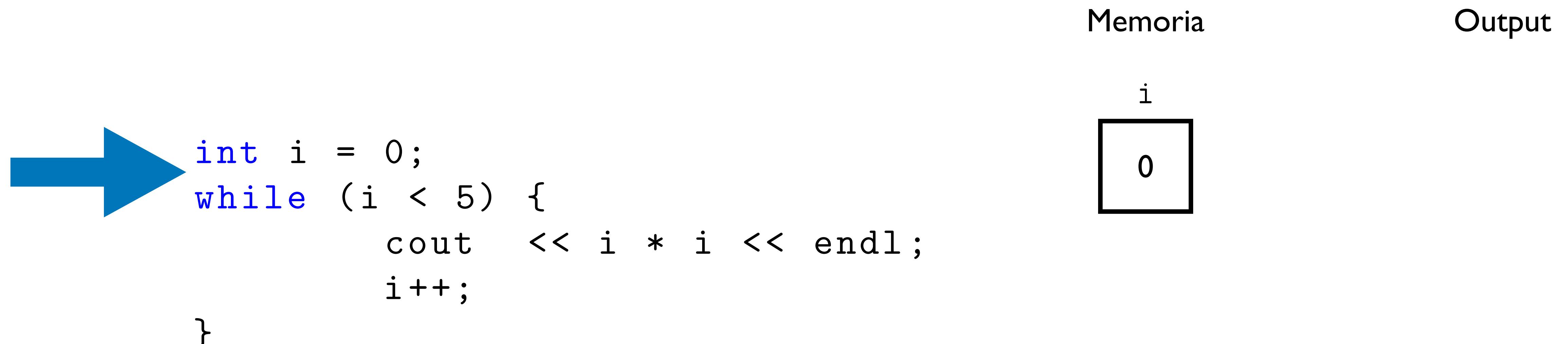


```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```

# Statement while

## Esercizio

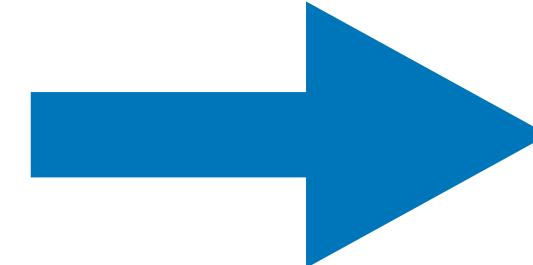
- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



# Statement while

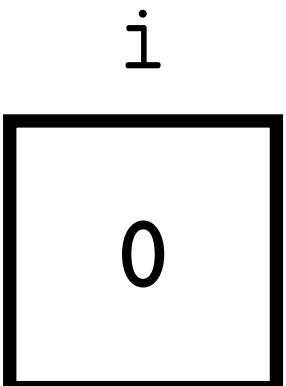
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```

Memoria



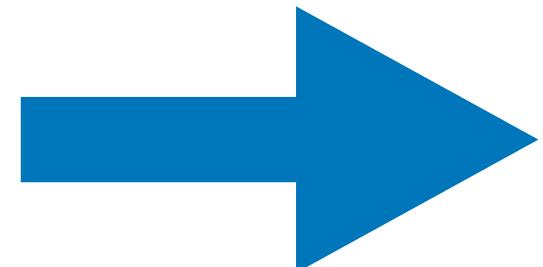
Output

# Statement while

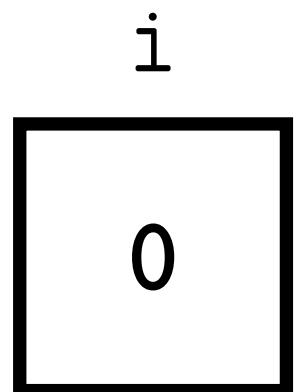
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



Output

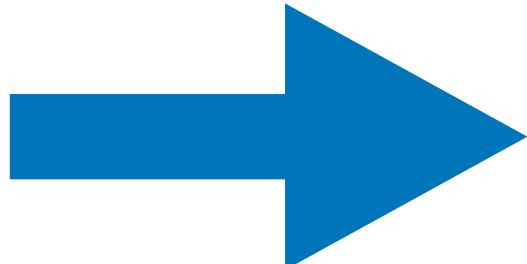
0

# Statement while

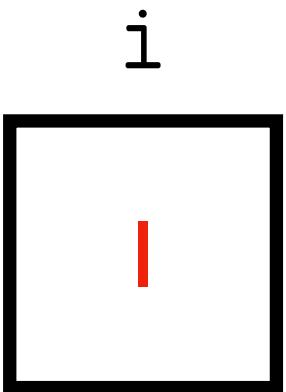
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



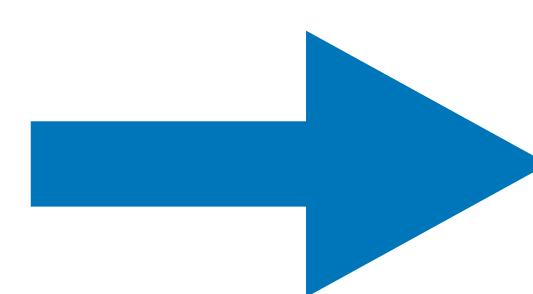
Output

0

# Statement while

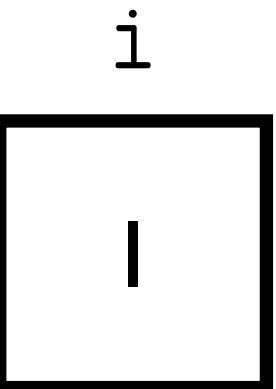
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```

Memoria



Output

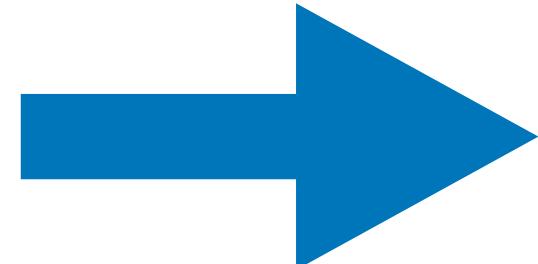
0

# Statement while

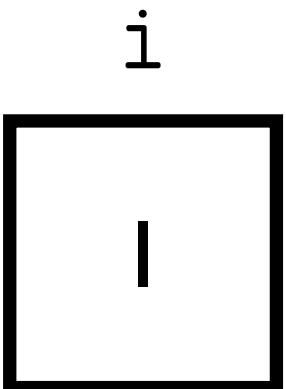
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



Output

0

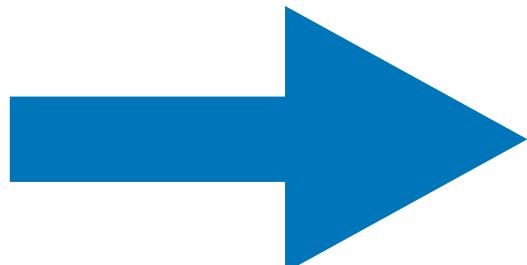
1

# Statement while

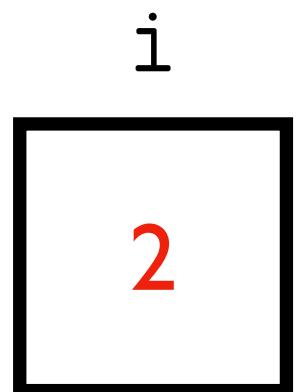
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



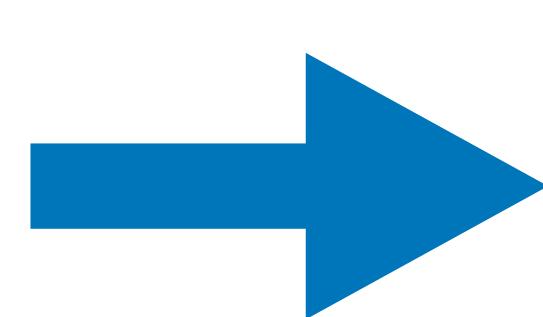
Output

0  
1

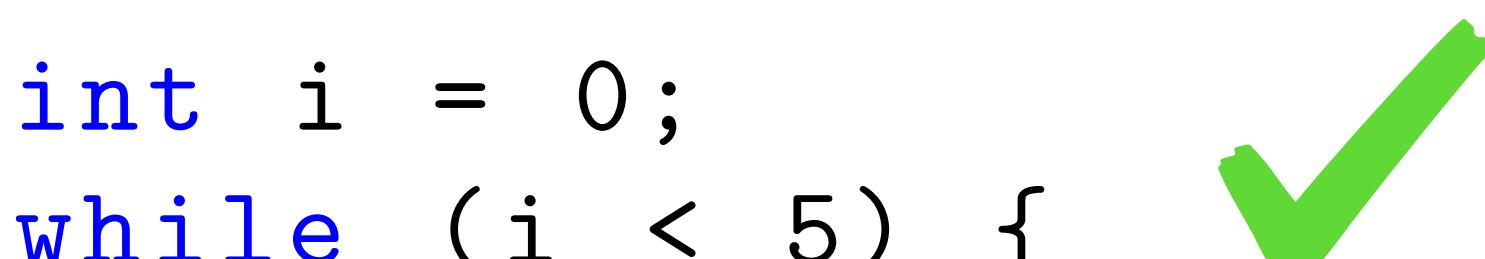
# Statement while

## Esercizio

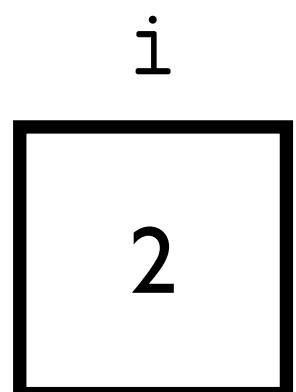
- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



Output

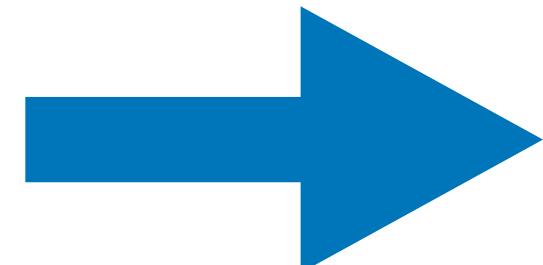
0  
1

# Statement while

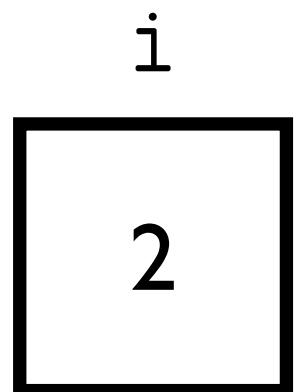
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



Output

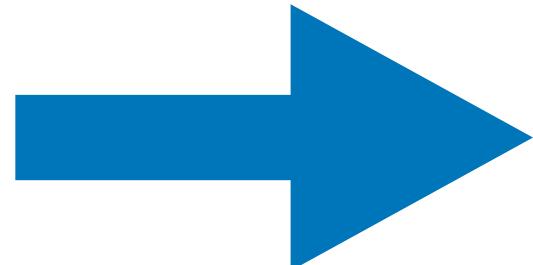
0  
1  
4

# Statement while

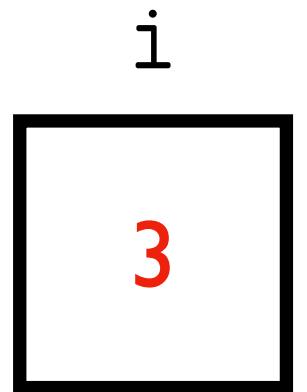
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



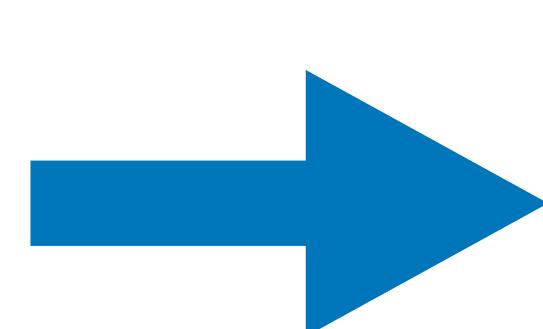
Output

0  
1  
4

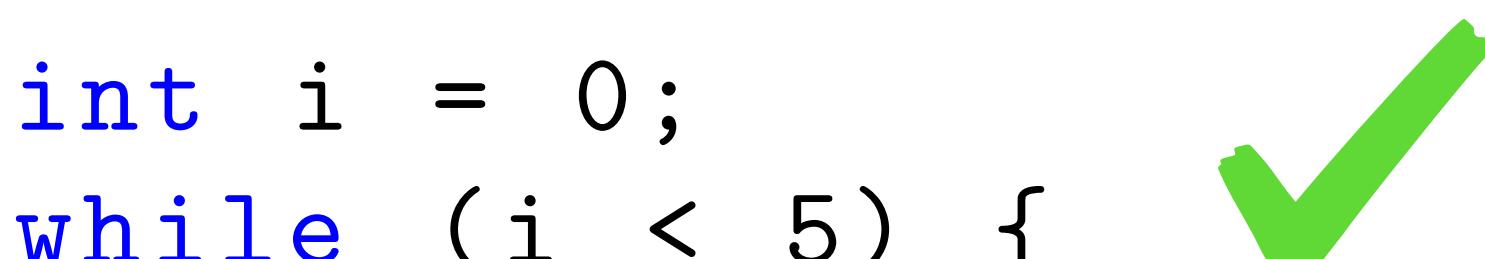
# Statement while

## Esercizio

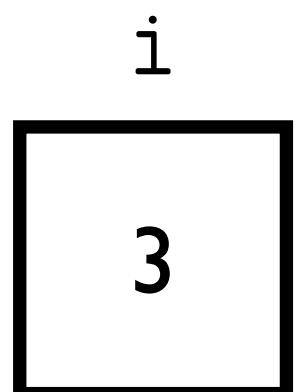
- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



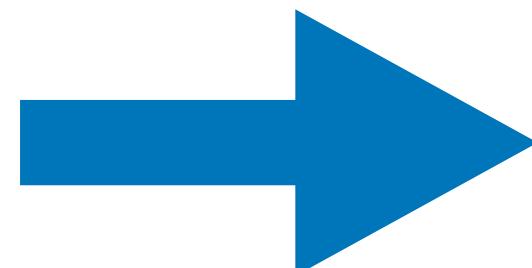
Output

0  
1  
4

# Statement while

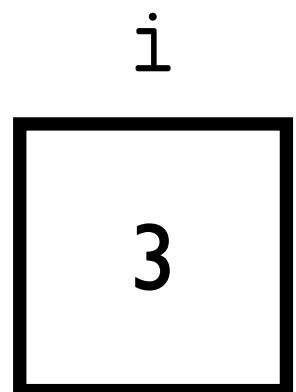
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```

Memoria



Output

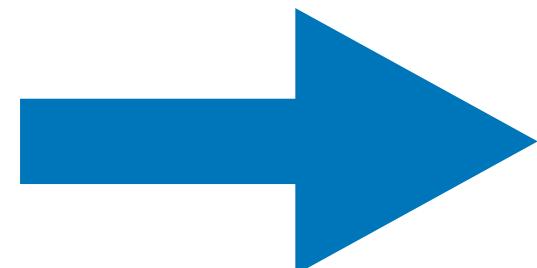
0  
1  
4  
9

# Statement while

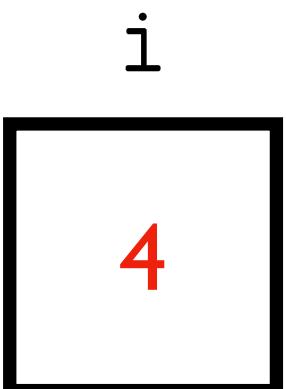
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



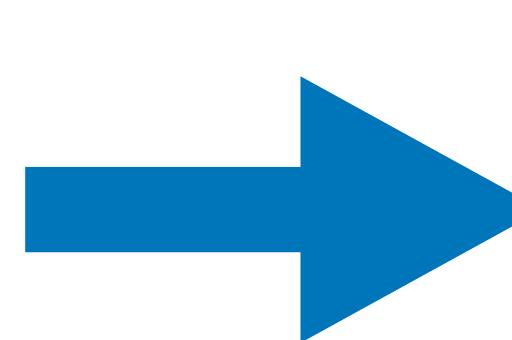
Output

0  
1  
4  
9

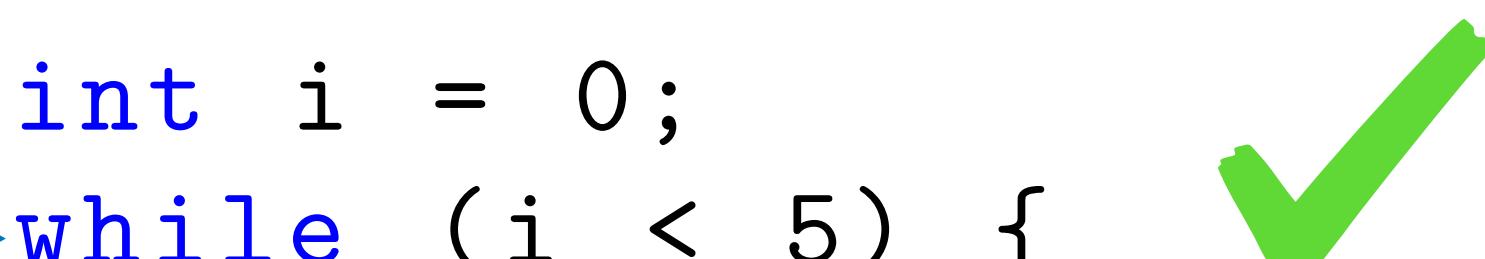
# Statement while

## Esercizio

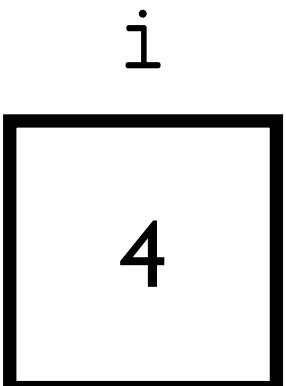
- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria



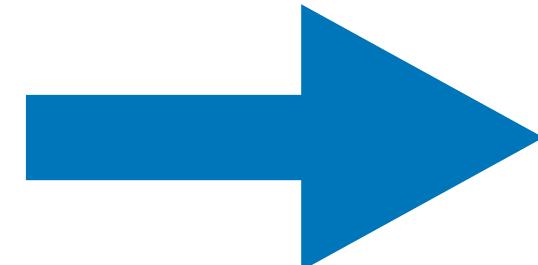
Output

0  
1  
4  
9

# Statement while

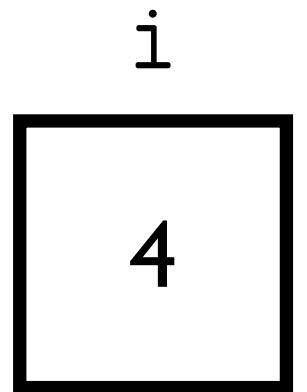
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```

Memoria



Output

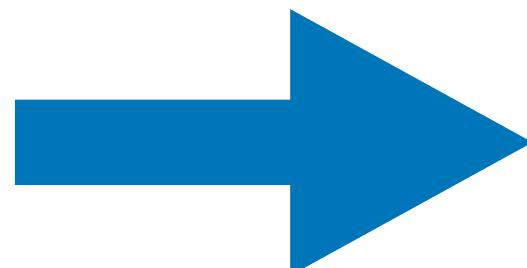
0
1
4
9
16

# Statement while

## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria

i	
---	--

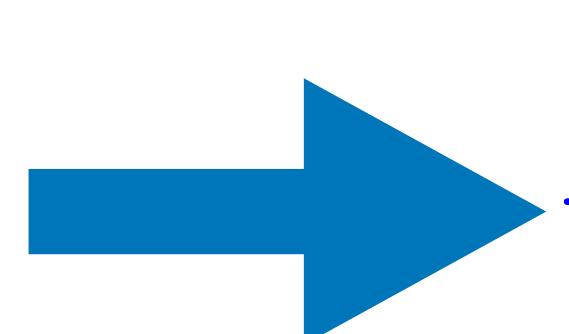
Output

0	0
1	1
4	4
9	9
16	16

# Statement while

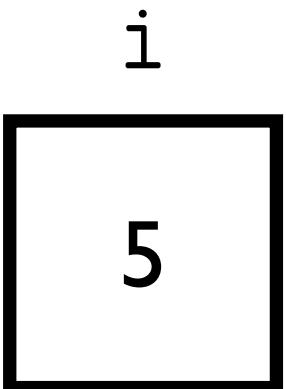
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$



```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```

Memoria



Output

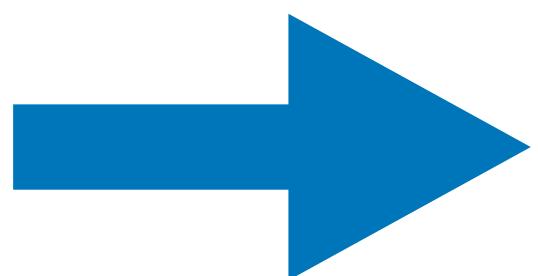
0
1
4
9
16

# Statement while

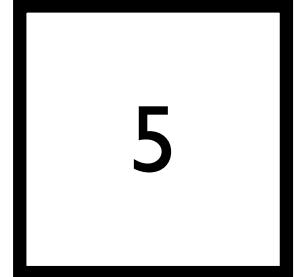
## Esercizio

- **Problema:** stampare i quadrati dei primi 5 interi  $\geq 0$

```
int i = 0;  
while (i < 5) {  
    cout << i * i << endl;  
    i++;  
}
```



Memoria

i	
---	---

Output

0	0
1	1
4	4
9	9
16	16

# cmath

```
#include <cmath>
```

- Libreria matematica per il calcolo di alcune operazioni matematiche
  - $\cos(e)$ : coseno
  - $\sin(e)$ : seno
  - $\sqrt{e}$ : radice quadrata
  - $\text{pow}(e_1, e_2)$ : potenza ( $e_1$  alla  $e_2$ )

<https://www.cplusplus.com/reference/cmath/>

# Statement while

```
while (exp)  
    stmt
```

- *stmt* è uno statement qualsiasi, quindi anche **blocchi e altri while statement**

# Statement while

## Esercizio

- **Problema:** preso in input un intero positivo  $n$ , stampare un triangolo rettangolo di asterischi di lato  $n$

$n = 1$

\*

$n = 3$

\*  
\* \*  
\* \* \*

$n = 5$

\*  
\* \*  
\* \* \*  
\* \* \* \*  
\* \* \* \* \*

# Statement while

## Esercizio

- **Problema:** preso in input un intero positivo  $n$ , stampare un triangolo rettangolo di asterischi di lato  $n$

$$n = 5$$

riga 0, stampo 1 carattere	*
riga 1, stampo 2 caratteri	* *
riga 2, stampo 3 caratteri	* * *
riga 3, stampo 4 caratteri	* * * *
riga 4, stampo 5 caratteri	* * * * *

alla riga  $i$ , stampo  $i + 1$  caratteri

- **Idea:** uso due contatori: uno mi tiene traccia di quale riga sto stampando, l'altro tiene traccia di quanti asterischi ho stampato su quella riga

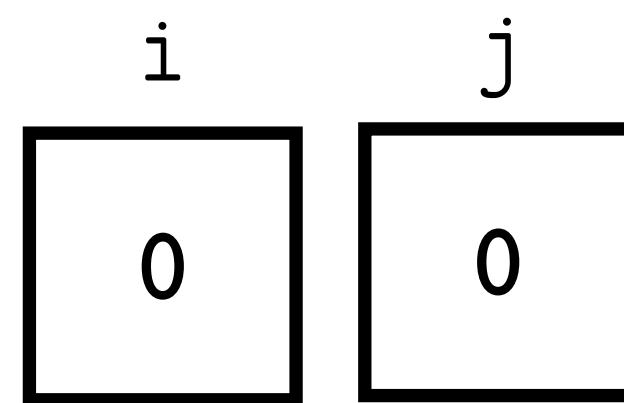
# Statement while

Esercizio ( $n = 3$ )

```
→ while (i < n) { ✓
    j = 0;
    while (j < i + 1) {
        cout << "* ";
        j++;
    }

    cout << endl;
    i++;
}
```

Memoria



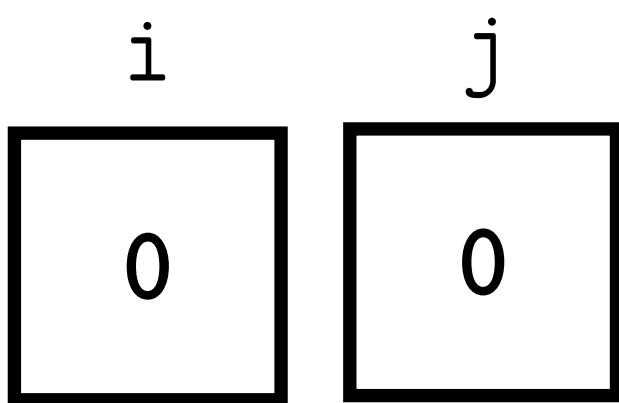
Output

# Statement while

# Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
    cout << endl;  
    i++;  
}
```

# Memoria



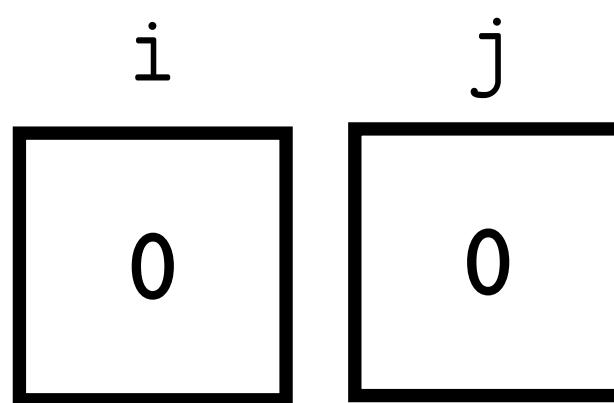
## Output

# Statement while

# Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
    cout << endl;  
    i++;  
}
```

## Memoria

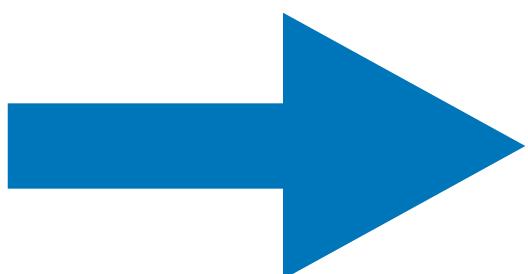


## Output

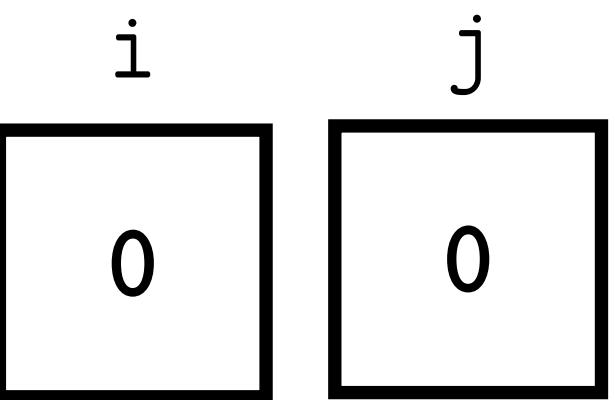
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



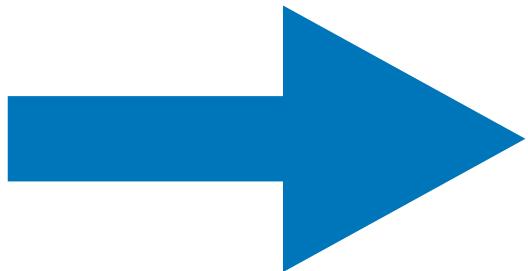
Output

\*

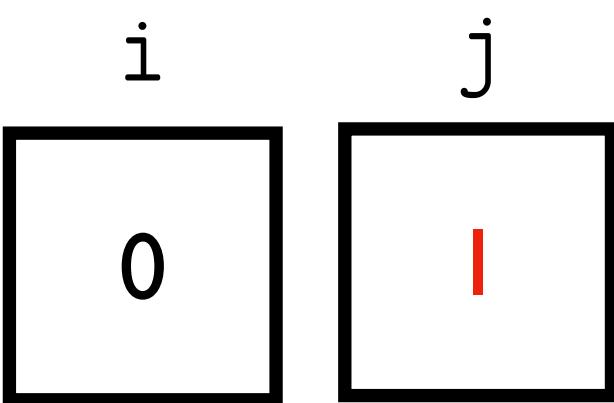
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



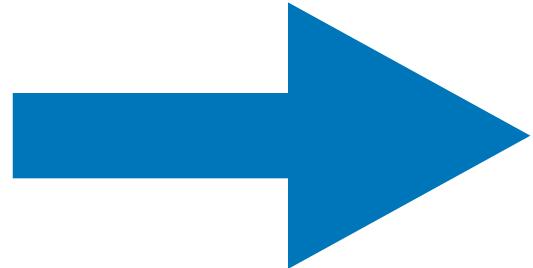
Output

\*

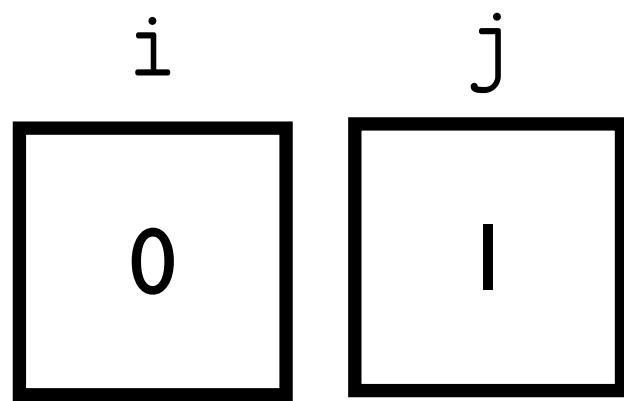
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



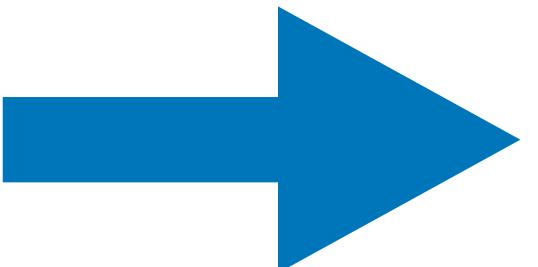
Output

\*

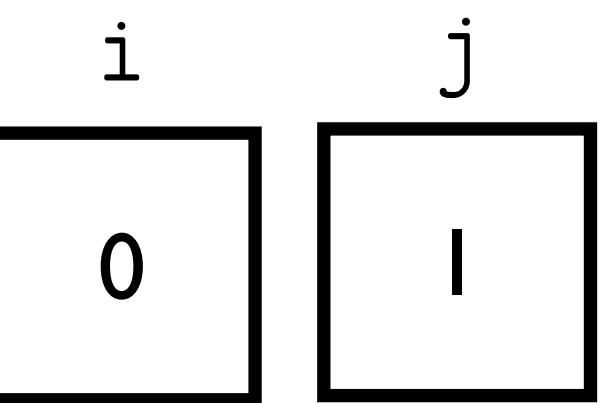
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



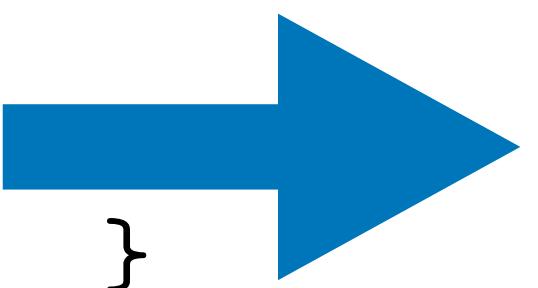
Output

\* (a capo)

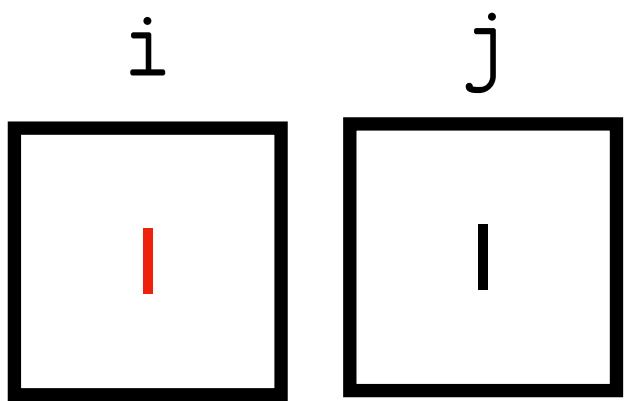
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



Output

\*

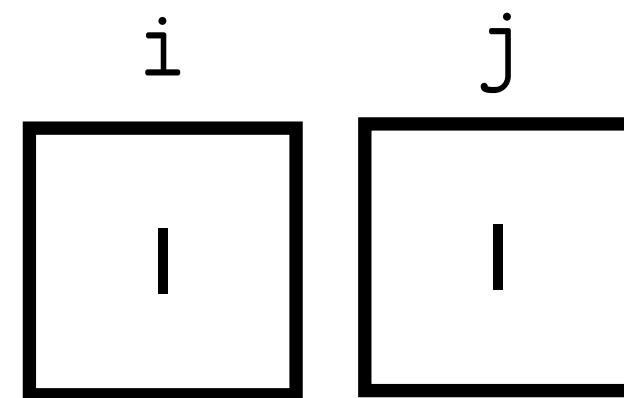
# Statement while

Esercizio ( $n = 3$ )

```
→ while (i < n) { ✓
    j = 0;
    while (j < i + 1) {
        cout << "* ";
        j++;
    }

    cout << endl;
    i++;
}
```

Memoria

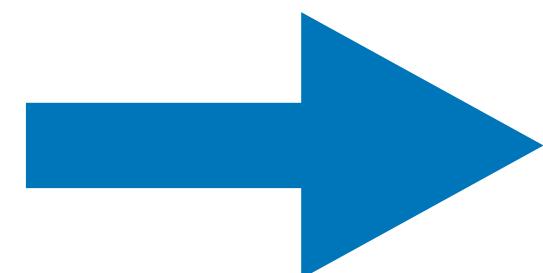


Output

\*

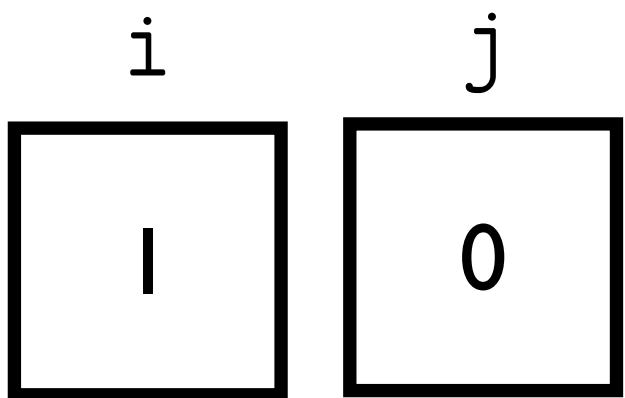
# Statement while

Esercizio ( $n = 3$ )



```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```

Memoria



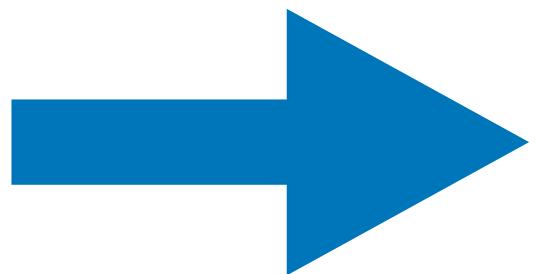
Output

\*

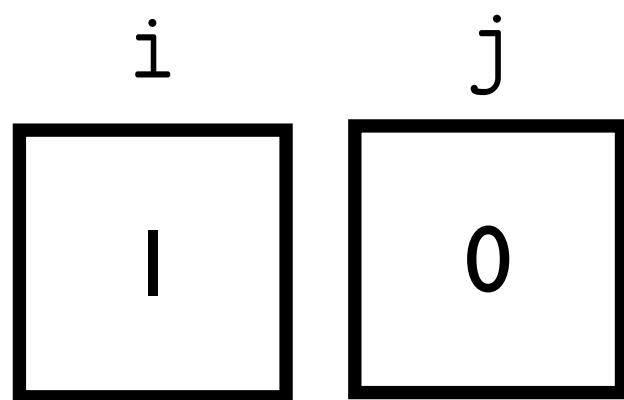
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria

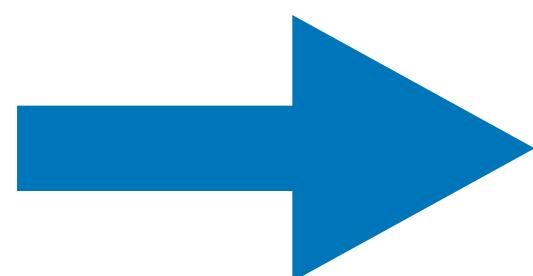


Output

\*

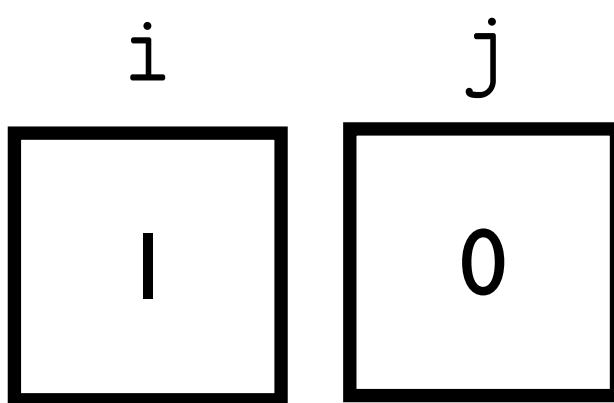
# Statement while

Esercizio ( $n = 3$ )



```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```

Memoria



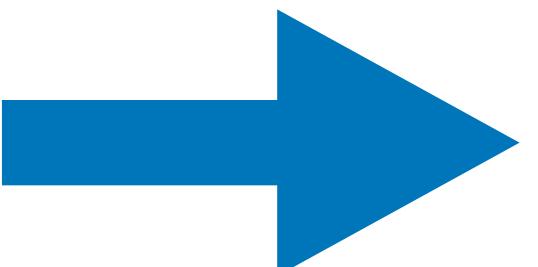
Output

\*  
\*

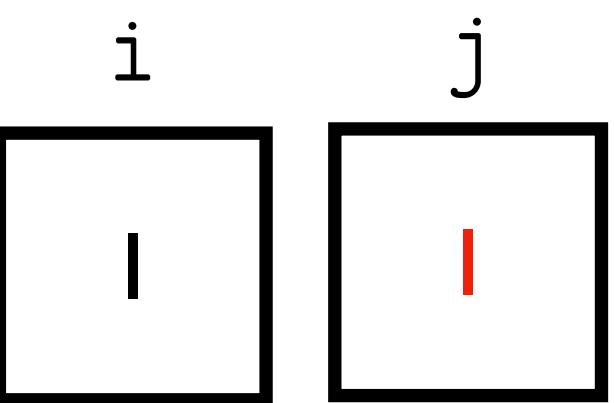
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



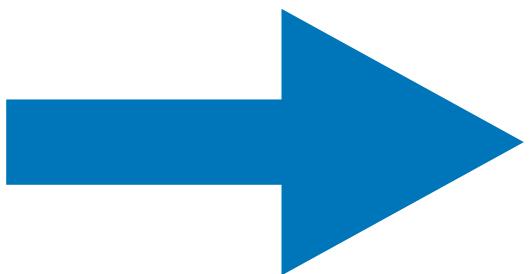
Output

\*  
\*

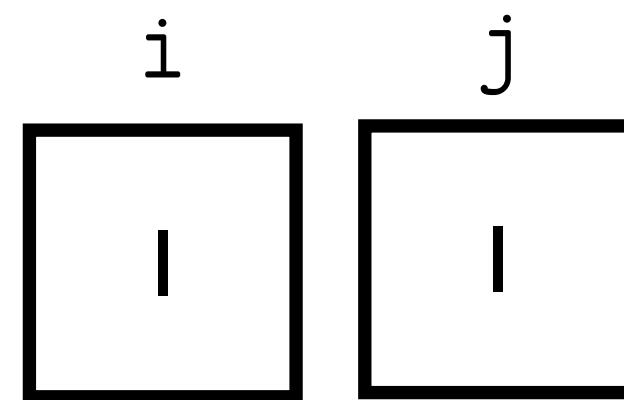
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



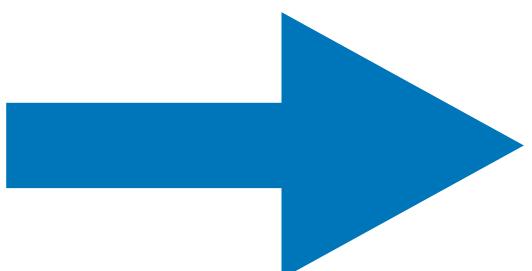
Output

\*  
\*

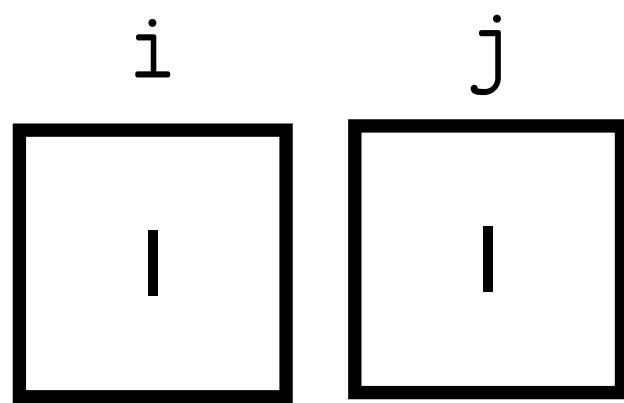
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



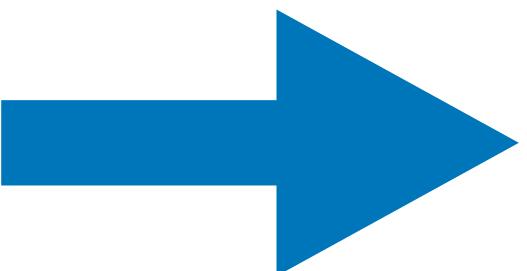
Output

\*  
\* \*

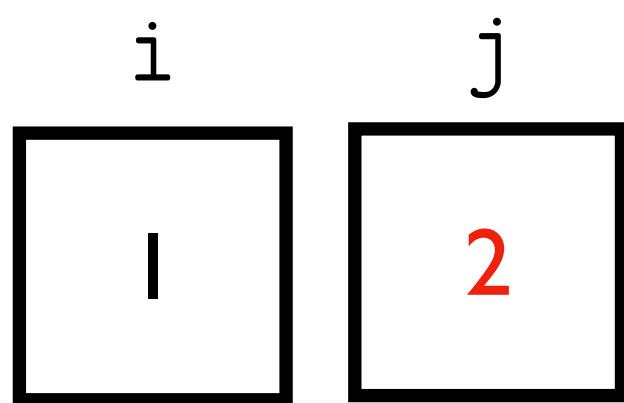
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



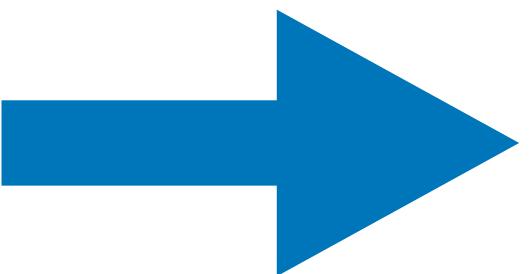
Output

\*  
\* \*

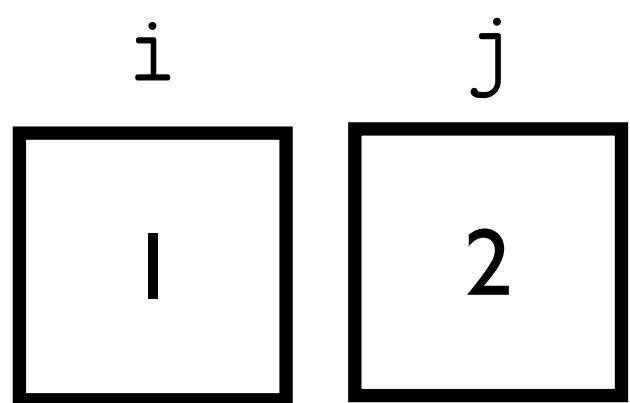
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



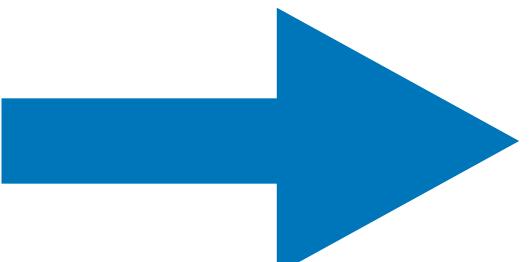
Output

\*  
\* \*

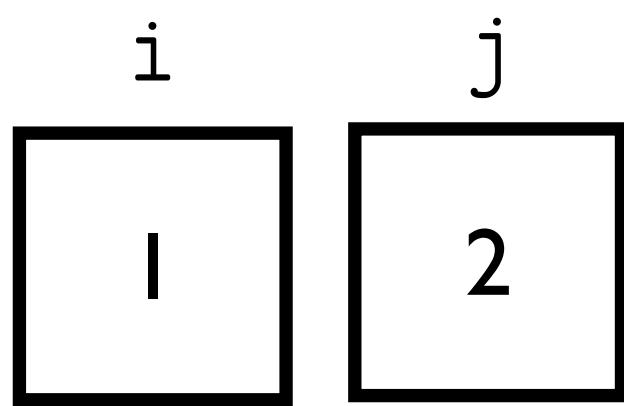
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



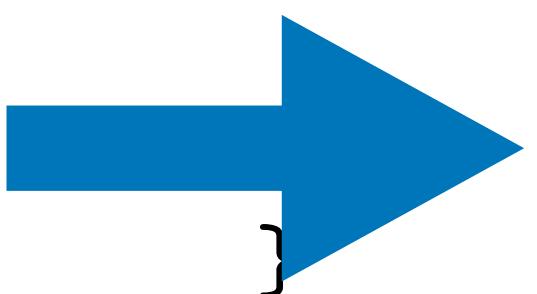
Output

\*  
\* \* (a capo)

# Statement while

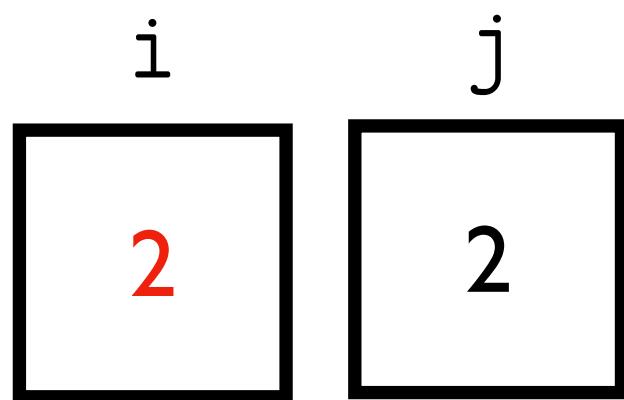
Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
}
```



```
cout << endl;  
i++;
```

Memoria

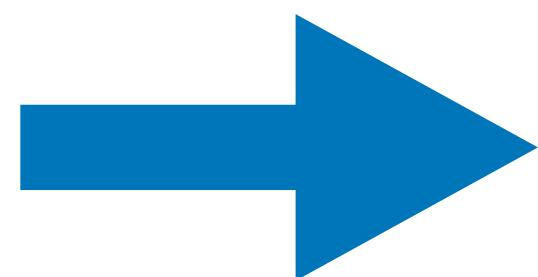


Output

\*  
\* \*

# Statement while

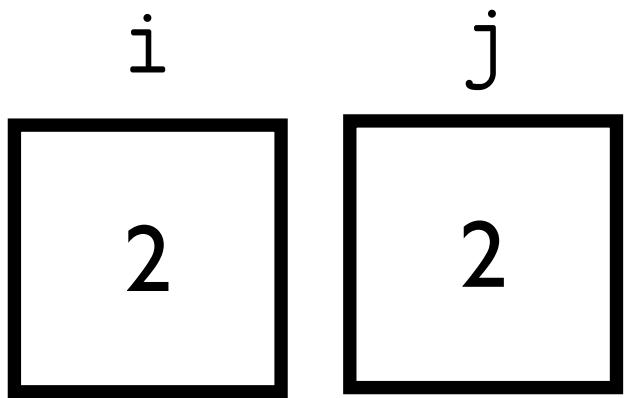
Esercizio ( $n = 3$ )



```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria

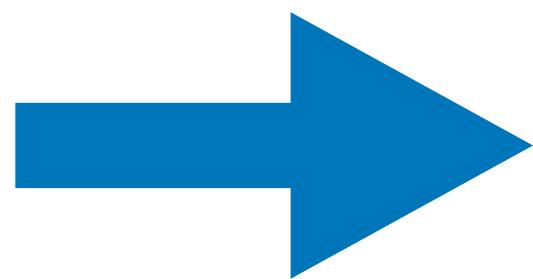


Output

\*  
\* \*

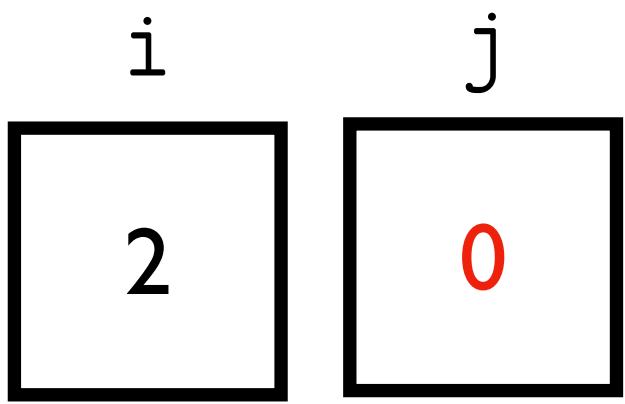
# Statement while

Esercizio ( $n = 3$ )



```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```

Memoria



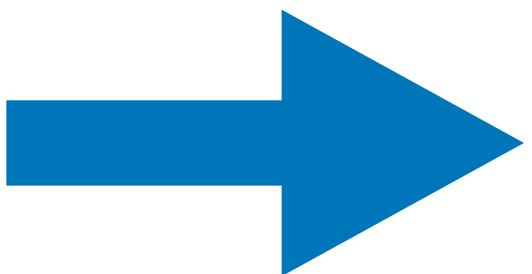
Output

\*  
\* \*

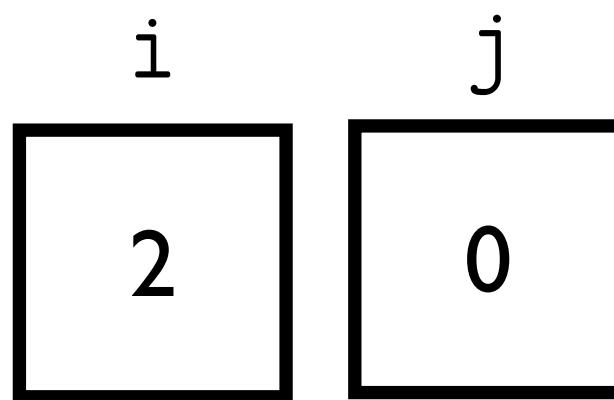
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



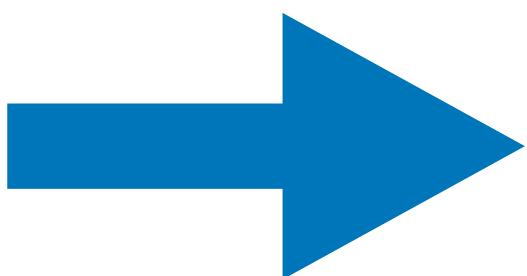
Output

\*  
\* \*

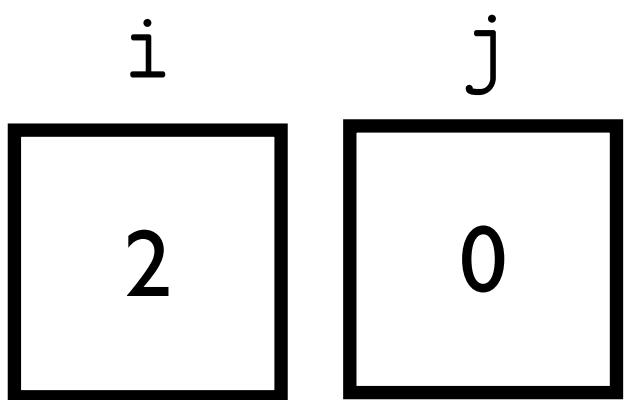
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



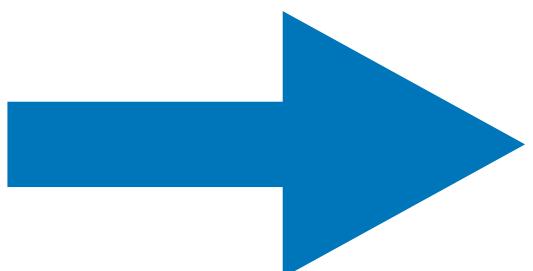
Output

\*  
\* \*  
\*

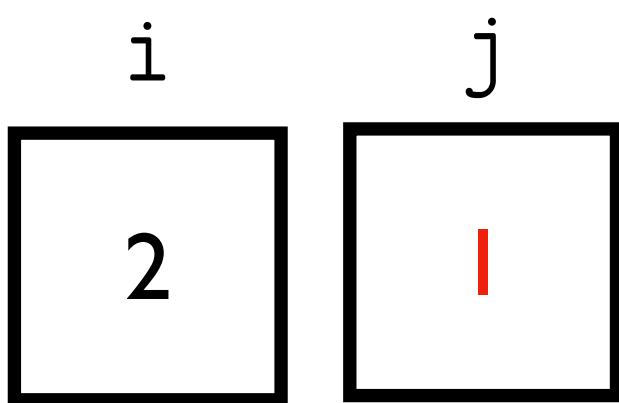
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



Output

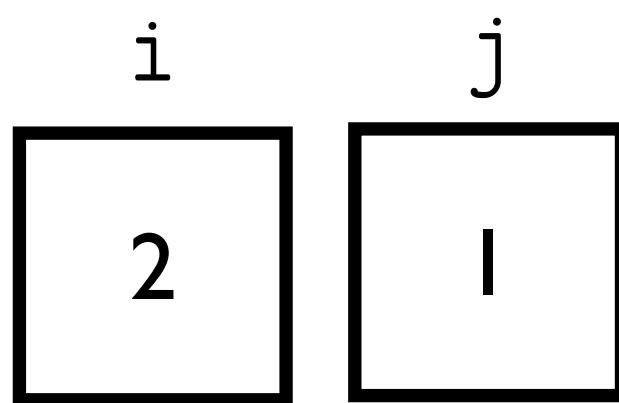
\*  
\* \*  
\*

# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```

Memoria



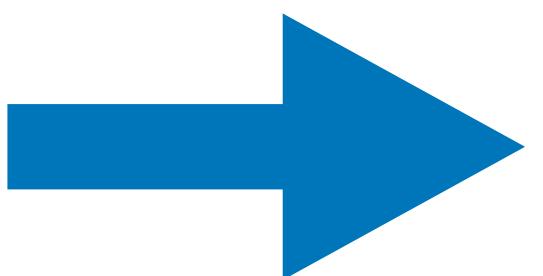
Output

\*  
\* \*  
\*

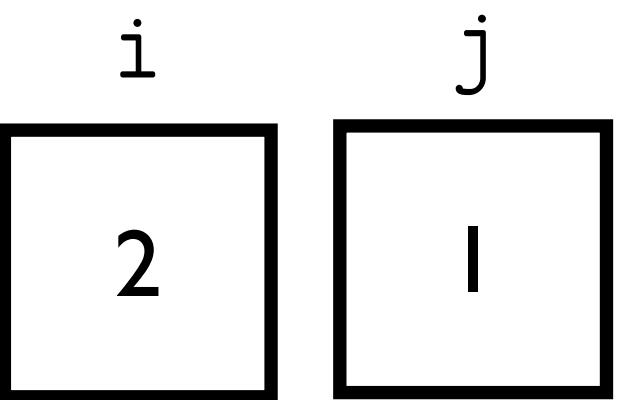
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



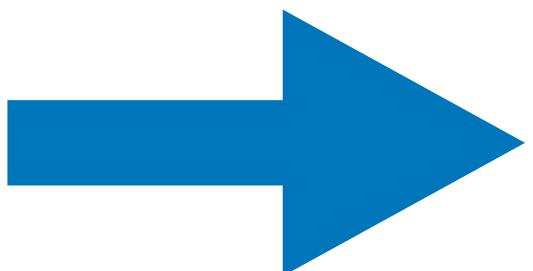
Output

\*  
\* \*  
\* \*

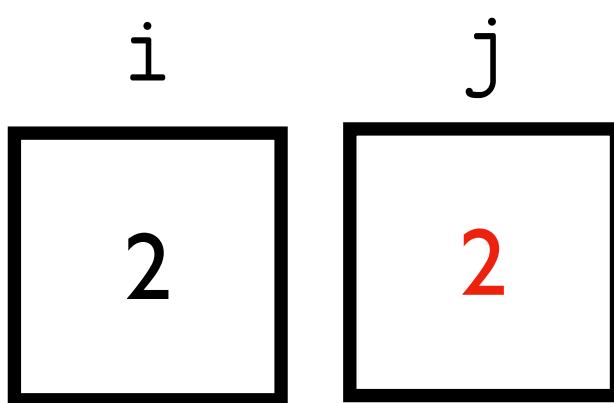
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



Output

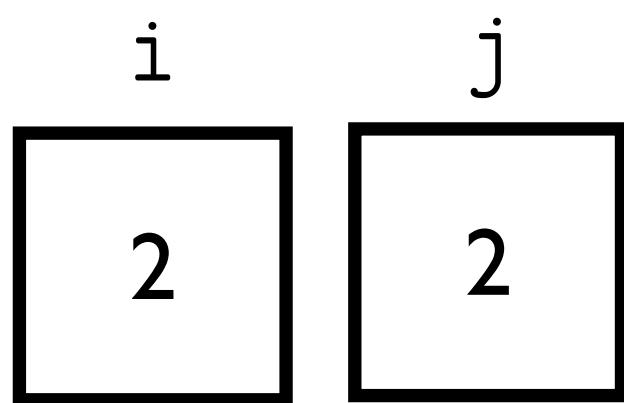
\*  
\* \*  
\* \*

# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```

Memoria



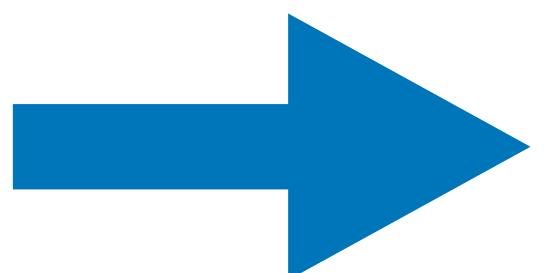
Output

\*  
\* \*  
\* \*

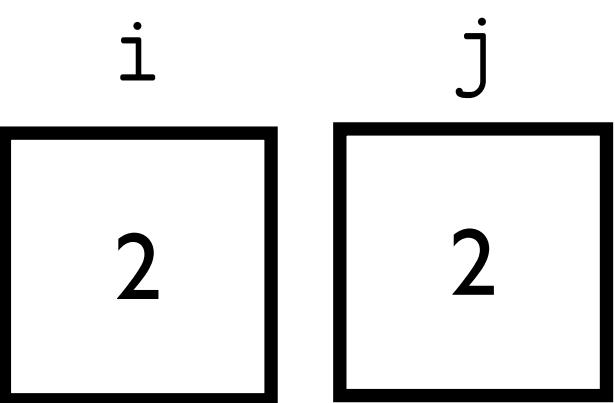
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



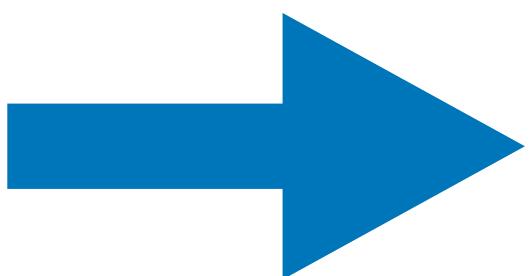
Output

\*  
\* \*  
\* \* \*

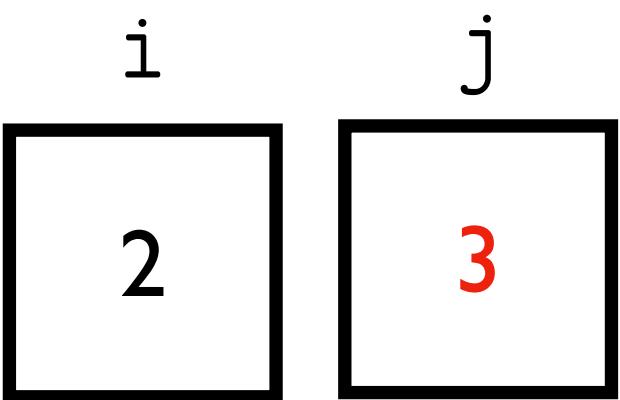
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



Output

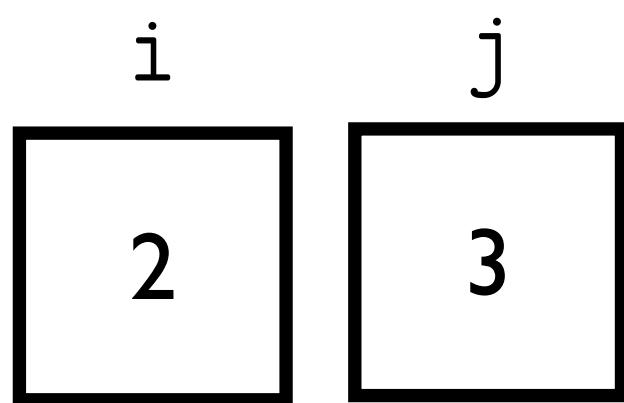
\*  
\* \*  
\* \* \*

# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```

Memoria



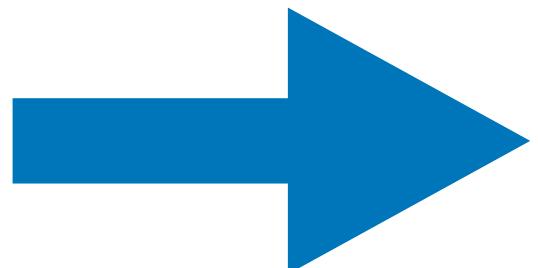
Output

\*  
\* \*  
\* \* \*

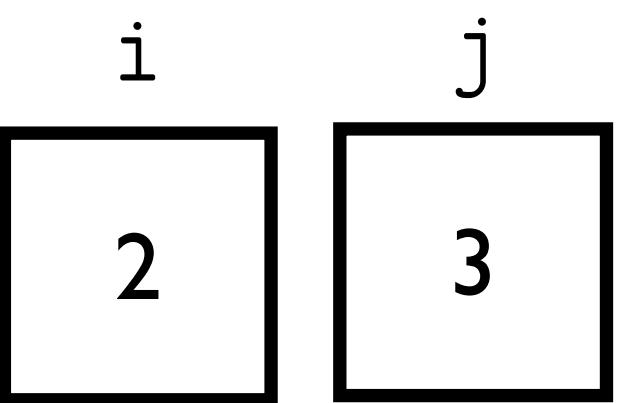
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



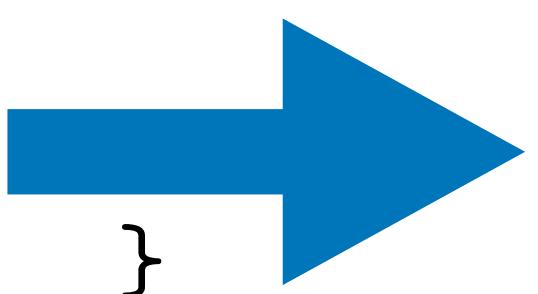
Output

\*  
\* \*  
\* \* \* (a capo)

# Statement while

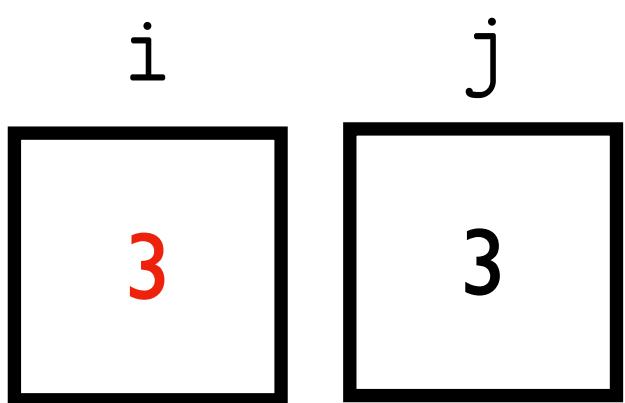
Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "*" ;  
        j++;  
    }  
}
```



```
cout << endl;  
i++;
```

Memoria

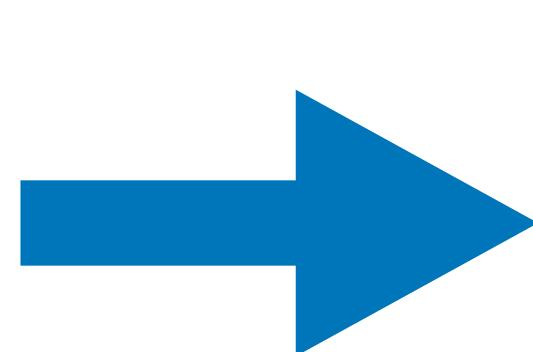


Output

\*  
\* \*  
\* \* \*

# Statement while

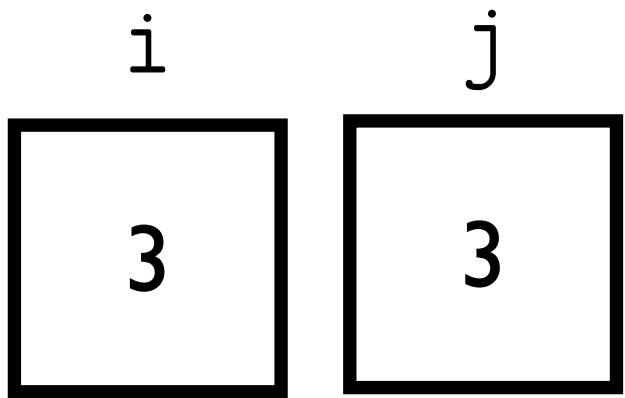
Esercizio ( $n = 3$ )



```
while (i < n) { X
    j = 0;
    while (j < i + 1) {
        cout << "*" " ";
        j++;
    }

    cout << endl;
    i++;
}
```

Memoria



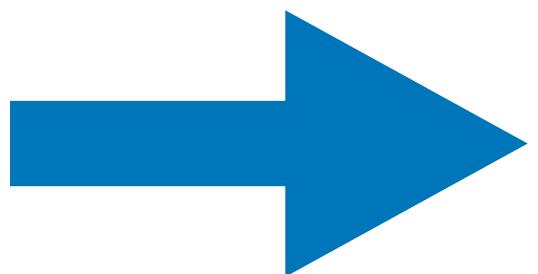
Output

\*  
\* \*  
\* \* \*

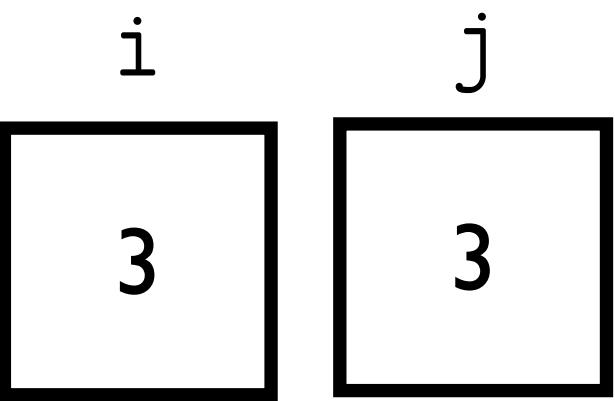
# Statement while

Esercizio ( $n = 3$ )

```
while (i < n) {  
    j = 0;  
    while (j < i + 1) {  
        cout << "* ";  
        j++;  
    }  
  
    cout << endl;  
    i++;  
}
```



Memoria



Output

\*  
\* \*  
\* \* \*

# Statement while

## Programmi che non terminano

```
int i = 0;  
while (i < 10)  
    somma = somma + 1;
```

La condizione  $i < 10$  è sempre vera!  $i$  non viene mai aggiornata

---

```
int i = 1;  
while (i > 0 || i <= 10)  
    i = i + 1;
```

La condizione  $i > 0$  è sempre vera!

---

```
int i = 1;  
while (i <= 10)  
    cout << i * i << endl;  
    i = i + 1;  
cout << "terminato!" << endl;
```

La condizione  $i \leq 10$  è sempre vera!  
(qual è il corpo del while?)

# Problema della terminazione

- **Errori semantici:** il programma è sintatticamente corretto (compila e non ha errori sintattici), ma si comporta in maniera inaspettata (e.g., non termina)
- **Gli errori sintattici sono rilevabili dal compilatore!**



# Problema della terminazione

- **Errore semantico:** il programma è sintatticamente corretto, ma si comporta in maniera inaspettata (e.g., non termina)
- **Gli errori semantici **NON** sono rilevabili dal compilatore!**
  - ◆ non può controllare se il programma termina o meno
- **Problema della terminazione (*halting problem*)**

Non esiste un programma  $H$  che, preso in input un qualsiasi programma  $P$ , riesca a dire se  $P$  termina o meno

# Problema della terminazione

- Link (~8mins)



# Problema della terminazione

- Tuono Pettinato, Francesca Riccioni. “*Enigma. La strana vita di Alan Turing*” (2012)

