



UNIVERSITÀ  
DI TRENTO

Dipartimento di Ingegneria e  
Scienza dell'Informazione  
DISI - Trento

# Programmazione 1

## 08 - Esercitazione

Andrea Mazzullo  
[andrea.mazzullo@unitn.it](mailto:andrea.mazzullo@unitn.it)

Anno Accademico 2023/2024

# Nelle puntate precedenti

- Tipo derivato riferimento-a (&)

```
int x = 5;  
int & y = x;
```

```
x++;      // 6  
cout << y << endl;    // 6  
y++;      // 7  
cout << x << endl;    // 7
```

```
int k = 20;  
y = k;  
cout << x << endl;    // 20
```

# Nelle puntate precedenti

- Tipo derivato puntatore-a (\*)

```
int x = 5;
int * y = &x; // L'operatore & da l'indirizzo di memoria

x++;          // 6
cout << y << " " << *y << endl;    // 6
(*y)++;       // 7
cout << x << endl;    // 7

int k = 20;
y = &k;
cout << x << endl;    // 7
```

# 00 - Funzione

- Sotto-programma con una funzionalità ben definita che ritorna un valore

parametri formali (esistono solo all'interno della funzione)

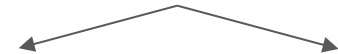
```
int calcolaSomma(int addendo1, int addendo2) {  
    int somma;  
    somma = addendo1 + addendo2;  
    return somma;  
}
```

↑  
valore di ritorno

# 00 - Procedura

- Sotto-programma con una funzionalità ben definita ~~che ritorna un valore~~

parametri formali



```
void stampaSomma(int addendo1, int addendo2) {  
    int somma;  
    somma = addendo1 + addendo2;  
    cout << somma << endl;  
    return;  
}
```

↑  
opzionale

# 00 - Struttura

```
1. #include <iostream>
2. using namespace std;
3.
4. int calcolaSomma(int addendo1, int addendo2);
5.
6. int main() {
7.     int a = 1, b = 2;
8.     cout << calcolaSomma(a, b);
9. }
10.
11. int calcolaSomma(int addendo1, int addendo2) {
12.     int somma;
13.     somma = addendo1 + addendo2;
14.     return somma;
15. }
```

# 00 - Struttura

```
1. #include <iostream>
2. using namespace std;
3.
4. int calcolaSomma(int addendo1, int addendo2);
5.
6. int main() {
7.     int a = 1, b = 2;
8.     cout << calcolaSomma(a, b);
9. }
10.
11. int calcolaSomma(int addendo1, int addendo2) {
12.     int somma;
13.     somma = addendo1 + addendo2;
14.     return somma;
15. }
```

**Dichiarazione**

# 00 - Struttura

```
1. #include <iostream>
2. using namespace std;
3.
4. int calcolaSomma(int addendo1, int addendo2);
5.
6. int main() {
7.     int a = 1, b = 2;
8.     cout << calcolaSomma(a, b);
9. }
10.
11. int calcolaSomma(int addendo1, int addendo2) {
12.     int somma;
13.     somma = addendo1 + addendo2;
14.     return somma;
15. }
```

**Chiamata**



# 00 - Struttura

```
1. #include <iostream>
2. using namespace std;
3.
4. int calcolaSomma(int addendo1, int addendo2);
5.
6. int main() {
7.     int a = 1, b = 2;
8.     cout << calcolaSomma(a, b);
9. }
10.
11. int calcolaSomma(int addendo1, int addendo2) {
12.     int somma;
13.     somma = addendo1 + addendo2;
14.     return somma;
15. }
```

**Definizione**

# 01 - Stampa Maiuscolo

**Dato in input un carattere, scrivere un programma che controlli con una funzione se il carattere fornito in input sia una lettera dell'alfabeto minuscola. In caso affermativo, il programma deve stampare a video il carattere in maiuscolo, usando una funzione per la conversione.**

`'a' => 'A'`

## 02 - Stampa Maiuscolo v2

**Dato in input un carattere, scrivere un programma che controlli con una funzione se il carattere fornito in input sia una lettera dell'alfabeto minuscola. In caso affermativo, il programma deve stampare a video il carattere in maiuscolo, usando una funzione per la conversione.**

**La funzione di conversione non deve ritornare alcun valore.**

`'a' => 'A'`

## 03 - Inverti Numero

**Scrivere un programma che prenda in input un numero e stampi a video il numero invertito usando una procedura.**

345 => 543

## 04 - Liar's Dice

**Scrivere un programma che lanci 10 dadi a 6 facce utilizzando la funzione presentata sotto. Chiedere poi all'utente di indovinare quanti dadi sono risultati essere 1.**

```
#include <cstdlib>
srand(time(NULL));

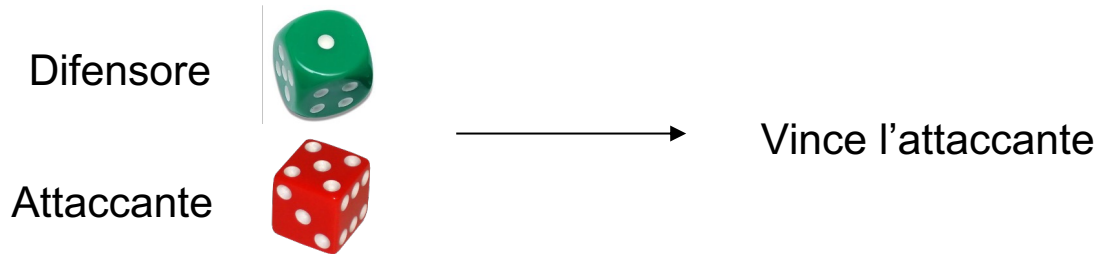
int random_number = rand() % 6 + 1
```

Per maggiori informazioni:

<http://www.cplusplus.com/reference/cstdlib/rand/>

## 05 - RisiKo! 1vs1

**Scrivere un programma che simuli un attacco 1 contro 1 a RisiKo!. Tirare 1 dado a 6 facce per l'attaccante e 1 dado per il difensore. Dichiarare infine il numero più alto come il vincitore.**



## 06 - RisiKo! 2vs2

**Scrivere un programma che simuli un attacco 2 contro 2 a RisiKo!. Tirare 2 dadi a 6 facce per l'attaccante e 2 dadi per il difensore. Confrontare il dado più alto dell'attaccante contro il più alto del difensore, e il più basso dell'attaccante contro il più basso del difensore. Dichiarare infine gli scontri vinti dall'attaccante e quelli vinti dal difensore.**

```
#include<algorithm>
int massimo = max(n1, n2)
int minimo= min(n1, n2)
```

Per maggiori informazioni:

<http://www.cplusplus.com/reference/algorithm/>