

# 27. Dal C al C++

Corso di Informatica

Corso di Laurea in Matematica (D.M. 270/04) - A.A. 2020/2021

Angelo Cardellicchio

[angelo.cardellicchio@uniba.it](mailto:angelo.cardellicchio@uniba.it)

09/12/2020

# Outline

- Rapporto tra C e C++
- Alcune differenze generali
- Introduzione alla programmazione orientata agli oggetti (OOP)
- Le classi
- Alcune differenze legate alla OOP

# Rapporto tra C e C++

- Il C++ è un *superset* del C
- Cio significa che un programma scritto in C può essere (abbastanza) tranquillamente eseguito da un compilatore C++...
  - ...*ma il contrario non è necessariamente vero!*
- In generale, il C++ è un linguaggio di concezione più moderna rispetto al C, ed ha, nel corso del tempo, acquisito diverse funzionalità che lo hanno reso indiscutibilmente più versatile e 'semplice' da utilizzare.

# Alcune differenze generali

Argomento	C	C++
Numero di keyword	32	52
Paradigma adottato	Imperativo	Imperativo/Orientato agli oggetti
Focus	Metodi	Dati
Tipi definiti dall'utente	Supporto limitato	Pieno supporto (classi)
Gestione eccezioni	Non presente	Presente

# Introduzione alla OOP

- La *programmazione orientata agli oggetti* è un paradigma che passa dal focus sulle **funzioni** (centrale nel C e nel paradigma procedurale/imperativo) a quello sui **dati**
- Nella OOP, **tutto è un oggetto**
- Ad esempio, abbiamo visto che C può combinare le parole chiave **typedef** e **struct** per definire il tipo **STUDENTE**
- In C++, possiamo creare degli oggetti di tipo **Studente** utilizzando adeguatamente il concetto di **classe**

# Introduzione alla OOP - Classi

- Una **classe** è un 'prototipo' per un determinato tipo di oggetti
  - Possiamo avere, come già detto, una classe **Studente**, che rappresenta tutte le proprietà e le azioni associate ad uno studente...
  - ...ma potremo avere anche una classe **Auto**, che rappresenta tutte le proprietà ed azioni associate ad un'auto...
  - ...ed una classe **Motore**, che definisce i comportamenti generici dei motori...
  - ...e *via dicendo*.
- Ogni 'prototipo' (classe) può essere utilizzato per creare una specifica **istanza** dell'oggetto.
- Ogni classe ha **metodi** ed **attributi**.

# Introduzione alla OOP - Classi

- Facciamo un breve esempio (che esanderemo nelle prossime lezioni).

```
class Studente {  
    char* nome;  
    char* cognome;  
    int eta;  
    // ...  
    char* getNome() {  
        return nome;  
    }  
    void setNome(char* nuovoNome) {  
        nome = nuovoNome;  
    }  
}
```

```
int main() {  
    // Creo il primo studente...  
    Studente tizio = new Studente();  
    // Creo il secondo studente...  
    Studente caio = new Studente();  
    // ...  
}
```

# Alcune differenze legate alla OOP

Argomento	C	C++
Concetto di namespace	Non presente	Presente
Allocazione dinamica	Sì, con malloc(), calloc() e free()	Sì, con new e delete
Variabili e funzioni	Indipendenti	Incapsulate negli oggetti
Modificatori di accesso	Non presenti	Presenti



# Domande?

42