

# C++

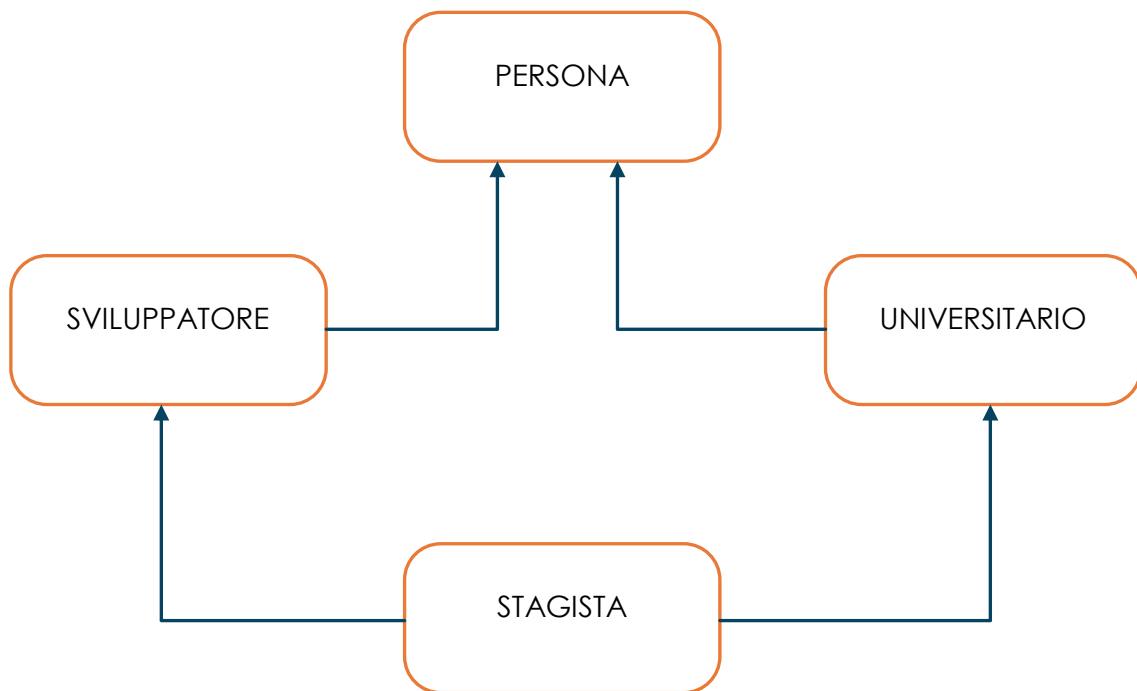
INFORMATICA III – MODULO A

LORENZO MILESI – 1030232

Università degli studi di Bergamo

## Presentazione

Il progetto realizzato in C++ descrive la struttura organica di una generica situazione aziendale ed extra-aziendale nell'ambito dello sviluppo software, in cui si affacciano persone di diverso tipo: lo sviluppatore, l'universitario e lo stagista (che è dunque uno sviluppatore-universitario). Ogni entità è realizzata attraverso una classe separata, secondo lo schema seguente:



La gestione di ogni classe è realizzata in file separati (*header* e *source*), in aggiunta al file contenente il *main* di avvio.

La memorizzazione di ogni entità passa attraverso una classe ulteriore, definita *Lista*, creata per agevolare la gestione delle chiamate in modo da rendere il codice più pulito, specialmente all'interno del *main*. La classe *Lista*, infatti, ha un metodo `aggiungi(Persona p)` che consente di effettuare un `push_back` di `p` su un oggetto di tipo *Lista*. Avere a disposizione questa classe ausiliaria consente inoltre di scalare meglio le funzionalità, per esempio aggiungendo nuovi metodi per filtrarne i dati.

La classe *Lista* calcola inoltre l'età media degli studenti utilizzando una funzione ausiliaria di **tipo T-generico** (template).

Ogni classe ha un proprio costruttore e un distruttore (sfruttato in caso di delete sull'oggetto). Inoltre, i vari campi e metodi sono dichiarati **private**, **public** o **protected** a seconda dei casi (per esempio, i metodi sono public e i campi protected o private a seconda del grado di "protezione" desiderato).

Molti metodi sono dichiarati **virtual** per consentirne un override nelle classi figlie.

Ogni classe contiene un metodo `toString()` di tipo void che effettua un `cout` diretto dell'elemento stesso, secondo un pattern hard-coded per ogni oggetto. Lo stesso

metodo è presente inoltre nella classe *Lista*, e si appoggia ad un iteratore per effettuare la lettura dei vari elementi che contiene. Ogni elemento chiamato con `toString()` riporta a schermo il tipo dell'oggetto: Universitario, Sviluppatore, Stagista o generica Persona.

Per restrizioni di MINGW su Windows la stampa a schermo di interi è demandata ad una prima fase di conversione utilizzando la libreria **sstream**.

Il main dell'applicazione effettua diverse prove, tra cui l'inserimento dei vari oggetti nella lista e l'invocazione di alcuni metodi sulla lista stessa:

- `stampa`: per la stampa di tutti gli elementi
- `stampa(1994)`: per la stampa di tutti gli elementi nati nel 1994
- `etaMedia`: per la stampa dell'età media (template generico T)

## Output

Esempio di output:

```
# STAMPA generica #
[Persona: Lorenzo, Milesi, 1994]
[Persona: Mario, Rossi, 1995]
[Universitario: Alberto, Bianchi, 1030232, Informatica]
[Universitario: Mario, Rossi, 1030333, Informatica]
[Sviluppatore: Giuseppe, Verdi, 1975, 2016@27, Madbit Entertainment]
[Sviluppatore: Matteo, Milesi, 1992, 2013@30, Madbit Entertainment]
[Persona: Luca, Neri, 1995]
[Stagista: Lorenzo, Milesi, 1994, 2018@20, Madbit Entertainment, SCADE:
ottobre 2019]
# STAMPA 1994 #
[Persona: Lorenzo, Milesi, 1994]
[Universitario: Alberto, Bianchi, 1030232, Informatica]
[Stagista: Lorenzo, Milesi, 1994, 2018@20, Madbit Entertainment, SCADE:
ottobre 2019]
# ETA' MEDIA TEMPLATE GENERICO #
Età media: 1991
```