

# Esercitazione - Grafi

## Fondamenti 2 - Corso di Laurea in Informatica

Facendo uso della classe Grafo e GrafoOrientato, scrivere un main implementando, e successivamente "testando", le seguenti funzioni:

- **void stampaGrafo(const Grafo& g)** che stampa tutte le informazioni relative al grafo g;
- **int getNodeConGradoMassimo(const GrafoNonOrientato& g)** che restituisce il nodo avente grado massimo all'interno del grafo non orientato g;
- **bool stessoNumeroNodiStessoGrado (const GrafoNonOrientato& g1, const GrafoNonOrientato& g2)** che restituisce *true* se e solo se g1 e g2 hanno lo stesso numero di nodi per ogni possibile grado.

Esempio:

g1:  $\text{deg}(0) = 0, \text{deg}(1) = 3, \text{deg}(2) = 3, \text{deg}(3) = 1$ ;  
g2:  $\text{deg}(0) = 3, \text{deg}(1) = 3, \text{deg}(2) = 1, \text{deg}(3) = 0$ .

La funzione dovrebbe restituire *true* in quanto il numero di nodi aventi grado 0 è pari a 1 per entrambi i grafi, il numero di nodo aventi grado 1 è pari a 1 per entrambi i grafi, il numero di nodi aventi grado 2 è pari a 0 per entrambi i grafi ed infine il numero di nodi aventi grado 3 è pari a 2 per entrambi i grafi.

- **bool almenoUnNodoAdiacenteATutti(const GrafoNonOrientato& g)** che restituisce *true* se e solo se esiste almeno un nodo in g che sia adiacente a tutti i rimanenti.
- **CoppiaNodi getCoppiaConPiuAdiacenti(const GrafoNonOrientato& g)** che restituisce la coppia di nodi che hanno il maggior numero di nodi adiacenti in comune.  
Si noti che tale coppia va restituita all'interno di un oggetto di tipo CoppiaNodi che deve rappresentare un `pair<int,int>`.
- **bool inUnCiclo(const Grafo& g, int nodo)** che restituisce *true* se e solo se il nodo "nodo" si trova in un ciclo all'interno di g.

### PER CASA:

- **bool connesso(const GrafoNonOrientato& g)** che restituisce *true* se e solo se il grafo *g* è connesso.  
Si noti che, essendo *g* un grafo non orientato, non occorre controllare che per ogni coppia di nodi debba esistere un cammino che li unisce.
- **bool proprieta\_1(const Grafo& g, int k)** che restituisce *true* se e solo se vale la seguente proprietà per il grafo *g*: il numero di coppie di nodi che hanno almeno un adiacente in comune deve essere maggiore o uguale a *k*. La coppia generica  $(i, j)$  è considerata uguale alla coppia  $(j, i)$  e, dunque, va contata una sola volta.