

## TP2 – Constructeurs, Destructeurs et Allocation dynamique

ING2 – Programmation C++

GSI – 2020/2021

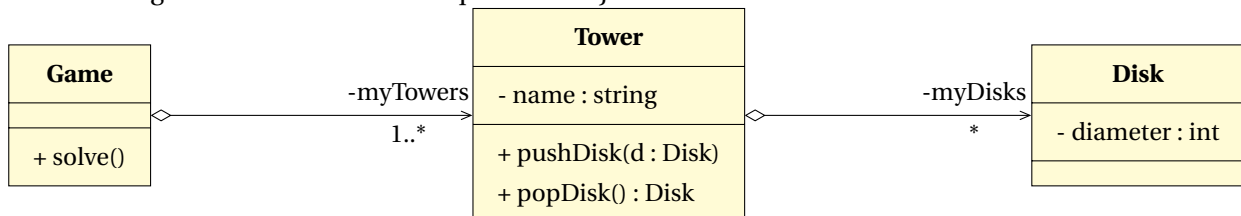


### 1 Paires

Récupérez les fichiers `Pair.h` et `main.cpp`, puis implémentez les méthodes de la classe `Pair` dans le fichier `Pair.cpp` et les testez.

### 2 Tours de Hanoï

Voici un diagramme de classes UML qui décrit le jeu Tours de Hanoï.



**Implémentez ces classes en C++, en respectant les contraintes suivantes :**

1. Un fichier.h et un fichier.cpp pour chaque classe.
2. L'encapsulation des données.
3. Des constructeurs et destructeurs que vous jugez utiles.
4. Des méthodes getter et setter que vous jugez utiles.
5. La déclaration des méthodes constantes si elles ne modifient pas les attributs des objets.
6. Le passage par référence pour ne pas copier des objets entiers.
7. Le mapping entre des associations en UML et en C++ (navigabilité, multiplicité, rôle, visibilité).
8. L'implémentation d'une fonction amie d'affichage pour chaque classe, de type :

**friend** std::ostream& **operator**<<(std::ostream& out, **const** Disk & d);

- ① Pour jouer, créez un jeu avec 3 tours (départ, intermédiaire et arrivée), puis ajoutez dans l'ordre au tour de départ 4 disques de diamètre 1, 2, 3 et 4. ☐
- ② Implémentez la méthode `solve()` avec des méthodes supplémentaires qui permettent de résoudre le problème de tour de Hanoï de manière récursive, en affichant l'état de jeu avant et après l'appel de `solve()`, ainsi que les déplacements de disque. ☐