

# TD n°2 – les avions



Le but de ce TP sera de faire un programme pour aider les contrôleurs aérien...

Pour cela, on décide de créer deux classes : l'une qui s'appellera **Aeroport**, et l'autre qui s'appellera **Avion**.

Détaillons la classe **Avion** (à vous de faire le constructeur et le destructeur...) :

- Elle possède 3 attributs `protected` : un **identifiant** (3DF-CSX par exemple), un **type** (Cessna, F-16) et un **etat** (arrêt, décollé, atterri et vol), tous de type **string**.
- Elle possède aussi les fonctions suivantes :
  - o **void changeEtat(string)** : permet de changer l'état de l'avion
  - o **string getEtat()** : permet de récupérer l'état de l'avion
  - o **string getNom()** : permet de récupérer le nom de l'avion
  - o **void description()** : qui affiche toutes les infos sur l'avion

Testez toutes vos fonctions dans un main, et vérifiez que tout marche bien (créez un avion, affichez le, modifiez son état, et enfin réaffichez le...)

Ensuite, attaquons nous à la classe **Aeroport**. Cette classe doit permettre de savoir quels avions sont en attente, et s'il y a des avions en train de décoller ou d'atterrir... Pour faire simple, on va supposer qu'il n'y a qu'une piste d'atterrissage. Je propose la classe suivante, mais si vous voulez rajouter des attributs pour vous faciliter la vie, je ne suis pas contre... Par ailleurs, si vous ne comprenez pas à quoi peut servir un attribut, lisez la partie sur le main, ça pourra peut être vous aider !

Elle possède ces attributs :

- **string nom** : le nom de l'aéroport...
- **Avion \*piste** : permet de pointer sur l'avion qui est sur la piste... S'il n'y a pas d'avion, le pointeur doit être à **NULL** (valeur spéciale de C++), sinon, il doit pointer sur l'avion qui est en train d'atterrir (ou de décoller)...
- **Avion \*garage[10]** : permet de savoir quels avions sont en attente. Les cases du tableau doivent être initialisées à **NULL** dans le constructeur, et dès qu'un avion se pose, il doit être ajouté.
- **int nbAvion** : permet de savoir combien d'avion sont dans le garage.

En plus de ces attributs, la classe possède ces méthodes :

- **void demandeAtterrisage(Avion)** : cette méthode permet à un avion de demander à atterrir. S'il n'y a pas d'avion en train d'atterrir ou en train de décoller, on fait pointer l'attribut **piste** sur l'avion passé en paramètre. Pensez à mettre à jour l'état de l'avion : on met son état à « **atterri** ». Par contre, s'il y a déjà un avion sur la piste, on affiche « Erreur, piste utilisé par » avec la description de l'avion qui utilise la piste, et on ne fait rien de plus.
- **void demandeDecollage()** : cette méthode prend le dernier avion du **garage** et s'il n'y a pas d'avion en train d'atterrir ou en train de décoller, on fait pointer l'attribut **piste** sur le dernier avion du **garage** et on enlève l'avion du tableau **garage**. Pensez à mettre à jour l'état de l'avion : on met son état à « **décolle** ». Par contre, s'il y a déjà un avion, on affiche un message et on ne fait rien de plus.
- **void pisteLibre()** : permet de prévenir la tour de contrôle que l'avion qui utilisait la piste a fini de s'en servir. Pensez à mettre à jour l'attribut piste (à NULL) mais aussi l'état de l'avion : s'il était en train de décoller, on met son état à « **en vol** », et s'il atterrissait, on met son état à « **arrêt** ». Pensez aussi à ajouter l'avion dans le tableau **garage** si l'avion atterrissait !
- **void description()** : permet d'afficher les informations de l'aéroport (quels appareils sont dans le garage, quel appareil est sur la piste...)

Le programme principal...

Commencez par créer 3 avions :

Un **F-18** avec comme identifiant « **ADF-23G** » et comme état « **en vol** ».

Un **concorde** avec comme identifiant « **CCD-56X** » et comme état « **en vol** ».

Un **A-380** avec comme identifiant « **YRF-38G** » et comme état « **en vol** ».

Créez ensuite l'aéroport qui a comme nom « **Marignane** ». Ensuite, déroulez ce scénario :

Le **F-18** demande d'atterrir (l'aéroport doit accepter). Le **concorde** demande juste après d'atterrir, mais l'aéroport doit refuser (normalement, c'est le **F-18** qui utilise la piste).

On signale à l'aéroport que la piste est libre. Le **concorde** redemande d'atterrir, ce que doit accepter l'aéroport. On signale ensuite que la piste est de nouveau libre.

L'aéroport reçoit ensuite une demande de décollage qui doit être acceptée (la piste est libre...). Affichez la description de l'aéroport (1 avion sur la piste : le **concorde**, et 1 avion dans le garage : le **F-18**).

On signale à l'aéroport que la piste est libre (le **concorde** a fini de décoller). L'**A-380** demande alors à atterrir (ce que l'aéroport doit accepter) et on signale que la piste est libre.

Affichez la description de l'aéroport. Il doit y avoir 2 avions dans le garage : le **F-18** et l'**A-380** !