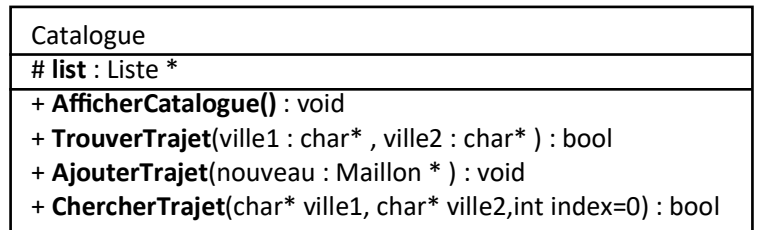
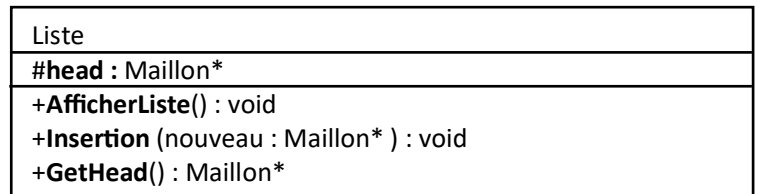
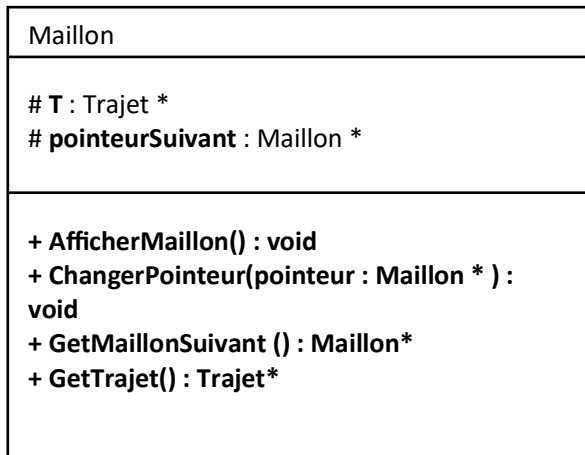
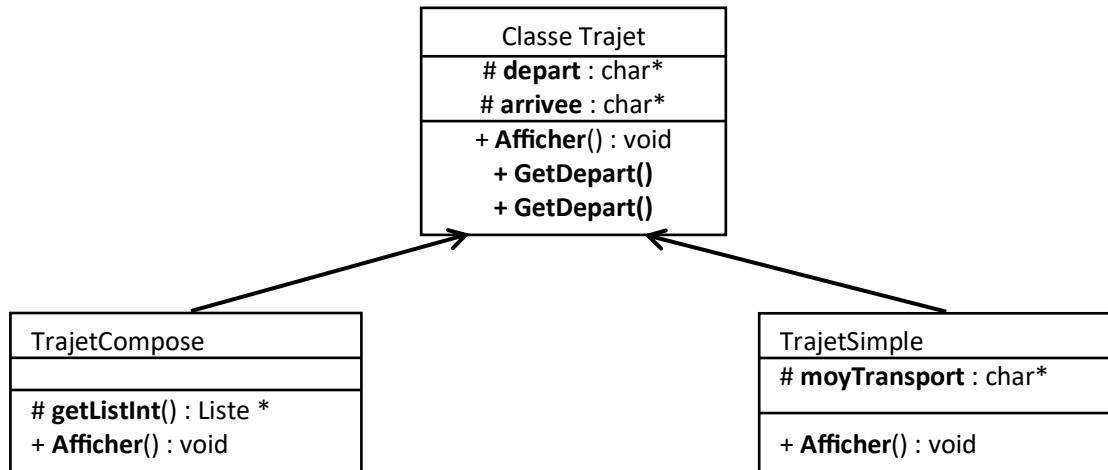


TP C++ n°2 : Héritage – Polymorphisme

I- Graphe d'héritage



+ = public, # = protected

Nous avons choisi de faire de Trajet une classe abstraite en raison des attributs communs entre TrajetComposee et TrajetSimple (ville de départ et d'arrivée), notamment avec la méthode virtuelle pure Afficher().

Nous avons choisi de créer une classe Catalogue dans l'optique d'anticiper d'éventuelles améliorations futures de la fonctionnalité de gestion des trajets. Par exemple, cette conception permettrait d'ajouter des filtres pour les trajets, tels que les coûts d'essence ou les émissions de CO2.

Les classes Liste et Maillon facilitent la gestion d'une liste chaînée de trajets.

Annexe : Description des classes

Classe Trajet :

La classe Trajet représente un trajet entre deux villes. C'est une classe abstraite.

Méthodes publiques :

- **GetDepart() const**: Renvoie le nom de la ville de départ du trajet.
- **GetArrivee() const**: Renvoie le nom de la ville d'arrivée du trajet.
- **Afficher() const = 0**: Afficher est une méthode virtuelle pure qui est donc implémentée dans ses fonctions filles.

Classe TrajetComposee :

La classe TrajetComposee représente un trajet composé, c'est-à-dire un trajet qui est constitué de plusieurs trajets simples.

Méthode publique :

- **Afficher() const**: Affiche les détails du trajet composé en appelant la méthode AfficherListe() de la liste de trajets internes.

Méthode protégée :

- **getListInt()** : Renvoie un pointeur vers la liste de trajets internes.

Classe TrajetSimple :

La classe TrajetSimple représente un trajet direct entre deux villes, spécifiant le moyen de transport utilisé.

Méthode publique :

- **Afficher() const**: affiche les détails du trajet simple, y compris les noms des villes de départ et d'arrivée ainsi que le moyen de transport.

Classe Liste :

La classe Liste représente une liste chaînée de maillons

Méthodes publiques :

- **AfficherListe() const** : Affiche les éléments de la liste.
- **Insertion(Maillon* nouveau)** : Insère un nouveau maillon à la fin de la liste.
- **GetHead() const** : Renvoie le pointeur vers le premier maillon de la liste.

Classe Catalogue :

La classe Catalogue représente un catalogue de trajets, utilisant une liste chaînée de maillons.

Méthodes publiques :

- **AfficherCatalogue() const** : Affiche les éléments du catalogue en appelant la méthode AfficherListe() de la liste associée.
- **TrouverTrajet(char* ville1, char* ville2)** : Recherche un trajet dans le catalogue en spécifiant les villes de départ et d'arrivée. Renvoie true si le trajet est trouvé, sinon false.
- **AjouterTrajet(Maillon* nouveau)** : Ajoute un nouveau trajet à la liste du catalogue.

Classe Maillon :

La classe Maillon représente les maillons de la liste chaînés.

Méthodes publiques :

- **AfficherMaillon() const** : Cette méthode utilise la méthode Afficher() de l'objet Trajet contenu dans le maillon pour afficher les détails du trajet.
- **ChangerPointeur(Maillon* pointeur)** : Cette méthode permet de changer le pointeur suivant du maillon.
- **GetMaillonSuivant() : Maillon*** : Renvoie le pointeur vers le maillon suivant.
- **GetTrajet() const : Trajet*** : Renvoie le pointeur vers l'objet Trajet contenu dans le maillon.