

INTRODUCTION

Vous avez dans Studium un **Projet_Aide_TP2_Listes_Chainees-ListeVoyelles_pointe_ListeAscii.zip**. Ce projet va vous aider à construire aisément la structure de listes demandées pour le TP2. Voir annexe pour la sortie de ce projet.

STRUCTURE DE PROJET EXEMPLE

En annexe vous avez la structure du projet. Ce projet vise à manipuler des **listes chaînées** en C++ pour gérer des mots et leurs voyelles associées. Le projet lit les données à partir de deux fichiers texte :

1. **mots.txt** : Contient une liste de mots.
2. **alpha_ascii.txt** : Contient des caractères et leurs valeurs ASCII.

Objectifs principaux :

1. Créer trois listes chaînées (en parenthèses la relation avec votre TP2):
 - **ListeMots** : Liste contenant les mots lus depuis texte.txt. (**ListeEpice**)
 - **ListeAscii** : Liste contenant tous les caractères et leurs valeurs ASCII. (**ListeFournisseur**)
 - **ListeVoyelles** : Une sous-liste associée à chaque mot, contenant uniquement les voyelles de ce mot via des **pointeurs** vers ListeAscii. (**Liste intermédiaire entre ListeMots et ListeFournisseur qui aura le nom que vous avez décidé dans votre TP2**)
2. Associer chaque mot à ses voyelles spécifiques en utilisant des pointeurs pour éviter la duplication des données ASCII.
3. Afficher chaque mot avec ses voyelles uniques et leurs valeurs ASCII.

GRAPHIQUE DES RELATIONS ENTRE LES 3 LISTES

Voici une représentation textuelle illustrant les relations entre ListeMots, ListeVoyelles et ListeAscii.

1. ListeMots

Représente la liste principale contenant les mots lus depuis le fichier.

ListeMots

Tête -> [simulation] -> [champs] -> [systèmes] -> nullptr

2. ListeAscii

Contient tous les caractères et leurs valeurs ASCII. Chaque voyelle de ListeVoyelles pointe vers un nœud de cette liste.

ListeAscii

Tête -> [A(41)] -> [E(45)] -> [I(49)] -> [O(4F)] -> [U(55)] -> [Y(59)] -> nullptr

3. ListeVoyelles

Chaque mot de ListeMots est associé à une sous-liste contenant ses voyelles via des pointeurs vers ListeAscii.

- Pour le mot "simulation" :

ListeVoyelles (simulation)

Tête -> [Pointeur -> I(49)] -> [Pointeur -> A(41)] -> [Pointeur -> O(4F)] -> nullptr

- Pour le mot "champs" :

ListeVoyelles (champs)

Tête -> [Pointeur -> A(41)] -> nullptr

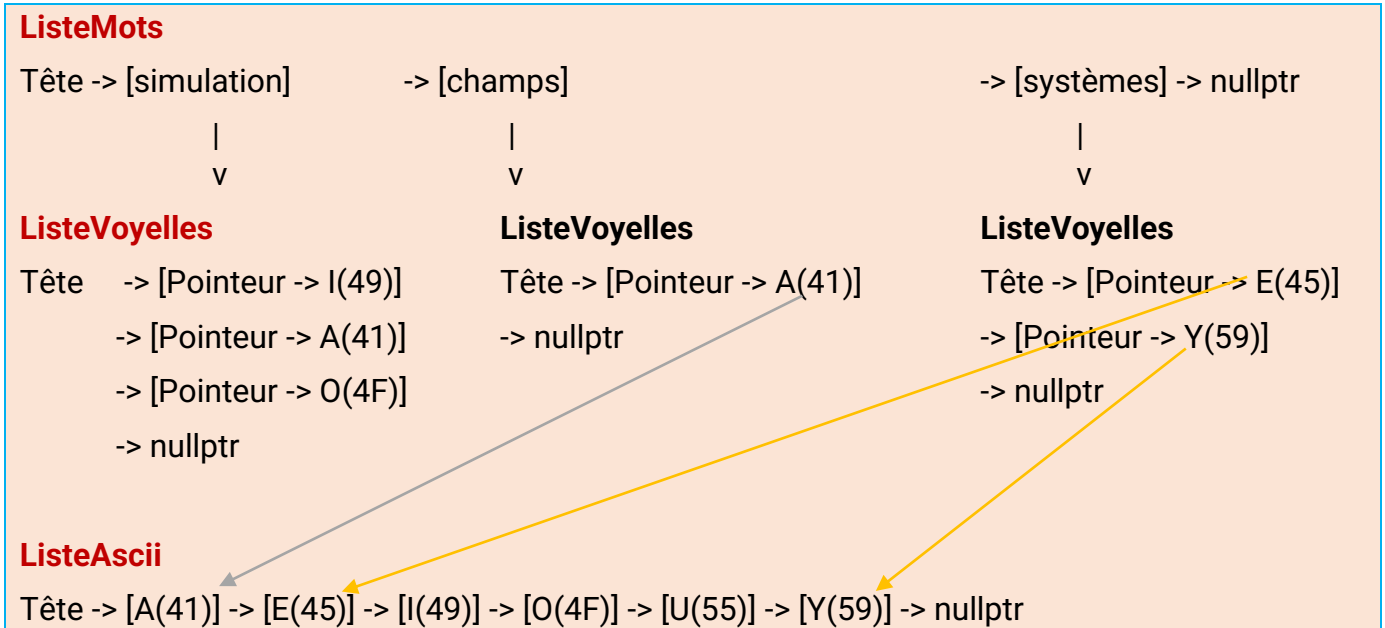
- Pour le mot "systèmes" :

ListeVoyelles (systèmes)

Tête -> [Pointeur -> E(45)] -> [Pointeur -> Y(59)] -> nullptr

Indique que c'est un pointeur sur le nœud de la **ListeAscii** qui contient les valeurs **E** et **45**. Le **E** est le caractère qu'est une voyelle et le **45** est la valeur ASCII en hexadécimal.

RELATION GLOBALE



RÉSUMÉ DES RELATIONS

1. ListeMots :

- Liste principale contenant les mots.
- Chaque nœud pointe vers une liste de voyelles associées.

2. ListeVoyelles :

- Liste secondaire contenant les voyelles d'un mot.
- Chaque voyelle pointe vers un nœud existant dans ListeAscii.

3. ListeAscii :

- Liste globale contenant les caractères et leurs valeurs ASCII.

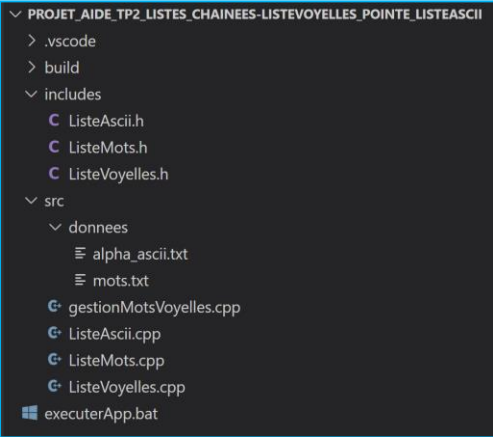
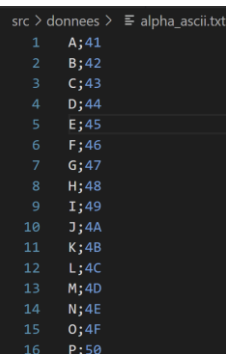
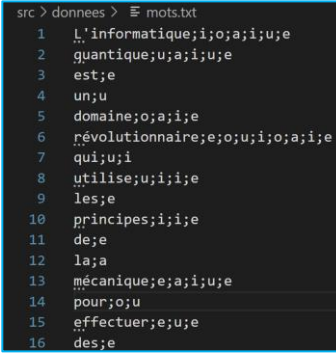
Si vous avez besoin adaptez ce projet à votre TP2.

Bon travail 😊

ANNEXE

STRUCTURE DU PROJET

Ce projet a été conçu en Visual Studio Code sur Windows. Vous pouvez également le faire exécuter sur un MAC (mais pas utiliser le fichier .bat qu'est exclusif à Windows). Vous pouvez l'exécuter selon la plateforme dans laquelle vous êtes d'habitude.

Structure du projet	Fichiers de données
	<div></div> <p>Le fichier de données mots.txt représente votre fichier épices. La donnée dans ListeMots est un string mais vous c'est un objet Epice. Après vous avez les voyelles du mot en question et vous avez les id de fournisseurs de cette épice. Le fichier alpha_ascii.txt représente votre fichier de fournisseur. Comme vous, on doit créer en premier ListeAscii, après créer ListeMots et après associer dans une ListeVoyelles qui a une instance dans la classe ListeMots qu'est la tête de la liste de nœuds de la classe NoeudVoyelles de la classe ListeVoyelles.</p>

Pour l'exécuter tapez : **executerApp**

```
F:\_A24\UDM\IFT1166\MesProjets\Projet_Aide_TP2_Listes_Chainees-ListeVoyelles_pointe_ListeAscii>executerApp
Compilation en cours...
Exécution du programme...
Mot : L'informatique
Voyelles : E(45) U(55) I(49) A(41) O(4F)
Mot : quantique
Voyelles : E(45) U(55) I(49) A(41)
Mot : est
Voyelles : E(45)
Mot : un
Voyelles : U(55)
Mot : domaine
Voyelles : E(45) I(49) A(41) O(4F)
Mot : révolutionnaire
Voyelles : E(45) I(49) A(41) O(4F) U(55)
Mot : qui
Voyelles : I(49) U(55)
Mot : utilise
Voyelles : E(45) I(49) U(55)
Mot : les
Voyelles : E(45)
Mot : principes
Voyelles : E(45) I(49)
```

Sur un autre IDE (interface de développement) faites comme d'habitude.