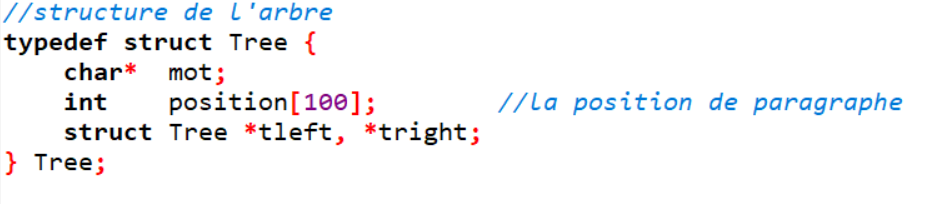
Rapport sur le projet de d'indexation d’un fichier

1. Création de la structure :

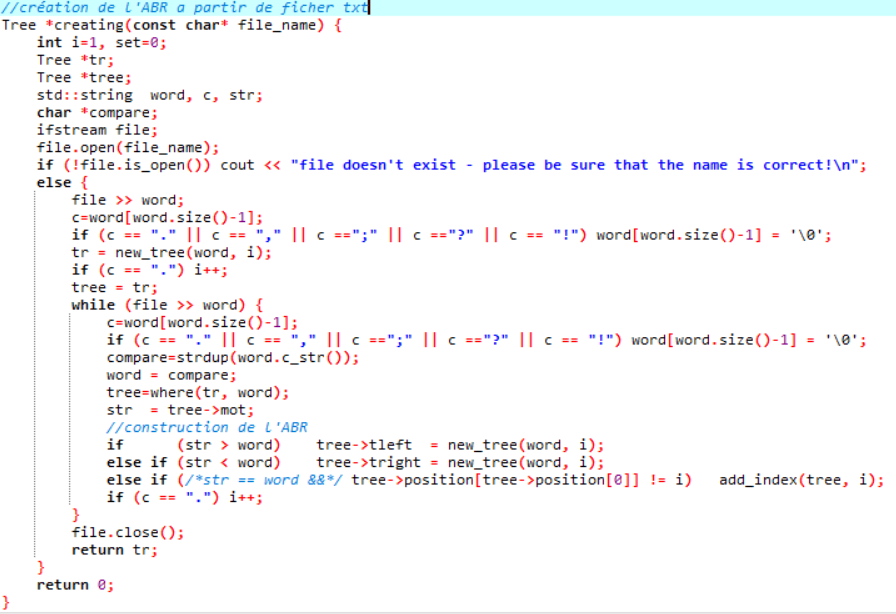
D'abord on va entamer pour créer notre structure de l'arbre.

Elle est composée de 3 champ : le champ mot(c’est pointeur pour faciliter la comparaison),le champ position c’est-à-dire l’index, le champ Tree car l’arbre binaire est composé d’un fils gauche et d’un fils droit.

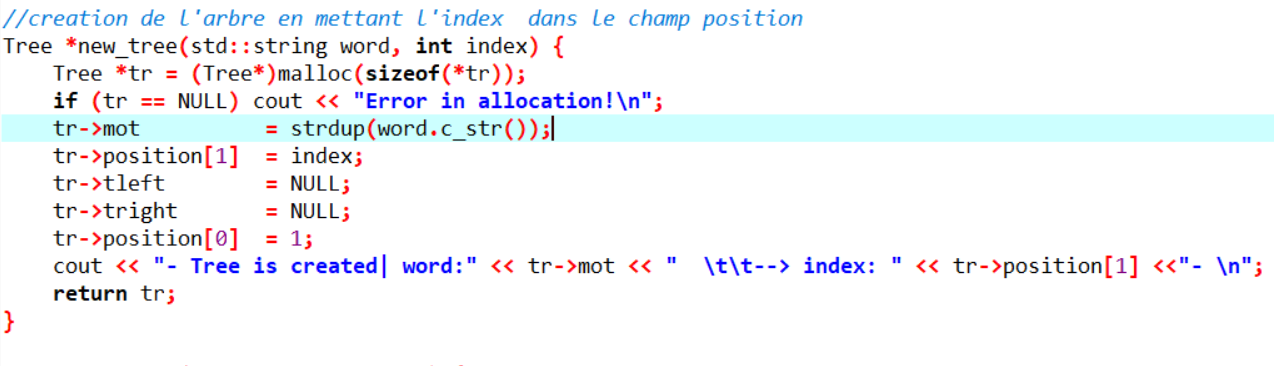


1. Construction de l'arbre de recherche + Création de la liste des positions :

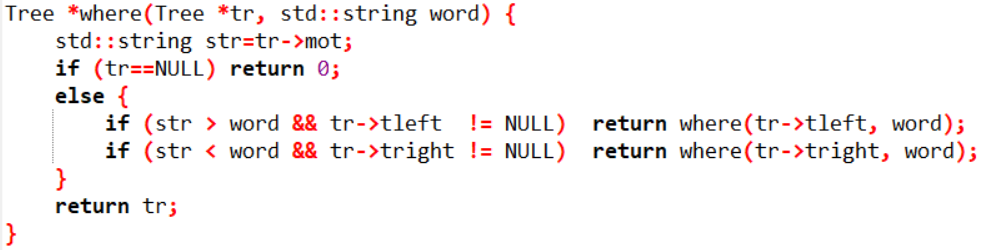
On va deviser cette partie a plusieurs étapes d’abord on va extraire notre ficher (on a supposé que le ficher créer appelé txt et chaque mot ce termine par un point et on va ignorer les points les virgules les points d'exclamation et d’interrogation : (. /,/ ?/ !) ) quand va extraire en mettant chaque mot de l’arbre en utilisant la fonction **new\_tree** **après on va mettre chaque la po(en utilisant fonction qui cherche le mot dans l’arbre crée where)** en utilisant la procédure dans le fichier et puis on compare le code ASCII de chaque mot pour on construire notre ABR en mettant l’indexation (en utilisant l’appelle a la procédure **add\_index.**



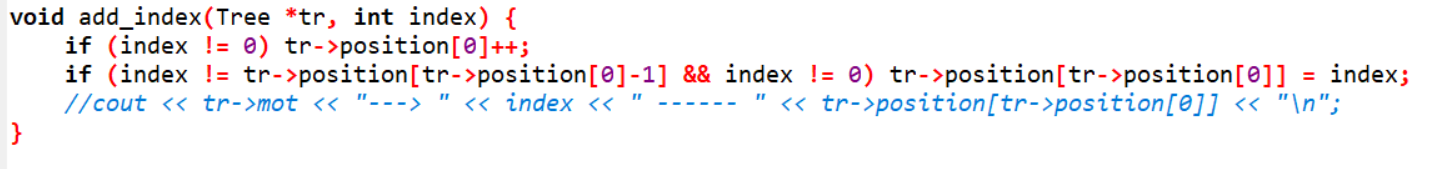
**Fonction new\_tree :**



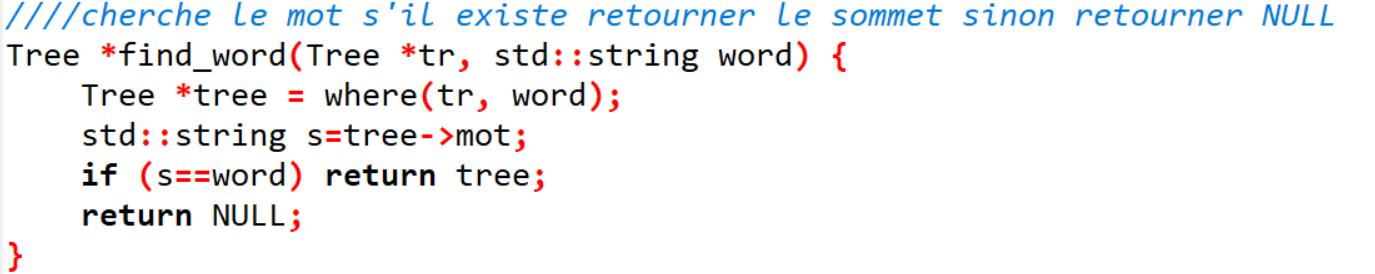
**Fonction where**



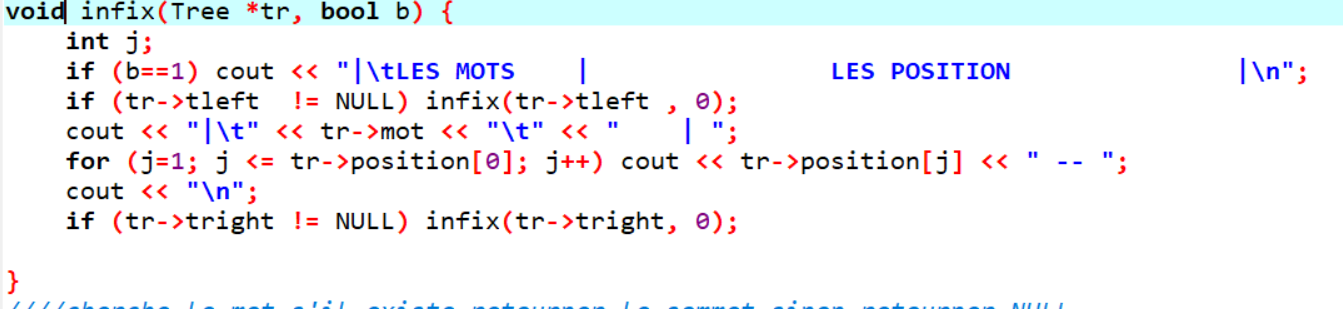
**Procédure add\_index**



On va utiliser aussi la fonction **find\_word** pour cherche si le mot existe ou non :

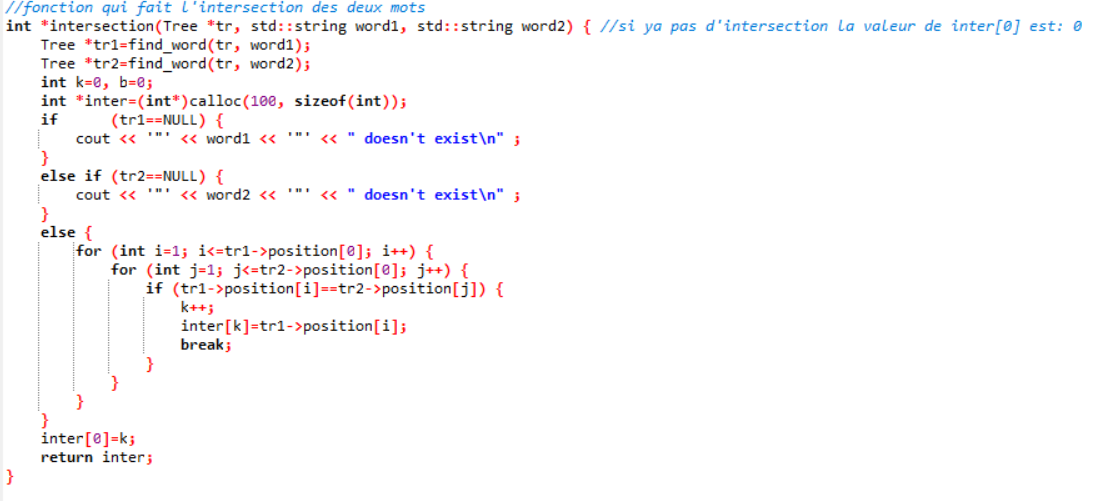


Pour afficher les mots dans l’on va utiliser le parcours infixé pour garder l’affichage par ordre alphabétique avec la procédure **infix :**

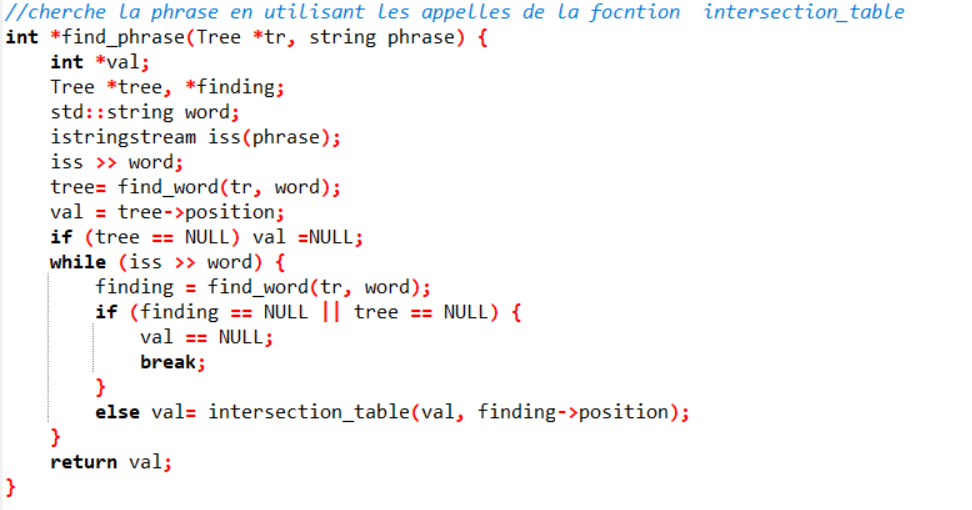


1. **Intersection de deux liste :**

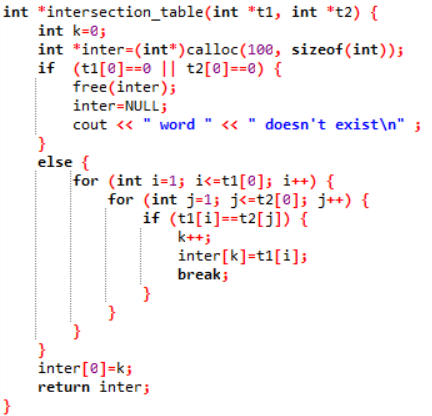
Cette fonction permet de trouver si deux mots on la même position et trouver si un mots existe dans cette liste



1. Chercher une phrase dans le texte :

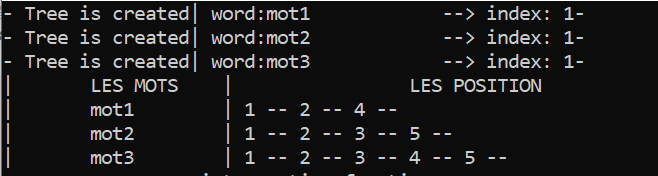
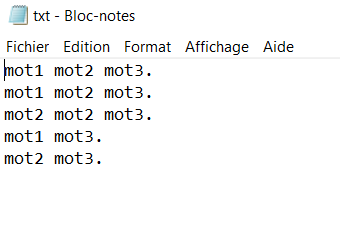


En utilisant l’appelle de fonction **intersection\_table** pour …



exemple d'exécution**:**

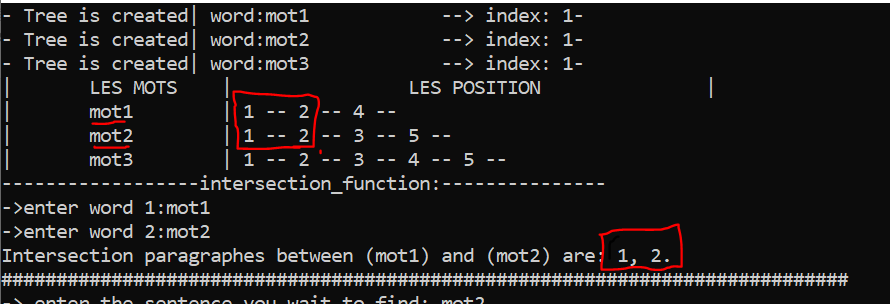
On a supposé que le nom de fichier txt on va remplir notre fichier avec des mots et on va exécuter notre programme :

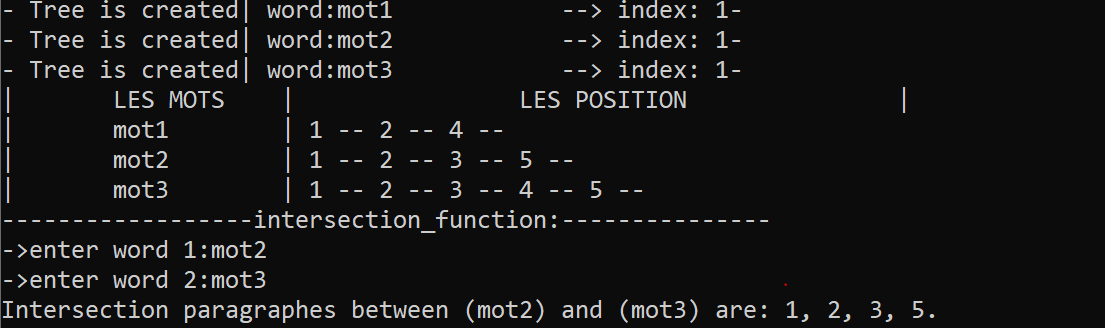


On voit que notre programme a indexé le fichier et mot1 se trouve da les lignes 1,2,4 , mot2 se trouve dans les lignes 1 2,3,5 et le mot3 dans les lignes 1,2,3,4,5

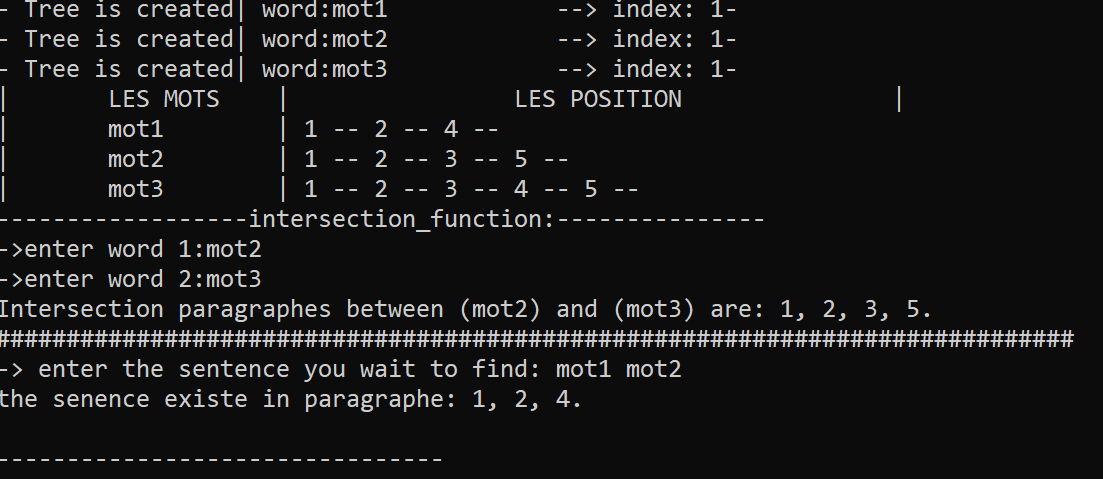
Maintenant on va aller sur l’intersection des deux mots d’abord on doit insérer ces mots puis le programme il va afficher l’intersection :

Ex1 :

Ex2 :

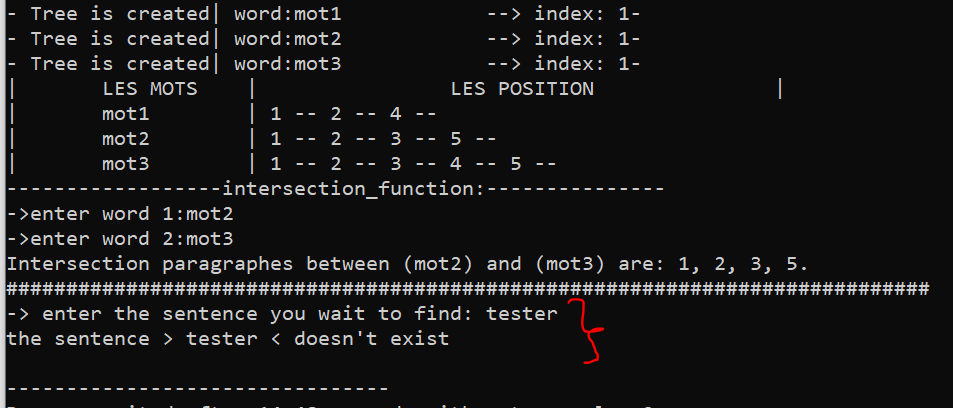


La dernière étape c’est on va chercher une phrase dans le texte : on prend des exemples le 1 er c’est on va la phrase : « mot1 mot2 »



Et on voit que la phrase : « mot1 mot2 » se trouve dans les lignes 1,2,et 4.

Et dans l’exemple 2 on va entrer une phrase n’existe pas dans le fichier et tester notre programme :



Un message apparait que notre phrase « tester » n’existe pas dans notre fichier.