

DM - Correcteur orthographique

Etape 1 : recherche des mots mal orthographié

Fonctions

Listes

- **Cellule * allouer_Cellule(char * mot)** : Alloue une cellule où l'on y met mot et la retourne.
- **int inserer_en_tete(Liste * L, char * mot)** : Insère en entête une nouvelle cellule dans Liste.
- **void liberer_Liste(Liste * L)** : Libère les nœuds alloués dans Liste.
- **void afficher_Liste(Liste L)** : Affiche les mots présents dans Liste.

ATR

- **ATR creer_ATR_vide()** : Alloue un arbre ternaire de recherche vide et la renvoie.
- **void liberer_ATR(ATR * A)** : Libère les _ATR alloués dans ATR.
- **ajoute_branche(ATR * A, char * mot)** : Ajoute une nouvelle branche à ATR contenant le nouveau mot.
- **int inserer_dans_ATR(ATR * A, char * mot)** : Insère dans ATR un nouveau mot.
- **void supprimer_dans_ATR(ATR * A, char * mot)** : Supprime le mot entré en paramètre dans ATR.
- **void afficher_aux(ATR A, char buffer[], int i)** : Affiche les mots présents dans ATR.
- **void afficher_ATR(ATR A)** : Initialise les valeurs buffer et i pour l'appel de la fonction afficher_aux().

correcteur_0

Conclusion

Nous avons rencontré certaines difficultés lors du codage de la fonction supprimer. En effet comme nous ne possédons pas l'algorithme nécessaire pour celle ci, cela nous a pris plus de temps que prévu puisque nous avons du tester et prendre en compte les différents cas possibles des mots à supprimer en testant plusieurs algorithmes différents.

Pour la répartition des tâches nous avons répartis les différentes fonctions à coder, excepté pour la fonction supprimer où l'on a travaillé dessus ensemble. Pour les autres fonctions, nous nous sommes entraïdées quand nous étions bloquées.