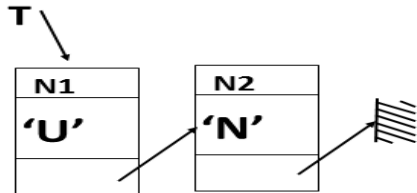


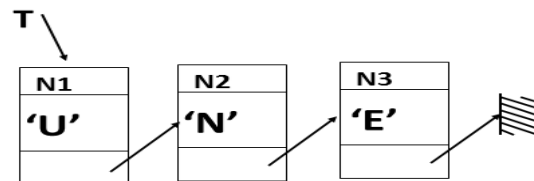
Prof : S.NOUH

Devoir Libre / Série- N°3 : Listes simplement chaînées (Suite)

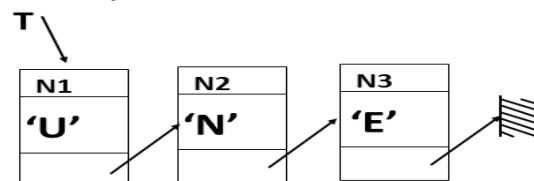
- 1) Déclarer le type struct nœud contenant un caractère et l'adresse de son suivant.
- 2) Redéfinir le type struct nœud * par car.
- 3) Ecrire la fonction car ajouterEnFin(car T, char x) qui ajoute le caractère c en fin de la liste T.
Par exemple si la liste T est :



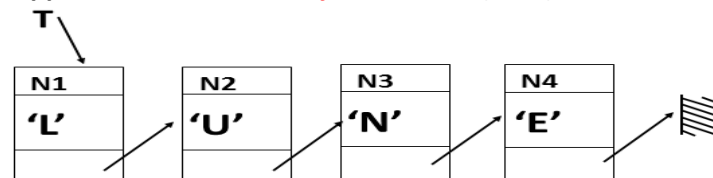
L'appel de la fonction T= ajouterEnFin(T, 'E') retourne la liste suivante :



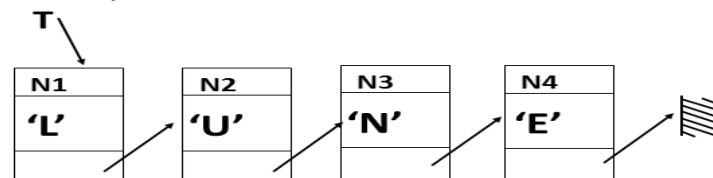
- 4) Ecrire la fonction car ajouterEnTete(car T, char x) qui ajoute le caractère c en tête de la liste T.
Par exemple si la liste T est :



L'appel de la fonction T= ajouterEnTete(T, 'L') retourne la liste suivante :

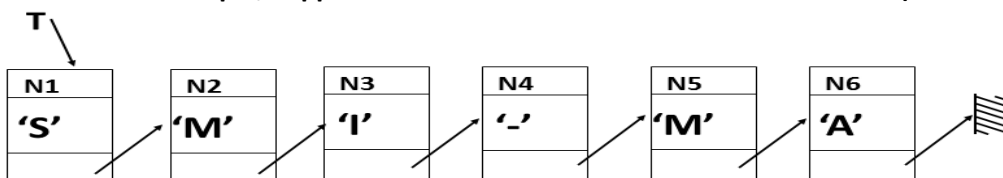


- 5) Ecrire la fonction int longueur(car T) qui compte le nombre d'éléments de la liste T
Par exemple, si T est :

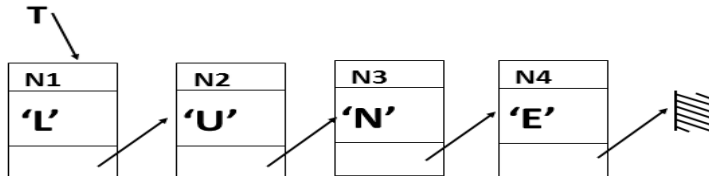


L'appel de la fonction longueur(T) retourne le nombre 4.

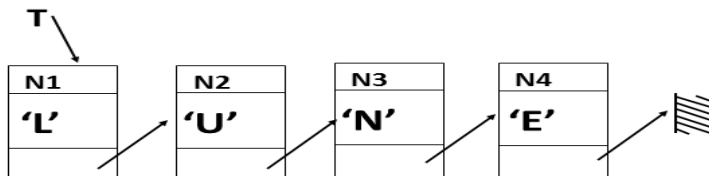
- 6) Ecrire la fonction car ConvertChaineToListeChaine(char X[]) qui convertit la chaîne de caractères X en une liste chaînée. Par exemple, l'appel de la fonction ConvertChaineToListeChaine("SMI-MA") retourne :



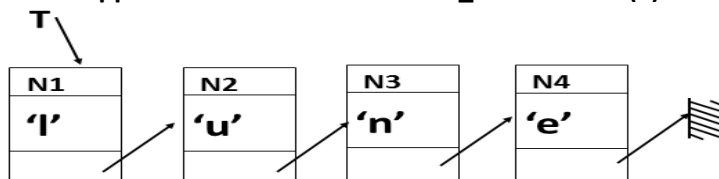
- 7) Ecrire la fonction `char* ConvertListeChaineToChaine(car T)` qui convertie la liste chaînée T en une chaîne de caractères X. (l'opération inverse de la fonction `car ConvertChaineToListeChaine(char X[])`)
- 8) Ecrire la fonction `car InverseListeChaine (car T)` qui retourne la liste chaînée représentant l'inverse de T. Par exemple Si T contient les caractères A,B,C,D,E dans cet ordre. La liste retournée contiendra E,D,C,B,A dans cet ordre.
- 9) Ecrire la fonction `int Existe(char x, car T)` qui retourne 1 si x est dans T, elle retourne 0 sinon. Par exemple, si T est la liste suivante, alors `Existe('n', T)` retourne 0 mais `Existe('N', T)` retourne 1.



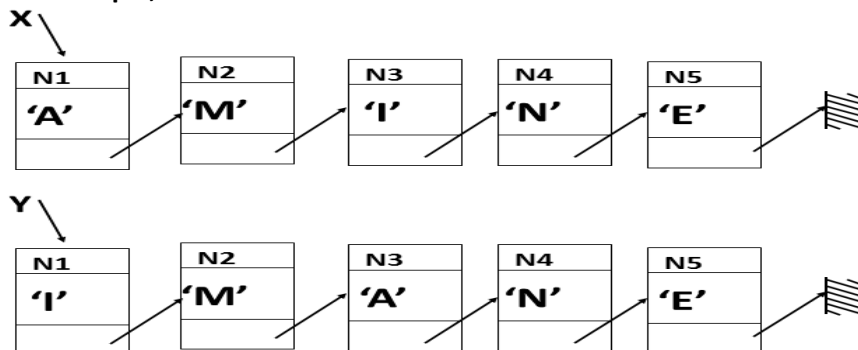
- 10) Ecrire la fonction `void Minuscule_ListeChaine(car T)` qui met en minuscule tous les caractères alphabétiques de la liste chaînée T. Les autres éléments ne seront pas modifiés. Par exemple, Si T est :



Alors l'appel de la fonction `Minuscule_ListeChaine(T)` rend T comme suit :



- 11) Ecrire la fonction `int ComparerListesChaineesOrdreRespecte(car A, car B)` qui retourne 1 si tous les éléments de A sont dans B et dans le même ordre et tous les éléments de B sont dans A. Par exemple, Si X et Y sont :



L'appel de la fonction `ComparerListesChaineesOrdreRespecte(X,Y)` retourne 0.

- 12) Ecrire la fonction `int ComparerListesChaineesOrdreQuelconque(car A, car B)` qui retourne 1 si tous les éléments de A sont dans B et tous les éléments de B sont dans A dans un ordre quelconque. Pour les deux listes X et Y de la question précédente, L'appel de la fonction `ComparerListesChaineesOrdreQuelconque (X,Y)` retourne 1.