

Programmation C : TD 2

Ibrahim ALAME

18/10/2023

1. On utilisera le type : `typedef char mot[NMAX]` où `NMAX` est une constante entière préalablement définie.
 - (a) Écrire la fonction `longMot` qui retourne la longueur d'une chaîne de caractères comme résultat.
 - (b) Écrire la fonction `min2maj` qui convertit toutes les lettres d'une chaîne en majuscules.
 - (c) Écrire la fonction `ajouteMot` à deux paramètres `mot1` et `mot2` qui copie la chaîne de caractères `mot2` à la fin de la chaîne `mot1`.
 - (d) Écrire et tester une fonction `void miroir(mot dest, mot src)` écrivant à l'envers un mot donné.
 - (e) Écrire et tester une fonction `int palindrome(mot src)` qui teste si un mot donné est un palindrome.

2. On utilisera les types suivants :

```
typedef char mot[NMAX]; // adresse d'une chaîne de caractère de longueur
                        // maximal NMAX = 50
typechar mot txt[TXTMAX]; // adresse d'un tableau de mots de longueur
                        // maximal TXTMAX = 100 dont le dernier valeur est "fin"
```

- (a) Écrire une fonction `void afficher(txt t)` affichant à l'écran les mots du texte `t`.
 - (b) Écrire une fonction `int appartient(mot m, txt t)` testant l'appartenance du mot `m` au texte `t`.
 - (c) Écrire une fonction `int ajouter(mot m, txt t)` qui ajoute au texte `t` et retourne son indice dans le tableau (ou -1 si `m=NULL`).
 - (d) Écrire une fonction `int supprime(mot m, txt t)` qui supprime au texte `t` le mot `m`.
3. Codage : On choisit un décalage (par exemple 5), et un `a` sera remplacé par un `f`, un `b` par un `g`, un `c` par un `h`, etc ... On ne cryptera que les lettres sans toucher ni à la ponctuation, ni à la mise en page (caractère blanc, etc.).
 - (a) Déclarer un tableau de caractères `cMessage` initialisé avec le message en claire.
 - (b) Écrire une procédure `crypte` de cryptage d'un caractère qui sera passé par adresse.
 - (c) Écrire le programme principal `main` qui activera `crypt` sur l'ensemble du message et imprimera le résultat.