

## DS pratique en langage C

### I- Écrire les fonctions suivantes: (3\*4pts)

1. **int strlen(char \* str)** : renvoie la taille de la chaîne de caractère **str**
2. **int compareStr(char\* str1, char\* str2)**: renvoie
  - a. -1 si **str1** précède **str2** dans un dictionnaire
  - b. 1 si **str1** précède **str2** dans un dictionnaire
  - c. 0 si **str1 == str2**
3. **char \* concatStr( char\* str1, char\* str2)**: renvoie la concaténation des deux chaînes de caractère **str1** et **str2**.
4. **char \* subStr( char\* str, int a, int b)**: renvoie la sous chaînes de caractère **str[a]...str[b-1]**

### II- Ecrire la fonction **void main(void)** pour tester ces trois fonctions (4\*2pts)

1. Demande à l'utilisateur deux chaînes de caractères
2. Affiche la taille de ces deux chaînes
3. Compare les deux chaînes
4. Affiche la concaténation de ces deux chaînes

#### NB:

1. On suppose que la taille maximale d'une chaînes est 10
  2. Avec **scanf()** le caractère '**\0**' est inserer automatiquement à la fin d'une chaine de caractère
- 

## DS pratique en langage C

### I- Écrire les fonctions suivantes: (3\*4pts)

1. **int strlen(char \* str)** : retourne la taille de la chaîne de caractère **str**
2. **int compareStr(char\* str1, char\* str2)**: retourne
  - a. -1 si **str1** précède **str2** dans un dictionnaire
  - b. 1 si **str1** précède **str2** dans un dictionnaire
  - c. 0 si **str1 == str2**
3. **char \* concatStr( char\* str1, char\* str2)**: retourne la concaténation des deux chaînes de caractère **str1** et **str2**
4. **char \* subStr( char\* str, int a, int b)**: renvoie la sous chaînes de caractère **str[a]...str[b-1]**

### II- Ecrire la fonction **void main(void)** pour tester ces trois fonctions (4\*2pts)

1. Demande à l'utilisateur deux chaînes de caractères
2. Affiche la taille de ces deux chaînes
3. Compare les deux chaînes
4. Affiche la concaténation de ces deux chaînes

#### NB:

1. On suppose que la taille maximale d'une chaînes est 10
2. Avec **scanf()** le caractère '**\0**' est inserer automatiquement à la fin d'une chaine de caractère