

Corrigé série 8.3 : Les chaînes de caractères

Exercice 1 : Inversion de mots [FACILE]

Ecrire un programme qui lit 5 mots, séparés par des espaces et qui les affiche ensuite dans une ligne, mais dans l'ordre inverse. Les mots sont mémorisés dans 5 variables M1, ... ,M5.

Exemple :

```
voici une petite phrase !  
! phrase petite une voici
```

```
#include <stdio.h>

int main()
{
    char M1[30], M2[30], M3[30], M4[30], M5[30];
    printf("Entrez 5 mots, séparés par des espaces :\n");
    scanf ("%s %s %s %s %s", M1, M2, M3, M4, M5);
    printf("%s %s %s %s %s\n", M5, M4, M3, M2, M1);

    return 0;
}
```

*Note:utiliser la fonction **fgets()** pour lire une ligne de texte comportant des espaces

Exercice 2 : Recherche de caractère [MOYEN]

Écrire un programme qui lit une ligne de texte (taille max. 50 caractères) et puis un caractère. Le programme doit indiquer si le caractère se trouve dans le texte. Si oui, il doit afficher à quelle position il se trouve (indice de la première occurrence).

```
#include <stdio.h>
#include <string.h>
#define N 50

int main()
{
    char texte[51];
    char caractere;
    int i;

    /* Saisie des données */
    printf("Texte (max.%d) : ",N);
    fgets(texte,50,stdin);           // saisie sécurisée
    fflush(stdin);                  // si saisie > 50 caractères
    printf("Caractere recherche: ");
    scanf(" %c",&caractere);

    /* RECHRCHE */
    i=0;
    while( texte[i]!=caractere && texte[i]!='\0')
    {
        i++;
    }

    /* Affichage methode 1 */
    if (i<strlen(texte))
        printf("\nCaractere trouve a la position %d",i);
    else
        printf("\nCaractere pas trouve\n");

    /* Affichage methode 2 */
    if (texte[i]==caractere)
        printf("\nCaractere trouve a la position %d",i);
    else
        printf("\nCaractere pas trouve\n");

    return 0;
}
```

Exercice 3 : Inversion de lettres [MOYEN-AVANCÉ]

Écrire un programme qui lit une ligne de texte* (taille max. 200 caractères) la mémorise dans une variable `texte` et affiche ensuite :

- a) la longueur `lChaine` de la chaîne.
- b) le nombre de 'e' contenus dans le texte.
- c) toute la phrase à rebours, **sans changer le contenu de la variable** `texte`.
- d) toute la phrase à rebours, **après avoir inversé** l'ordre des caractères dans `texte`:

voici une petite phrase !
! esarhp etitep enu iciov

```
#include <stdio.h>

int main()
{
    /* Déclarations */
    char texte[201];      /* chaîne donnée          */
    int i, j;             /* indices courants          */
    int lChaine;          /* longueur de la chaîne     */
    int c;                /* compteur des lettres 'e'  */
    int aide;             /* pour l'échange des caractères */

    /* Saisie des données */
    printf("Entrez une ligne de texte (max.200 caractères) :\n");
    gets(texte);

    /* a) Compter les caractères */
    lChaine = 0;
    while (texte[lChaine] != '\0')
        lChaine++;
    printf("Le texte est composé de %d caractères.\n", lChaine);

    /* b) Compter les lettres 'e' dans le texte */
    c=0;
    for (i=0; texte[i]; i++)
        if (texte[i]=='e') c++;
    printf("Le texte contient %d lettres 'e'.\n", c);

    /* c) Afficher la phrase à l'envers */
    for (i = lChaine-1; i >= 0; i--)
        putchar(texte[i]);          /* ou printf("%c",texte[i]); */
    putchar('\n');                  /* ou printf("\n"); */

    /* d) Inverser l'ordre des caractères */
    for (i=0, j=lChaine-1 ; i < j ; i++,j--)
    {
        aide = texte[i];
        texte[i] = texte[j];
        texte[j] = aide;
    }
    puts(texte);                    /* ou printf("%s\n", texte); */

    return 0;
}
```

Exercice 4 : Suppression d'espaces [AVANCÉ]

Écrire un programme qui lit une ligne de texte* (taille maximale: 128 caractères), puis qui supprime tous les espaces dans la chaîne et tasse les éléments restants.

Exemple:

```
Entrez un texte : Il fait beau
Resultat :Ilfaitbeau
```

Indication : Commencez par faire l'exercice sur papier. Essayez de raisonner d'abord avec deux chaînes de caractères, puis avec une seule.

```
#include <stdio.h>
#include <string.h>

#define TAILLEMAX 128

int main
{
    char texte[TAILLEMAX]; /* chaîne de caractères */
    int i, j;

    /* Saisie des données */
    printf("text (max.%d) : ", TAILLEMAX);
    fgets(texte, TAILLEMAX, stdin);
    puts(texte);
    putchar('\n');

    /* Effacer les espaces et compresser : */
    for (i=0, j=0 ; i<strlen(texte)+1 ; i++)
    {
        if (texte[i]!=' ')
        {
            texte[j] = texte[i];
            j++;           //incrément conditionnel!
        }
    }

    puts(texte);          // affiche le texte sans espace

    putchar('\n');
    return 0;
}
```

Exercice 5 : Conversion chaine à long

[AVANCÉ]

Écrire un programme qui lit une ligne de texte* (taille maximale: 128 caractères), y récupère un chiffre (positif ou négatif), le mémorise dans une variable de type long et l'affiche.

Exemple :

```
Entrez un texte : En l'an de grâce 1748 il y eut...
Valeur récupérée: 1748
```

Note : on supportera les formats simples comme "1748", "+1748", "-1748".

```
#include <stdio.h>
#include <stdbool.h>      // pour bool, true et false
int main()
{
    char texte[128];
    int i=0;
    bool estNegatif=false;

    /* Saisie des données */
    printf("Entrer une phrase (max.%d caracteres): ", 128);
    fgets(texte, 128, stdin);
    printf("\n");

    /* Recherche du premier symbole d'un chiffre */
    while ((texte[i]<'0' || texte[i]>'9') && // pas un chiffre
           texte[i]!='\0' &&                // chaine pas terminée
           texte[i] != '-'                  // pas '-' (chiffre négatif)
        )
    {
        i++;
    }

    /* Marquage des chiffres négatifs */
    if ( texte[i] == '-' )
    {
        estNegatif = true;
        i++;                // on passe au premier chiffre
    }
    Ou, pour éviter l'approche : « si oui, oui sinon non »
    estNegatif = (texte[i] == '-');
    if (estNegatif)i++ ;

    /* Calcul de la valeur du chiffre entier */
    long valeur=0;
    while ( (texte[i]>='0') && (texte[i]<='9') )
    {
        valeur *=10;        // chiffres précédents décalés d'une colonne
        valeur += texte[i] - '0' ; // avec '0' --> +0 , avec '1'-->+1, ...
        i++;
    }
    if (estNegatif)
        valeur *= -1;
```

```
/* Affichage */  
printf("\n\n La valeur saisie est: %ld\n\n",valeur);  
}
```