

Cours de soutien de C (pour Introduction au Système)

**Planche d'exercices n°2
(donnée le 7/9 - à faire pour la semaine suivante)**

Exercice 1. Saisir le code ci-dessous dans un fichier exo1.c

```
#include <stdio.h>
#define N 250

int main(void) {
    char tabl[N];
    int i;
    int compteur;
    for (i=0; i < N; i++)
        tabl[i] = 'k';
    for (i=0; i < N; i++) {
        if (tabl[i] == 'k') compteur++;
    }
    printf("Compteur = %d\n", compteur);
    return 0;
}
```

Compilez exo1.c puis exécutez. Que vous attendez-vous à avoir comme résultat à l'écran ? Obtenez-vous cela ? Pourquoi, et quelle conclusion importante en tirez-vous ?

Corrigez ce programme pour qu'il se comporte comme attendu.

Exercice 2. Saisir le code ci-dessous dans un fichier exo2.c

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

char *tableau_rempli (char c) {
    char tabl[N];
    int i;
    for (i=0; i < N; i++)
        tabl[i] = c;
    return tabl;
}

int main(void) {
    char x;
    char *t;
    int j;
    printf("Quel caractere met-on dans le tableau ? ");
    scanf("%c", &x);
    t = tableau_rempli(x);
    printf("Voici le tableau rempli : \n");
    printf(" [ ");
    for (j=0; j < N; j++) {
        printf("%c ", t[j]);
    }
    printf(" ] \n");
    return 0;
}
```

Compilez exo2.c puis exécutez. Que vous attendez-vous à avoir comme résultat à l'écran ? Obtenez-vous cela ? Pourquoi ?

Corrigez la fonction `tableau_rempli` pour obtenir le comportement attendu.

Exercice 3. Ecrire les fonctions de conversion suivantes :

- une fonction `convert_vers_bin` qui renvoie en résultat un vecteur de caractères représentant le codage binaire d'un nombre entier positif `x`, sur un nombre donné `n` de bits :

```
char *convert_vers_bin(int x, int n);
```

- une fonction `convert_vers_dec` qui renvoie en résultat un nombre entier positif correspondant à un vecteur de bits (chaque bit étant représenté par le caractère '0' ou '1') d'une taille donnée :

```
int convert_vers_dec(char v[], int n);
```

On prendra comme convention que les bits sont représentés du poids faible vers le poids fort. On ne devra pas utiliser de fonction puissance.

On définira de plus une fonction pour l'affichage d'un vecteur de bits (sous forme de caractères), afin d'afficher le résultat de `convert_vers_bin`.

Votre fichier contiendra au préalable la définition (par `#define`) d'une valeur `N` pour le nombre de bits souhaité pour la représentation des vecteurs (fixée par exemple à 8) qui servira de paramètre effectif pour le paramètre formel `n`.

Ecrire le programme qui saisit au clavier un nombre entier positif et affiche à l'écran sa valeur en binaire, puis qui saisit au clavier une suite de caractères '0' et '1' représentant une valeur binaire donnée à partir du bit de poids faible et qui calcule et affiche la valeur décimale correspondante. On aura par exemple à l'exécution :

```
-- Essai de conversion decimal -> binaire --
Saisir la valeur a convertir en binaire : 83
Resultat converti :
0 1 0 1 0 0 1 1
-- Essai de conversion binaire -> decimal --
Saisir la chaine a convertir en decimal (poids faible -> fort) : 11001010
Valeur convertie = 83
```