

TP 1 : STRUCTURES DE CONTRÔLE

On veut faire afficher un damier de n par n cases. Les cases noires (resp. blanches) sont représentées par $p * p$ un caractère $C1$ (resp. $C2$). Par exemple pour $n=4$, $p = 3$, $C1=\&$, $C2=+$, le programme doit afficher :

```
&&&+++&&&+++
&&&+++&&&+++
&&&+++&&&+++
+++&&&+++&&&
+++&&&+++&&&
+++&&&+++&&&
&&&+++&&&+++
&&&+++&&&+++
&&&+++&&&+++
+++&&&+++&&&
+++&&&+++&&&
+++&&&+++&&&
```

Les nombres n et p , les caractères $C1$ et $C2$ sont choisis par l'utilisateur (n et p sont supposés être inférieurs ou égaux à 9).

Question 1 : On écrira ce programme de telle façon que :

- l'entrée des données soit "sécurisée" en ce sens qu'en cas de faute de frappe lors de l'acquisition des nombres n et p , le programme ne s'arrête pas mais qu'au contraire il demande à nouveau les valeurs n ou p (par exemple on frappe sur la touche "U" au lieu de la touche "8"). De façon similaire, on s'assurera que le nombre n est pair et que les caractères $C1$ et $C2$ sont différents.
- après chaque affichage, le programme "recommence". Il s'arrêtera lorsque la valeur donnée à n est nulle.

Question 2 :

Ecrire trois versions de la partie affichage du damier proprement dit n'utilisant que des structures itératives ("pour", "tant que", "jusqu'à ce que"). L'exécution de ces trois versions sera contrôlée par un menu interactif. Par exemple sur l'écran s'affichera le teste suivant :

Voulez-vous afficher le damier :

- en utilisant la structure pour (réponse a ou A)
- en utilisant la structure tant que (réponse b ou B)
- en utilisant la structure pour (réponse c ou C)
- acquérir de nouvelles valeurs pour n , p , $C1$ et $C2$ (réponse d ou D)

Si la réponse est a, b ou c le damier sera affiché puis le menu réapparaîtra. Si la réponse est d on redemandera les paramètres du problème puis le menu réapparaîtra. Si la réponse est autre, le menu réapparaîtra.

Indications : l'impression se fait caractère par caractère. Pour ce faire on peut utiliser soit la fonction `printf()` soit la fonction `putchar()`. Par exemple pour imprimer la lettre 'a', on peut écrire :

- Soit `putchar ('a');`
- Soit `printf("%c", 'a');`

TP 2 : TABLEAUX A UNE DIMENSION

Exercice 1 :

- a. Saisir une chaîne de caractères.
- b. Après la saisie, calculer (sans utiliser de fonction) et afficher la longueur de la chaîne de caractères.

Exercice 2 :

- a. Saisir, caractère par caractère, une chaîne composée exclusivement de chiffres et afficher la chaîne saisie. Le programme vérifiera que chaque caractère entré au clavier est bien un chiffre compris entre 0 et 9.
- b. Convertir le tableau de caractères en un tableau d'entiers et afficher ce tableau.
- c. Convertir le nombre entré sous forme de chaîne de caractères en nombre entier et l'afficher.

Exercice 3 :

- a. Saisir un nombre entier positif.
- b. Convertir ce nombre en une chaîne de caractères.
- c. Afficher la chaîne convertie.

Exercice 4 :

- a. Saisir deux chaînes de caractères
- b. Comparer, caractère par caractère, ces deux chaînes et déterminer si elles sont identiques ou leur ordre lexicographique.

Exercice 5 :

Un palindrome est un mot qui reste le même qu'on le lise de gauche à droite ou de droite à gauche (par exemple, PIERRE n'est pas un palindrome, alors que OTTO est un palindrome).

- a. Ecrire un programme qui vérifie si une chaîne simple (sans espace) introduite au clavier est un palindrome.
- b. Faire la même chose, mais cette fois avec une phrase complète (avec espaces et/ou ponctuation).

Exercice 6 :

Refaire les exercices précédents avec les fonctions de la bibliothèque « `stdlib.h` » et « `string.h` ».

