DEUG MIAS IF121, groupe D15
Contrôle continu n° 2

Sujet conçu par Berke Durak et Gilles Peskine

13 décembre 2002

durée : 1 heure

Barème indicatif : 5 points pour chaque partie.

L'usine Un fabricant d'accessoires de mode confectionne, ou plutôt fait confectionner à des ouvrières sous-payées et non syndiquées, des colliers par une méthode brevetée qui consiste à enfiler successivement sur un fil des perles de différentes couleurs.

Nombre de perles par collier Le nombre M de perles qui vont sur un collier dépend de la taille des perles et de la longueur du fil, données qui peuvent changer en fonction de l'évolution des marchés, mais qui sont fixées entre deux vagues de licenciements :

```
const int M = 5;
```

Les couleurs Les perles viennent en quatre couleurs, chaque couleur étant représentée par un entier comme suit : 0 pour le noir, 1 pour le rouge, 2 pour le vert et 3 pour le bleu.

Les colliers On peut distinguer les deux bouts du collier par leur boucle de fixation : l'une s'ouvre et l'autre pas. Par conséquent un collier peut se décrire par la suite ordonnée des couleurs de ses perles, en commençant par le bout qui s'ouvre :

```
typedef int collier[M];
```

L'entête L'entête de votre programme doit être comme suit :

```
#include <iostream.h>
#include <stdlib.h>

const int M = 5;
typedef int collier[M];
```

1 Partie programmation

Entrées et sorties

Exercice 1. *Écrivez une fonction* `void lire_collier(collier c)` *qui lit M entiers en les plaçant dans l'ordre dans le collier c. Si jamais il rencontre un entier ne correspondant pas à une couleur valide, elle doit afficher le message* `MAUVAISE COULEUR` *et utiliser l'instruction*

```
exit(1);
```

qui termine votre programme.

Exercice 2. *Écrivez une fonction* `void afficher_collier(collier c)}` *qui affiche le collier* `c` *comme suite de couleurs. Par exemple si* `c` *est le tableau 3 1 0 1 0 votre fonction devra afficher*

```
bleu rouge noir rouge noir
```

Statistiques

Exercice 3. *Ajoutez à la fin de l'entête la déclaration*

```
typedef int statistiques[4];
```

1. *Écrivez une fonction* `void compter(collier c, statistiques s)` *qui place chaque couleur représentée par l'entier* $i \in \{0, 1, 2, 3\}$ *le nombre de perles de couleur* i *dans le collier* c *dans* $s[i]$.

   Attention, lorsqu'on appelle votre fonction, il se peut que s *contienne des valeurs bizarres. Il ne faut donc pas oublier d'initialiser chaque case de* s *dans votre fonction.*

2. *Écrivez une fonction* `int nombre_de_couleurs(collier c)` *qui retourne le nombre de couleurs du collier. Par exemple si le collier n'est fait que de perles bleues, ce sera 1, s'il comporte au moins une perle de chacune des quatre couleurs, ce sera 4.*

2 Partie théorique

Colliers palindromiques On peut mettre un collier dans deux sens : soit le bout qui s'ouvre est du côté du bras gauche derrière le cou, soit il est du côté du bras droit.

Un collier qui a la même apparence quel que soit le sens dans lequel on le met s'appelle un *collier palindromique*. Par exemple les colliers

`noir rouge vert vert rouge noir`

et

`rouge vert bleu vert rouge`

sont palindromiques.

Formellement, un collier de longueur M (entier positif quelconque) peut être vu comme un tableau $c[0] \ldots c[M-1]$. Le collier sera donc palindromique si et seulement si

$$\forall i \ \ 0 \leq i < M \Longrightarrow c[i] = c[M-1-i].$$

Exercice 4.

1. *Donnez* en pseudo-code *un algorithme (une fonction) pour décider si un collier donné par le tableau des entiers de ses couleurs est palindromique ou non.*

2. *Donnez la preuve de correction de votre algorithme.*

   Indication : il pourra être utile de distinguer dans une partie de la preuve le cas où M *est pair et le cas où* M *est impair.*