

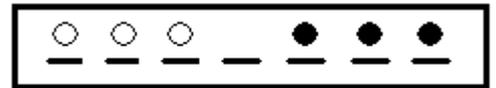
Remarque préliminaire pour remettre vos fichiers

- a) Créer un répertoire sur le disque dur local (D:) avec votre nom et prénom.
- b) Dans ce répertoire, créer trois sous-répertoires PBI, PBII et PBIII pour les trois problèmes.
- c) Sauvegarder vos fichiers sources et exécutables dans les sous-répertoires correspondants.

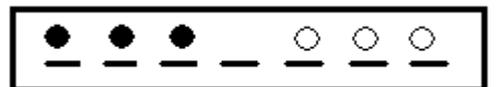
Problème I - Shuttle puzzle

40 points

On considère p boules blanches et p boules noires placées sur un support avec 2p+1 emplacements. Au début, les p boules blanches se trouvent à gauche et les p boules noires se trouvent à droite, laissant l'emplacement du milieu vide (dans notre exemple ci-contre: p = 3).

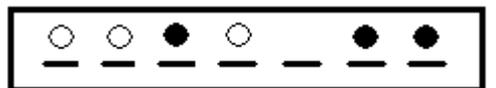
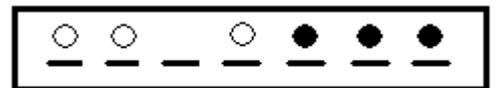


Les boules blanches se déplacent seulement vers la droite, les boules noires seulement vers la gauche. A la fin, toutes les boules blanches doivent se retrouver à droite et toutes les boules noires à gauche.



Pour y arriver, deux déplacements différents sont autorisés:

- 1) glissement d'une boule adjacente à l'emplacement vide dans l'emplacement vide (l'image montre le glissement d'une boule blanche);
- 2) saut d'une boule adjacente à une boule de couleur opposée dans l'emplacement vide (l'image montre le saut d'une boule noire).



Pour décrire le jeu, nous utilisons deux lettres pour la couleur des boules, B et N ainsi que le symbole souligné (_) pour l'emplacement vide.

Ecrire un programme qui lit la valeur de p au clavier et qui affiche, pas par pas, le jeu à l'écran. A la fin, le programme affiche le nombre de pas nécessaires.

Exemple de sortie

Pour p = 3, votre programme doit afficher la séquence suivante:

```

BBB_NNN
BB_BNNN
BBNB_NN
BBNB_N
BBN_NBN
B_NBNBN
_BNBNNB
NB_BNBN
NBNB_BN
NBNNB_
NBNNB_B
NBN_NBB
N_NBNBB
NN_BNBB
NNNB_BB
NNN_BBB
15
    
```

Restrictions

$$1 \leq p \leq 30$$



Remettre le programme sous le nom SHUTTLE.xxx, avec xxx=PAS ou C(PP). Remettre également le fichier binaire exécutable SHUTTLE.EXE correspondant au programme.

Dans cet exercice, il s'agit de réaliser un programme pour décoder des messages SMS données en code hexadécimal.

A titre d'information, la procédure d'encodage est la suivante:

- a) coder le message donné en code ASCII décimal;
- b) transformer en code ASCII binaire sur 7 bits;
- c) miroiter les 7 bits;
- d) concaténer tous les bits;
- e) ajouter des 0 pour obtenir une chaîne dont la longueur est un multiple de 8;
- f) grouper par 8 bits afin d'obtenir des octets;
- g) miroiter les 8 bits des octets;
- h) transformer les octets en code hexadécimal.



Exemple

SMS: Bonjour

- a) 66 111 110 106 111 117 114
- b) 1000010 1101111 1101110 1101010 1101111 1110101 1110010
- c) 0100001 1111011 0111011 0101011 1111011 101011 0100111
- d) 010000111110110111011010101111110111010110100111
- e) 0100001111101101110110101011111101110101101001110000000
- f) 01000011 11101101 11011010 10111111 10111010 11010011 10000000
- g) 11000010 10110111 01011011 11111101 01011101 11001011 00000001
- h) C2 B7 5B FD AE CB 01

Exemple d'entrée

Le fichier d'entrée SMSIN.TXT contient sur une seule ligne les octets en code hexadécimal. Pour notre exemple, on aurait donc:

```
C2 B7 5B FD AE CB 01
```

Exemple de sortie

Le fichier de sortie SMSOUT.TXT contient le message décodé sur une seule ligne. Pour notre exemple, on aurait donc:

```
Bonjour
```

Restriction

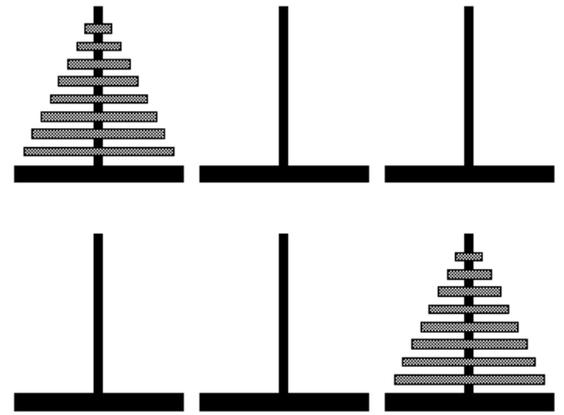
Le fichier d'entrée ne contiendra pas plus de 85 octets en code hexadécimal.



Remettre le programme sous le nom SMS.xxx, avec xxx=PAS ou C(PP). Remettre également le fichier binaire exécutable SMS.EXE correspondant au programme.

Le problème des tours de Hanoi est un jeu de réflexion imaginé par le mathématicien français Édouard Lucas. Il consiste à déplacer des disques de diamètres différents d'une tour de «départ» (celui à gauche dans notre illustration) à une tour d'«arrivée» (celui à droite dans notre illustration) en passant par une tour «intermédiaire» et ceci en un minimum de déplacements, tout en respectant les règles suivantes:

- on ne peut déplacer plus d'un disque à la fois;
- on ne peut placer un disque que sur un autre disque plus grand que lui ou sur un emplacement vide;
- cette dernière règle est également respectée dans la configuration de départ.



Ecrire un programme qui accepte en entrée la hauteur de la tour en disques et qui calcule les déplacements nécessaires. En plus, on demande que le programme compte le nombre de déplacements. Les 3 tours sont numérotées de 1 à 3. Les affichages se font à l'écran.

Exemple d'entrée et de sortie

Pour une hauteur de 4 disques, votre affichage devient:

```
4
1 -> 2
1 -> 3
2 -> 3
1 -> 2
3 -> 1
3 -> 2
1 -> 2
1 -> 3
2 -> 3
2 -> 1
3 -> 1
2 -> 3
1 -> 2
1 -> 3
2 -> 3
15
```



Remettre le programme sous le nom HANOI.xxx, avec xxx=PAS ou C(PP). Remettre également le fichier binaire exécutable HANOI.EXE correspondant au programme.