



Problème I - Détermination de la constante royale

La **constante royale** est l'unique nombre entier naturel de 9 chiffres satisfaisant aux conditions suivantes:

1. Tous les chiffres de 1 à 9 figurent exactement une fois dans le nombre.
2. Si le nombre est représenté par **abcdefghi**, alors

le nombre **a** est divisible par 1
le nombre **ab** est divisible par 2
le nombre **abc** est divisible par 3
le nombre **abcd** est divisible par 4
le nombre **abcde** est divisible par 5

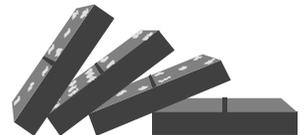
le nombre **abcdef** est divisible par 6
le nombre **abcdefg** est divisible par 7
le nombre **abcdefgh** est divisible par 8
le nombre **abcdefghi** est divisible par 9

Produisez un fichier texte ROYAL.CTE qui contient la constante royale - soit qu'un programme détermine la constante et l'inscrit dans le fichier, soit que vous inscrivez la constante vous-même dans le fichier. Dans ce dernier cas il faut que vous produisiez le fichier texte ROYAL.DOC où vous expliquez comment vous avez trouvé la constante.

 Si vous produisez un programme, sauvegardez le dans votre répertoire privé sous le nom ROYAL.xxx, avec xxx=PAS ou C(PP). Remettez dans ce cas également le fichier binaire exécutable ROYAL.EXE correspondant au programme.

Problème II - Domino

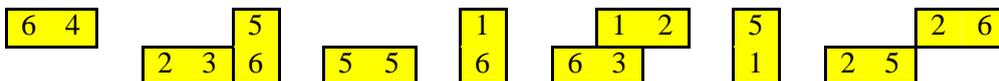
Le jeu de domino classique consiste dans un certain nombre de pierres rectangulaires plates divisées en deux parties dont chacune porte un nombre de points de 1 à 6. Le but du jeu est d'aligner le plus de pierres possibles de façon à ce que les parties avec le même nombre de points soient juxtaposés.



Pour un certain nombre de pierres données, écrivez un programme qui détermine un alignement de longueur maximale.

Exemple

Les 10 pierres suivantes



peuvent être alignées comme suit:



et il reste la pierre:



(Il existe d'autres possibilités d'alignement, mais il restera toujours une pierre.)

Entrée et sortie du programme

⇒ Le fichier texte DOMINO.IN dont la première ligne contient le nombre de pierres **M** (entier naturel non nul) et les **M** lignes suivantes (une ligne pour chaque pierre) contiennent deux chiffres **ab** correspondant au nombre de points des deux parties d'une pierre.

Un fichier test correspondant à l'exemple ci-dessus se trouve dans le sous- répertoire *CIL2003* sur le disque virtuel à lecture seule *Common On Server (S:)*. Faites-en une copie vers votre répertoire privé.

⇐ Le fichier texte DOMINO.OUT dont la première ligne contient le nombre **A** de pierres alignées (**A** est un entier naturel non nul) et les **A** lignes suivantes contiennent chaque fois deux chiffres **ab** correspondant au nombre de points des deux parties de chacune des pierres alignées. L'ordre des pierres dans le fichier doit être l'ordre de l'alignement de longueur maximale trouvé: le deuxième chiffre d'une des **A** lignes doit être égal au premier chiffre de la ligne suivante. Il suffit de trouver un seul alignement.

Exemple entrée et sortie

DOMINO.IN

10
64
23
56
55
16
63
12
51
25
26

DOMINO.OUT

9
16
63
32
25
55
51
12
26
64

Restrictions

$$1 \leq M \leq 32768$$

 Sauvegardez le programme dans votre répertoire privé sous le nom DOMINO.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable DOMINO.EXE correspondant au programme.

Problème III - Ecran tactile

Sur un écran tactile (anglais: touchscreen) se trouve une grille de jeu avec 3×3 carrés numérotés de 1 à 9 dont chacun peut prendre une des deux couleurs rouge ou vert. En touchant un des carrés, les couleurs des carrés changent suivant les règles suivantes.

1	2	3
4	5	6
7	8	9

En touchant

- ♦ un carré d'angle (1, 3, 7 ou 9), alors ce carré et les 2 carrés adjacents changent de couleur;
- ♦ un carré de côté (2, 4, 6 ou 8), alors ce carré et les 2 carrés d'angle adjacents changent de couleur;
- ♦ le carré du milieu (5), alors ce carré et les 4 carrés de côté adjacents changent de couleur.

Si on touche p.ex. le carré 2, alors les carrés 1, 2 et 3 changent de couleur. Si on touche le carré 9, alors les carrés 6, 8 et 9 changent de couleur. Si on touche le carré 5, alors les carrés 2, 4, 5, 6 et 8 changent de couleur.

Le jeu est terminé si le carré 5 est de couleur rouge et tous les autres carrés sont de couleur verte.

Ecrivez un programme qui détermine une suite de carrés à toucher afin d'arriver à la configuration finale, à partir d'une configuration initiale quelconque donnée. **Le nombre des carrés à toucher doit être minimal.**

Entrée et sortie du programme

- ⇒ Le programme lit du clavier les couleurs initiales des 9 carrés dans l'ordre de leurs numéros (R pour rouge et V pour vert).
- ⇐ Le programme affiche sur l'écran l'ordre des numéros des carrés à toucher.

Exemple entrée et sortie

Entrée standard: RVRVVVVVR

Sortie standard: 3854



Sauvegardez le programme dans votre répertoire privé sous le nom TACT.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable TACT.EXE correspondant au programme.

