Algorithmique - TD5

IUT 1ère Année

26 novembre 2012

1 Les chaînes

Exercice 1. Ecrire un algorithme permettant de résoudre le problème suivant :
- Données : une chaîne représentant un mot en minuscules (nom simple, verbe, adjectif) de la langue française.
- Résultat : le nombre de voyelles et le nombre de consonnes dans la chaîne.

Par exemple, le mot *élève* contient trois voyelles et deux consonnes.

Exercice 2. Ecrire un algorithme permettant de résoudre le problème suivant :
- Données : deux chaînes x et y représentant chacune un mot en majuscules (on n'utilise pas les accents) de la langue française.
- Résultat : *vrai* si x et y sont des anagrammes et *faux* sinon.

Rappelons que x est un *anagramme* de y si x est une permutation des lettres de y Par exemple, le mot CHIEN est un anagramme du mot NICHE. De la même manière, le mot LIERRE est un anagramme du mot RELIER. Mais LIRE n'est pas un anagramme de LIERRE puisqu'il n'utilise qu'un seul R et qu'un seul E.

Exercice 3. Ecrire un algorithme permettant de résoudre le problème suivant :
- Données : une chaîne de caractères x représentant un mot ou une expression de la langue française dont tous les caractères sont en majuscules.
- Résultat : *vrai* si x est un palindrome et faux sinon.

Rappelons que x est un *palindrome* s'il se lit de la même manière de gauche à droite et de droite à gauche. Par exemple, le mot RADAR et l'expression MON NOM sont des palindromes. Mais l'expression LE SEL n'est pas (strictement) un palindrome à cause du caractère 'espace'.

Exercice 4. Ecrire un algorithme permettant de résoudre le problème suivant :
- Données : une phrase de la langue française ne contenant pas de symbole de ponctuation autre que le point final.
- Résultat : le nombre de mots dans la phrase.

Par exemple, la phrase "*Le chat est couché dans le canapé.*" contient 7 mots.

Exercice 5. Ecrire un algorithme permettant de résoudre le problème suivant :
- Données : une phrase P de la langue française et un mot m, tous deux écrits en majuscules.
- Résultat : le nombre d'occurrences de m dans P.

Par exemple, la phrase "LE CHAT S'APPELLE PACHA." contient 2 fois le mot LE et 2 fois le mot CHA.

2 Les tableaux multi-dimensionnels

Exercice 6. Ecrire un algorithme permettant de résoudre le problème suivant :
- Données : une matrice **M** de dimension $n \times n$
- Résultat : *vrai* si **M** est la matrice identité et faux sinon.

Dans la matrice identité, tous les éléments de sa diagonale sont égaux à 1 et tous les autres éléments sont égaux à 0.

Exercice 7. Ecrire un algorithme permettant de résoudre le problème suivant :
- Données : deux matrices **A** et **B** de dimension $m \times n$
- Résultat : la somme $\mathbf{A}+\mathbf{B}$ des deux matrices.

Rappelons que si **C** est la somme de **A** et de **B**, alors pour toute rangée $1 \leq i \leq m$ et toute colonne $1 \leq j \leq n$ de **C**, nous avons $c_{ij} = a_{ij} + b_{ij}$.

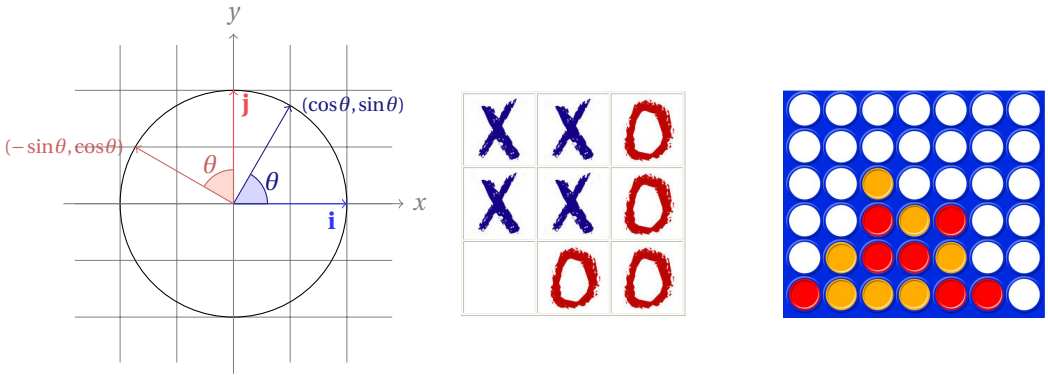

FIGURE 1 – Partie gauche : rotation selon un angle θ. Partie centrale : tic-tac-toe. Partie droite : puissance 4

Exercice 8. Ecrire un algorithme permettant de résoudre le problème suivant :
- Données : deux matrices **A** et **B** de dimension $n \times n$
- Résultat : le produit **AB** des deux matrices.

Rappelons que si **C** est le produit de **A** et de **B**, alors pour toute rangée $1 \leq i \leq n$ et toute colonne $1 \leq j \leq n$ de **C**, nous avons $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$.

Exercice 9*. Nous cherchons à construire un algorithme permettant d'effectuer des rotations d'objets en 2D. Etant donné un réel positif θ, la rotation en 2D selon l'angle θ est donnée par la matrice suivante :

$$\mathbf{R} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

1. Ecrire une procédure qui prend en entrée un point **P** dans $\mathbb{R}^2$ et un angle θ, et qui effectue une rotation de **P** par θ.
2. Ecrire un algorithme qui prend en entrée un tableau de n points (l'objet géométrique que l'on cherche à faire tourner) et un angle θ et qui effectue une rotation par θ de tous ses points.

Exercice 10*. Le *tic-tac-toe* (morpion) est un jeu de plateau à deux joueurs, le joueur × et le joueur ∘. Le plateau est un tableau 3×3 initialement vide. A chaque tour de jeu, le joueur × place une croix dans une case vide, puis le joueur ∘ place un cercle dans une autre case vide. Le jeu se termine lorsqu'un des deux cas se produit :
- L'un des joueurs a rempli une rangée, une colonne ou une diagonale. Dans ce cas le joueur gagne 1 points et son adversaire -1 point.
- Tout le plateau est rempli, sans qu'aucune rangée, colonne ou diagonale n'ait été remplie par un des joueurs. Dans ce cas, les deux joueurs ont 0 point.

Vous devez construire un algorithme qui prend en entrée un état du jeu et teste si le jeu est terminé. Si c'est bien le cas, l'algorithme calcule le nombre de points gagnés par chaque joueur.

Exercice 11*. *Puissance 4* est un jeu à deux joueurs constitué d'un plateau vertical de 6 rangées et de 7 colonnes. Le plateau est initialement vide. A chaque tour de jeu, le joueur *rouge* pose un jeton rouge dans une colonne j. Par gravité, le jeton vient au-dessus de la dernière case remplie de la colonne j. Le joueur *jaune* réalise la même opération en posant un jeton jaune dans une colonne. Vous devez construire une procédure qui prend en entrée un état du jeu, un joueur (jaune ou rouge) et une colonne (j), et produit en sortie l'état suivant du jeu. On supposera que la colonne j choisie par le joueur contient au moins une case vide.

Exercice 12*. Le jeu *Puissance 4* se termine lorsqu'un des deux cas se produit.
- L'un des joueurs a construit une rangée, colonne ou diagonale avec 4 pions consécutifs de sa couleur. Dans ce cas le joueur gagne 1 points et son adversaire -1 point.
- Tout le plateau est rempli, sans qu'aucune rangée, colonne ou diagonale de 4 pions consécutifs n'ait été construite par un des joueurs. Dans ce cas, les deux joueurs ont 0 point.

Comme pour l'exercice 10, vous devez construire un algorithme qui prend en entrée un état du jeu et teste si le jeu est terminé. Si c'est bien le cas, l'algorithme calcule le nombre de points gagnés par chaque joueur.