

Concours Mathématiques et Physique, Physique et Chimie, Technologie et
Biologie et Géologie
Epreuve d'Informatique

Date : Mardi 14 Juin 2005 Heure : 15 H Durée : 2 H Nombre de pages : 4

Barème : Exercice 1 : 4 points Exercice 2 : 4 points Problème : 12 points

L'usage des calculatrices est interdit.
Les exercices 1, 2 et le problème sont indépendants.

Exercice 1 :

Soit S le système linéaire suivant :

$$\begin{cases} 2mx + 3y - 2z = 3 \\ x + y + z = m \\ 5x + 4my + 2z = 0 \end{cases}$$

où m est un paramètre réel.

Ecrire les commandes Maple permettant de :

- 1) Définir S .
- 2) Résoudre le système S de deux manières :
 - 2-1 en utilisant la commande *solve*
 - 2-2 en utilisant la bibliothèque de fonctions « *linalg* ». On notera par A la matrice du système et b le vecteur second membre.
- 3) Calculer le déterminant k de A .
- 4) Trouver les racines $m1$ et $m2$ de k .
- 5) Déterminer la matrice IA l'inverse de A , sachant que IA n'est définie que pour les valeurs de $m \neq m1$ et $m \neq m2$.
- 6) Soit la matrice B définie par :

$$B = \begin{pmatrix} x+y & x+2y & x+3y \\ 2x+y & 2x+2y & 2x+3y \\ 3x+y & 3x+2y & 3x+3y \end{pmatrix}$$

- 6-1 Définir la fonction $f(i, j)$ permettant de construire les éléments de B avec i indice de ligne et j indice de colonne.
- 6-2 Construire B en utilisant la fonction f définie en 6-1.
- 7) Calculer et afficher les produits matriciels AB et AB^T (B^T désigne la matrice transposée de B).

Exercice 2 :

Soit n un entier naturel, on définit la suite polynomiale d'ordre n , par la relation de récurrence suivante :

$$\begin{cases} P_0(t) = 1 \\ P_1(t) = t \\ P_n(t) = \frac{2n-1}{n} t P_{n-1}(t) - \frac{n-1}{n} P_{n-2}(t) \end{cases} \quad \text{Pour } n > 1$$

- 1) Ecrire une procédure Maple nommée *Legendre* qui reçoit comme paramètres t et n , et retourne comme résultat du polynôme $P_n(t)$ sous forme développée. On utilisera la commande prédéfinie *expand* de Maple.
- 2) Ecrire la commande Maple permettant de représenter sur le même graphique, l'allure des polynômes $P_4(t)$, $P_5(t)$ et $P_6(t)$ pour $-1 \leq t \leq 1$.

Problème :

Soient A et B deux tableaux à une dimension de N entiers strictement positifs. On se propose de définir, à partir des tableaux A et B triés par ordre décroissant, les polynômes P et Q fonction de x et y , développés selon les puissances décroissantes de x puis selon les puissances décroissantes de y en utilisant les définitions suivantes :

$$P = \sum_{i=1}^N A[i] x^{B[i]} + \sum_{i=1}^N B[i] y^{A[i]}$$

$$Q = \sum_{i=1}^N B[i] x^{A[i]} + \sum_{i=1}^N A[i] y^{B[i]}$$

Exemple :

Si A et B sont deux tableaux à 4 éléments :

A

1	2	5	4
---	---	---	---

B

3	1	4	6
---	---	---	---

Les tableaux A et B triés par ordre décroissant sont :

A

5	4	2	1
---	---	---	---

B

6	4	3	1
---	---	---	---

Les polynômes P et Q définis à partir de A et B triés sont :

$$P = 5x^6 + 4x^4 + 2x^3 + x + 6y^5 + 4y^4 + 3y^2 + y$$

$$Q = 6x^5 + 4x^4 + 3x^2 + x + 5y^6 + 4y^4 + 2y^3 + y$$

Travail Demandé :

Partie A : Algorithmique

On utilisera la constante $NMAX$ et le type TAB définis par :

CONSTANTE $NMAX=20$

TYPE $TAB = \text{TABLEAU } [1..NMAX] \text{ d'ENTIER}$

- 1) Ecrire une fonction algorithmique, sans paramètres, appelée **TAILLE**, qui permet de saisir un entier compris entre 1 et **NMAX** puis retourner ce même entier.
- 2) Ecrire une procédure algorithmique, appelée **REEMPLIR_TABLEAU**, qui permet de remplir un tableau **T** avec des entiers strictement positifs et qui prend comme paramètres :

N : entier paramètre donné passé par valeur
T : TAB paramètre résultat passé par variable.

- 3) Dans la suite du problème, on précise l'hypothèse suivante :

On suppose que les éléments du tableau **T** saisis par la procédure **REEMPLIR_TABLEAU** sont tous distincts.

On se propose de trier, par ordre décroissant, un tableau **T** en deux étapes selon le principe suivant :

▪ 1^{ère} étape : *Création d'un tableau de compteurs.*

Chaque élément du tableau **T** à trier est comparé à tous les autres afin de déterminer le nombre d'éléments qui lui sont strictement supérieurs. Les résultats des **N** comptages sont rangés dans un tableau de compteurs, appelé **C**, possédant lui aussi **N** éléments. Les valeurs du tableau **C** sont comprises entre 0 et (**N-1**) : le plus grand élément de **T** aura pour valeur du compteur 0 car il n'a aucun élément plus grand que lui et le plus petit élément aura pour valeur du compteur (**N-1**).

Exemple :

Tableau à trier : **T**

1	2	5	4
---	---	---	---

Tableau de compteurs : **C**

3	2	0	1
---	---	---	---

▪ 2^{ème} étape : *Utilisation du tableau de compteurs pour le tri.*

Le premier élément du tableau **T** à trier, est permuté avec l'élément dont le compteur est égal à 0. Les éléments correspondants dans le tableau de compteurs sont aussi permutés. Le deuxième élément du tableau **T** est permuté avec l'élément de **C** égal à 1 et ainsi de suite. Ce processus est répété jusqu'à la valeur (**N-1**) de **C**.

- 3-1 Ecrire une procédure algorithmique, appelée **COMPTAGE**, qui a pour rôle de remplir le tableau de compteurs **C** à partir du tableau **T** et qui prend comme paramètres :

N : entier paramètre donné passé par valeur
T : TAB paramètre donné passé par valeur
C : TAB paramètre résultat passé par variable.

- 3-2 Ecrire une procédure algorithmique, appelée **TRI_TABLEAU**, pour trier par ordre décroissant un tableau **T**, selon le principe de tri détaillé précédemment, en utilisant la procédure **COMPTAGE** écrite en 3-1.

La procédure **TRI_TABLEAU** prend comme paramètres :

N : entier paramètre donné passé par valeur
T : TAB paramètre donné/résultat passé par variable.

- 4) Ecrire une procédure algorithmique, appelée **AFFICH_POLY**, qui affiche un polynôme de la forme :

$$\sum_{i=1}^N A[i]x^{B[i]} + \sum_{i=1}^N B[i]y^{A[i]}$$

sous le format donné par l'exemple suivant :

Exemple : Si P est un polynôme défini par :

$$P = 5x^4 + 3x^3 + 2x^2 + x + 4y^5 + 3y^3 + 2y^2 + y$$

Le format d'affichage du polynôme P sera :

$$P = 5x^4 + 3x^3 + 2x^2 + x + 4y^5 + 3y^3 + 2y^2 + y$$

Cette procédure prend comme paramètres :

N : entier	paramètre donné passé par valeur
A : TAB	paramètre donné passé par valeur
B : TAB	paramètre donné passé par valeur.

5) Ecrire un algorithme principal **AFF_POLY_ORD** qui :

- permet de saisir le nombre N d'éléments, d'un tableau à une dimension, en appelant la fonction **TAILLE** écrite en 1) de la partie A.
- remplit un tableau A ensuite un tableau B avec des entiers strictement positifs en appelant la procédure **REEMPLIR_TABLEAU** écrite en 2) de la partie A.
- trie le tableau A ensuite le tableau B en utilisant la procédure **TRI_TABLEAU** écrite en 3-2) de la partie A.
- affiche les polynômes P et Q (décrits au début du problème) selon les puissances décroissantes en x puis en y en appelant la procédure **AFFICH_POLY** écrite en 4) de la partie A.

Partie B : Maple

- 1) Traduire en Maple la procédure algorithmique **COMPTAGE** écrite en 3-1 de la partie A.
- 2) Traduire en Maple la procédure algorithmique **TRI_TABLEAU** écrite en 3-2 de la partie A.
- 3) Donner la commande Maple permettant d'ordonner un polynôme P selon les puissances décroissantes de x .
- 4) Ecrire la commande Maple qui convertit le polynôme P en une fonction f à deux variables x et y .
- 5) Donner la commande Maple pour définir le domaine de définition de la fonction f .
- 6) Donner la commande Maple pour définir l'expression de la dérivée première de f par rapport à x .
- 7) Donner la commande Maple pour définir la fonction dérivée seconde de f par rapport à y .