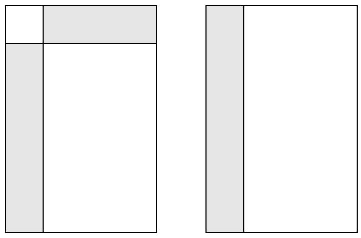
Devoir Surveillé XX d'Informatique (Concours blanc 2015)

- **Durée : 4 heures**
- **L'usage de tout dispositif électronique (calculatrice, téléphone portable, . . .) est interdit.**
- La clarté et la lisibilité de la copie sont des qualités essentielles. Toute copie sale, peu claire ou à la rédaction approximative sera **pénalisée**. Numérotez vos feuilles doubles et indiquez votre nom sur chacune d'entre elles.



Première feuille Autres feuilles

  La première page de la copie **doit laisser un cadre de 5cm**. Votre copie doit comporter une marge d'**au moins 3cm**. Toute copie ne respectant pas ces spécifications perdra des points.
- Il est **TRÈS fortement** conseillé de lire le sujet en entier avant de s'attaquer aux réponses.
- Un problème, ou un exercice à plusieurs questions, n'est pas un grand chelem, et il est approprié en l'absence de réponse à une question de supposer les résultats et de passer à la suivante. **Ne vous arrêtez pas à la première question dont vous n'avez pas la réponse !** Aucune copie ne sera pénalisée pour avoir tenté un début de réponse.
- S'il vous semble qu'une erreur s'est glissée dans un énoncé, justifiez-vous et poursuivez l'exercice *mutatis mutandis*, c'est-à-dire *en changeant ce qui doit être changé*.

Les deux parties sont **indépendantes**. Chacune des deux parties vise à concevoir des algorithmes efficaces et à en étudier la complexité. Chacune des parties tire parti du paradigme “diviser pour régner".

1 Multiplication de Matrices, algorithme de Strassen

1.1 Opérations sur les matrices

Soient $n, m \in \mathbb{N}$. Une matrice $n \times m$ est un *tableau* de $n * m$ éléments (ici des entiers relatifs), indicés par deux indices $1 \leqslant i \leqslant n$ et $1 \leqslant j \leqslant m$. On note $A = [a_{i,j}]_{i \leqslant n, j \leqslant m}$ où $a_{i,j}$ est l'élément de la *ligne* i et de la *colonne* j.

Par exemple, la matrice $A = [i + \frac{1}{j^2}]_{i \leqslant 3, j \leqslant 4} = \begin{bmatrix} 2 & \frac{5}{4} & \frac{10}{9} & \frac{17}{16} \\ 3 & \frac{9}{4} & \frac{19}{9} & \frac{33}{16} \\ 4 & \frac{13}{4} & \frac{28}{9} & \frac{49}{16} \end{bmatrix}$. Soit $B = \begin{bmatrix} -4 & \frac{1}{4} & 0 & \frac{17}{16} \\ 3 & 2 & \frac{1}{9} & -\frac{15}{16} \\ 6 & \frac{13}{4} & \frac{1}{9} & \frac{49}{16} \end{bmatrix}$.

La somme de deux matrices $A = [a_{i,j}]_{i \leqslant n, j \leqslant m}$ et $B = [b_{i,j}]_{i \leqslant n, j \leqslant m}$ est la matrice $C = A + B = [a_{i,j} + b_{i,j}]_{i \leqslant n, j \leqslant m}$. La différence de deux matrices est définie de façon similaire. Par exemple, $A - B = \begin{bmatrix} 6 & 1 & \frac{10}{9} & 0 \\ 0 & \frac{1}{4} & 2 & 3 \\ -2 & 0 & 3 & 0 \end{bmatrix}$.

Le produit de deux matrices $A = [a_{i,j}]_{i \leqslant n, j \leqslant m}$ et $B = [b_{i,j}]_{i \leqslant m, j \leqslant p}$ résulte en la matrice $A * B = C = [c_{i,j}]_{i \leqslant n, j \leqslant p}$ définie par $c_{i,j} = \sum_{1 \leqslant k \leqslant m} a_{i,k} b_{k,j}$. Notons que la matrice $A * B$ n'est définie que si les dimensions sont compatibles, c'est-à-dire que le nombre de colonnes de A doit être égal au nombre de lignes de B [1].

1. Bien que cette définition puisse paraître étrange au premier abord, elle devient naturelle lorsque l'on considère une matrice comme une application dans un espace vectoriel, par exemple une rotation dans le plan, et que la multiplication correspond à la composition de deux applications.

$$
\begin{array}{cc}
 & \begin{bmatrix} b_{1,j} \\ b_{2,j} \\ \dots \\ b_{i,j} \\ \dots \\ b_{m,j} \end{bmatrix} = B \\
A = \begin{bmatrix} \\ a_{i,1} & a_{i,2} & \dots & a_{i,j} & \dots & a_{i,m} \\ \\ \end{bmatrix} & \begin{bmatrix} & & & \dots & \\ & & & \dots & \\ .. & .. & .. & c_{i,j} = \sum_{1 \leqslant k \leqslant m} a_{i,k} b_{k,j} & \\ \\ \end{bmatrix} = A * B
\end{array}
$$

Question 1	Soit $C = \begin{bmatrix} 1 & 3 & 1 \\ 0 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix}$ et $D = \begin{bmatrix} 6 & 1 & 0 \\ 0 & 2 & 3 \\ -2 & 0 & 0 \end{bmatrix}$. Calculer $D * C$ et $C * D$. Conclusion ?

Question 2	Donner un algorithme qui calcule le produit d'une matrice $n \times m$ avec une matrice $m \times p$ en $O(npm)$ opérations élémentaires (multiplications, additions et soustractions d'entiers). Prouver la correction et la complexité de l'algorithme. En CAML, une matrice peut être représentée par un vecteur de vecteurs (lignes). Donner le code CAML correspondant.

Question 3	Soient A et B deux matrices $n \times m$ et C une matrice $m \times n$. Prouver que $C * (A + B) = C * A + C * B = C * B + C * A$. Conclusions ? Sans détailler la preuve, comparer $(A + B) * C$ et $A * C + B * C$.

1.2 Matrices par blocs

Pour simplifier les notations, il est possible de définir une matrice *par blocs*. Soient quatre matrices $U = [u_{i,j}]_{i \leqslant \ell, j \leqslant k}$, $V = [v_{i,j}]_{i \leqslant \ell, j \leqslant n-k}$, $W = [w_{i,j}]_{i \leqslant m-\ell, j \leqslant k}$, $R = [r_{i,j}]_{i \leqslant m-\ell, j \leqslant n-k}$.

La matrice $X = [x_{i,j}]_{i \leqslant m, j \leqslant n}$, notée $X = \begin{bmatrix} U & V \\ W & R \end{bmatrix}$, est définie par

- $x_{i,j} = u_{i,j}$ si $i \leqslant \ell$ et $j \leqslant k$,
- $x_{i,j} = v_{i,j}$ si $i \leqslant \ell$ et $k < j \leqslant n$,
- $x_{i,j} = w_{i,j}$ si $\ell < i \leqslant m$ et $j \leqslant k$, et
- $x_{i,j} = r_{i,j}$ si $\ell < i \leqslant m$ et $k < j \leqslant n$.

Par exemple, $\begin{bmatrix} A & B \\ A - B & 0 \end{bmatrix} = \begin{bmatrix} 2 & \frac{5}{4} & \frac{10}{9} & \frac{17}{16} & -4 & \frac{1}{4} & 0 & \frac{17}{16} \\ 3 & \frac{9}{4} & \frac{19}{9} & \frac{33}{16} & 3 & 2 & \frac{1}{9} & -\frac{15}{16} \\ 4 & \frac{13}{4} & \frac{28}{9} & \frac{49}{16} & 6 & \frac{13}{4} & \frac{1}{9} & \frac{49}{16} \\ 6 & 1 & \frac{10}{9} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 2 & 3 & 0 & 0 & 0 & 0 \\ -2 & 0 & 3 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

Notons que le bloc "0" correspond à la matrice dont tous les éléments sont nuls et dont les dimensions sont implicitement définies par les dimensions de $A - B$ et B.

Question 4	Soient 8 matrices $A = [a_{i,j}]_{i \leqslant n, j \leqslant m}$, $B = [b_{i,j}]_{i \leqslant n, j \leqslant q}$, $C = [c_{i,j}]_{i \leqslant \ell, j \leqslant m}$, $D = [d_{i,j}]_{i \leqslant \ell, j \leqslant q}$, $U = [u_{i,j}]_{i \leqslant m, j \leqslant p}$, $V = [v_{i,j}]_{i \leqslant m, j \leqslant t}$, $W = [w_{i,j}]_{i \leqslant q, j \leqslant p}$ et $R = [r_{i,j}]_{i \leqslant q, j \leqslant t}$. Soit $X = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ et $Y = \begin{bmatrix} U & V \\ W & R \end{bmatrix}$. Prouver que $X * Y = \begin{bmatrix} A * U + B * W & A * V + B * R \\ C * U + D * W & C * V + D * R \end{bmatrix}$.

1.3 Algorithme de Strassen

Soit $p \in \mathbb{N}^*$. Soient A, B, C, D, E, F, G et H huit matrices $2^{p-1} \times 2^{p-1}$. Soient $M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ et $N = \begin{bmatrix} E & F \\ G & H \end{bmatrix}$.

Soient $P_1 = A * (F - H)$, $P_2 = (A + B) * H$, $P_3 = (C + D) * E$, $P_4 = D * (G - E)$, $P_5 = (A + D) * (E + H)$, $P_6 = (B - D) * (G + H)$ et $P_7 = (A - C) * (E + F)$.

Question 5	Quelles sont les dimensions de $M, N, M * N$ et P_i, pour tout $i \leqslant 7$? Soit $i \leqslant 7$. Quel est le nombre d'opérations élémentaires (opérations sur les entiers) pour calculer P_i ?

Question 6	Prouver que $M * N = \begin{bmatrix} -P_2 + P_4 + P_5 + P_6 & P_1 + P_2 \\ P_3 + P_4 & P_1 - P_3 + P_5 - P_7 \end{bmatrix}$ Les matrices P_i, $i \leqslant 7$, étant données, donner le nombre d'opérations élémentaires (additions et soustractions) pour calculer $M * N$.

Question 7	Soit $c(p)$ le nombre d'opérations élémentaires (multiplications, additions et soustractions d'entiers) pour calculer le produit de deux matrices $2^p \times 2^p$. Exprimer $c(p)$ en fonction de $c(p-1)$ et de p.

Question 8	Calculer $c(p)$ et comparer à l'algorithme naïf proposé plus haut.

Question 9	Donner, en fonction de n, un ordre de grandeur du calcul du produit de deux matrices $n \times n$ en utilisant l'algorithme de Strassen.

2 La chasse aux fantômes

Un groupe de n chasseurs de fantômes combat un groupe de n fantômes. Chaque chasseur est armé d'un canon à ions, capable d'éradiquer un fantôme d'un coup de rayon. Un rayon se propage en ligne droite et termine sa course en touchant le fantôme. Les chasseurs sont confrontés à deux problèmes : il est nécessaire d'éliminer tous les fantômes en même temps, sinon ils se multiplient pour revenir au nombre initial de n fantômes ; il est très dangereux que deux rayons se croisent, cela ne doit donc pas se produire. Le combat se déroule dans une grotte cylindrique creuse, dont la base est un disque. Les déplacements (chasseurs et fantômes) ne sont possibles que sur un rebord confondu avec la circonférence, mais les tirs se font entre deux points de la circonférence, selon des cordes du cercle.

2.1 Stratégie et stratégie admissible

Les $2n$ protagonistes sont représentés par des points distincts $P_1, P_2, \cdots, P_{2n}$ répartis dans l'ordre des indices croissants sur la circonférence d'un cercle parcouru dans le sens trigonométrique ; n sont des chasseurs, n des fantômes.

Dans cette section 2.1, on cherche les liens possibles entre les P_i sans s'occuper de qui est chasseur et qui est fantôme. Les points doivent être reliés deux à deux par une corde du cercle selon une stratégie gagnante. Pour cela, il faut donc respecter les deux conditions :

1. chaque point est une extrémité d'une et une seule corde,
2. les différentes cordes ne doivent pas se couper.

Dans toute la suite on fixe un entier $n > 0$.

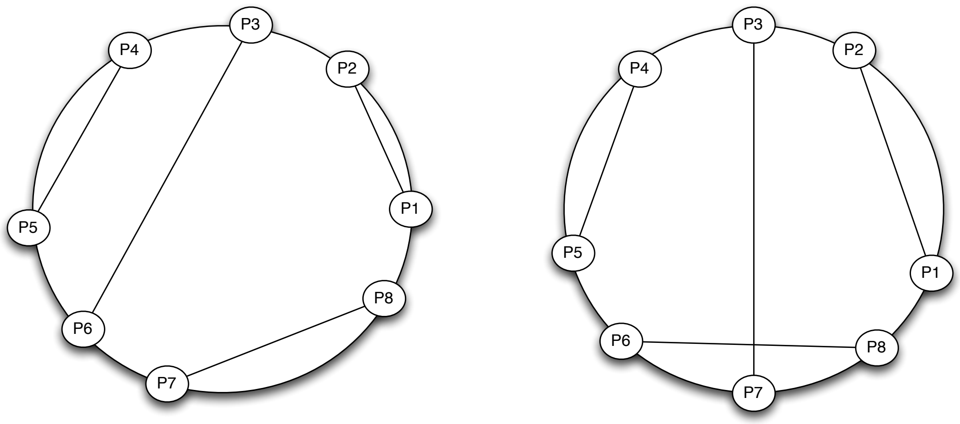
Définition	On appelle *stratégie* de taille n un vecteur $(c[1], c[2], \cdots, c[2n])$, tel que pour tout entier i vérifiant $1 \leqslant i \leqslant 2n$ on ait : – $1 \leqslant c[i] \leqslant 2n$, – $c[c[i]] = i$, et – $c[i] \neq i$.

Cette notation exprime bien le lien entre les points P_i et P_j lorsque $j = c[i]$.

Définition	On considère des stratégies où les points d'une paire sont reliés par des segments de droites situés dans un même plan. Une stratégie est dite *admissible* si les segments ainsi formés ne se coupent pas.

On admet que cette propriété ne dépend pas de la position des points sur la circonférence mais seulement de la stratégie.

Par exemple, les 2 stratégies $c_1 = (2, 1, 6, 5, 4, 3, 8, 7)$ et $c_2 = (2, 1, 7, 5, 4, 8, 3, 6)$ sont représentées sur la Figure 2.1. La stratégie c_1 (à gauche) est admissible alors que la stratégie c_2 (à droite) ne l'est pas.



Stratégie c1 Stratégie c2

Question 10	Dans le cas $n = 3$, donner le nombre de stratégies possibles (admissibles ou non).

Question 11	Préciser le nombre de stratégies de taille 3 qui sont admissibles. Donner les vecteurs correspondants et les représenter sur une figure.

Question 12	Prouver que le nombre de stratégies (admissibles ou non) de taille n est $\frac{(2n)!}{2^n n!}$.

2.2 Décider si une stratégie est admissible, algorithme naïf

Dans cette section 2.2, on ne s'intéresse toujours pas à qui est chasseur et qui est fantôme.

Soient i, j, k, ℓ quatre entiers distincts tels que $1 \leqslant i < j \leqslant 2n$ et $1 \leqslant k < \ell \leqslant 2n$.

Question 13	Donner une condition nécessaire et suffisante, portant uniquement sur ces entiers, pour que les segments $[P_i P_j]$ et $[P_k P_\ell]$ se croisent.

Question 14	Écrire en $Caml$ une fonction “$croise$" prenant pour paramètres quatre entier i, j, k, ℓ tels que $1 \leqslant i < j \leqslant 2n$ et $1 \leqslant k < \ell \leqslant 2n$, et renvoyant un booléen dont le résultat est $true$ si et seulement si les segments $[P_i P_j]$ et $[P_k P_\ell]$ se croisent.

Question 15	En utilisant la fonction $croise$, écrire en $Caml$ une fonction “$estAdmissible$" qui prend en paramètre le vecteur et renvoie un booléen dont le résultat est $true$ si et seulement si la stratégie est admissible.

Question 16	Donner, en la justifiant, la complexité de cette méthode, en terme de temps de calcul (on se contentera d'une évaluation du type $O(n^\alpha)$, en précisant la valeur de α).

2.3 Décider si une stratégie est admissible en temps linéaire

Encore une fois, dans cette section 2.3, on ne s'intéresse pas à qui est chasseur et qui est fantôme.

On souhaite tester si une stratégie c est admissible de façon plus efficace. Pour cela on utilise une fonction “$evalue$" qui prend deux entiers i et j comme paramètres (i et j étant compris entre 1 et $2n$) et dont le résultat est $true$ si et seulement si les deux propositions suivantes sont vraies :

- $(i \leqslant k \leqslant j) \Rightarrow (i \leqslant c[k] \leqslant j)$
- les segments ayant leurs extrémités dans l'arc (fermé) $[P_i P_j]$ ne se croisent pas.

Question 17	Donner la valeur de $evalue(i, j)$ dans les trois cas particuliers suivants : 1. $j < i$ 2. $j > i$ et $c[i] < i$ 3. $j > i$ et $c[i] > j$.

Question 18	Pour $i < c[i] < j$, montrer que $evalue(i, j)$ se déduit de $evalue(i+1, c[i]-1)$ et de $evalue(c[i]+1, j)$.

Question 19	Écrire en $Caml$ une fonction $testStrategie$ qui prend en paramètre un vecteur (une stratégie de taille n) et renvoie un booléen dont la valeur est $true$ si et seulement si la stratégie est admissible. Cette fonction devra être de complexité $O(n)$.

Question 20	Démontrer avec soin que la fonction $testStrategie$ est bien de complexité $O(n)$.

2.4 Construire une stratégie admissible

Pour finir, on s'intéresse à qui est chasseur et qui est fantôme.

La méthode précédente teste si une stratégie est admissible. On souhaite maintenant, le statut (chasseur ou fantôme) des points étant donné, déterminer directement une stratégie admissible, de façon également efficace.

Question 21	Montrer que si P_i est un chasseur ($1 \leqslant i \leqslant 2n$), alors il existe $j \neq i$, avec $1 \leqslant j \leqslant 2n$ tel que P_j est un fantôme et tel que le nombre de chasseurs soit égal au nombre de fantômes de chaque côté de l'axe $P_i P_j$.

Question 22	En déduire un algorithme simple pour construire une stratégie admissible.

Question 23	Déterminer la complexité de cette méthode dans le pire des cas.

Un vecteur p contient la nature des points P_i : $p[i] = 1$ pour un chasseur, $p[i] = -1$ pour un fantôme $(1 \leqslant i \leqslant 2n)$. On a donc $\sum_{i=1}^{2n} p[i] = 0$ puisqu'il y a autant de fantômes que de chasseurs.

Question 24	Écrire en Caml une fonction $cibles$, qui prend en paramètre le vecteur p et qui renvoie la liste des indices (i, j) des couples (P_i, P_j) (où P_i est un chasseur et P_j un fantôme), composant une stratégie admissible du problème.