ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
Section de Systèmes de Communication

Examen final 5 juillet 2004

Algorithmique I-II

5 Juillet 2004, 14h-17h

- Le seul document autorisé est une feuille *A4* recto-verso
- Les calculatrices, téléphones, ordinateurs etc... sont interdits.
- Utiliser les feuilles de couleurs comme brouillons.

Nom :

Prénom :

Exercice 1 / points	Exercice 2 / points	Exercice 3 / points	Exercice 4 / points	Exercice 5 / points
5	10	4		

Exercice 6 / points	Exercice 7 / points	Exercice 8 / points	Exercice 9 / points	Exercice 10 / points

Total / 100

Problème 1 [5 points]. Donner une spécification formelle du problème suivant : étant donné un ensemble fini de nombres naturels M, combien y a-t-il d'éléments de M qui sont multiple de 5 ?

a) Quel est l'ensemble des inputs ?

b) Quel est l'ensemble des outputs ?

c) Quel est la dépendance relationelle ?

Problème 2 [10 points]. Supposons que nous avons 4 matrices aux formats suivants.

Matrice	Format
M_1	6×6
M_2	6×5
M_3	5×9
M_4	9×4

En utilisant la méthode de la programmation dynamique, calculer le nombre minimum de multiplications nécessaires et la disposition des parenthèses de l'expression $M_1 \cdot M_2 \cdot M_3 \cdot M_4$ qui donne ce minimum.

Problème 3 [7 points].

a) Calculer le nombre de fois que l'on vérifie la propriété P dans l'algoritme qui suit. Noter que $\sum_{i=1}^{n} i^3 = \frac{n^4+2n^3+n^2}{4}$.

Input: Un entier n
Output: Verifications de la propriété P
  for $i = 1, 2, 3, ..., n$ **do**
    for $j_1 = 1, 2, 3, ..., 2i$ **do**
      for $j_2 = 1, 2, 3, ..., 3i$ **do**
        for $j_3 = 1, 2, 3, ..., 4i$ **do**
          Verifier la proprieté P
        end for
      end for
    end for
  end for

b) Calculer le nombre d'èvaluations de la fonction f dans l'algorithme ci-dessous :

Input: Un entier n
Output: Impressions du résultat de $f(x, y, t)$
  for $y = 1, 2, 3, ..., n$ **do**
    for $x = y, 2y, 3y, ..., y^2$ **do**
      for $t = 1, 2, 3, ..., \frac{x}{y}$ **do**
        Imprimer le résultat de $f(x, y, t)$
      end for
    end for
  end for

Problème 4 [12 points]. Rappelons que la série de Fibonaci est définie par,

$$\begin{cases} f(n) = f(n-1) + f(n-2) \\ f(1) = 1, f(2) = 1 \end{cases}$$

Soit la série définie par

$$\begin{cases} g(n) = g(n-1) + g(n-2) \\ g(1) = 1, g(2) = 3 \end{cases}$$

a) Montrer que $g(n) = f(n-1) + f(n+1)$ et $f(n)g(n) = f(2n)$

b) Calculer la limite l du rapport $\frac{f(n)}{f(n-1)}$.

c) Supposons que $l = \frac{1+\sqrt{5}}{2}$. Montrer que

$$\begin{cases} \frac{f(n)}{f(n-1)} < l \Longrightarrow \frac{f(n+1)}{f(n)} > l \text{ et} \\ \frac{f(n)}{f(n-1)} > l \Longrightarrow \frac{f(n+1)}{f(n)} < l \end{cases}$$

d) Calculer $\frac{f(2)}{f(1)}$. Deduire que

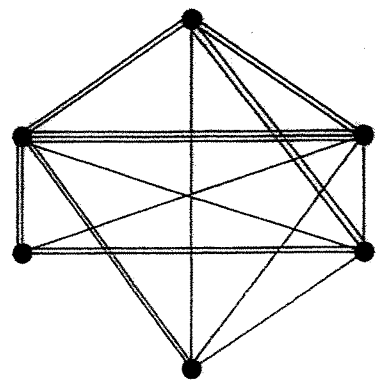
$$\frac{f(n)}{f(n-1)} \begin{cases} < l & \text{si } n \text{ est pair} \\ > l & \text{si } n \text{ est impair} \end{cases}$$

Problème 5 [12 points].

Soit $G = (V, E)$ un graphe. Un *cycle de longueur* $k+1$ dans G est une suite de sommets $(v_0, v_1, \ldots, v_k)$ telle que $v_k = v_0$, et pour tout $0 \leq i < k$ on a $(v_i, v_{i+1}) \in E$. Le graphe est dit *2-colorable* si on peut colorer les sommets avec les couleurs rouge ou bleu tel que deux sommets adjacents ne soient pas de couleur identique.

1. Montrer que si G est 2-colorable, alors G ne contient pas de cycle de longeur impaire.
2. Donner un algorithme glouton qui colore G avec les 2 couleurs. Quand est-ce que votre algorithme ne marche pas ? Combien d'operations nécessite votre algorithme en fonction du nombre d'arêtes et du nombre de sommets ?
3. Montrer que si G ne contient pas de cycle de longueur impaire, alors G est 2-colorable.

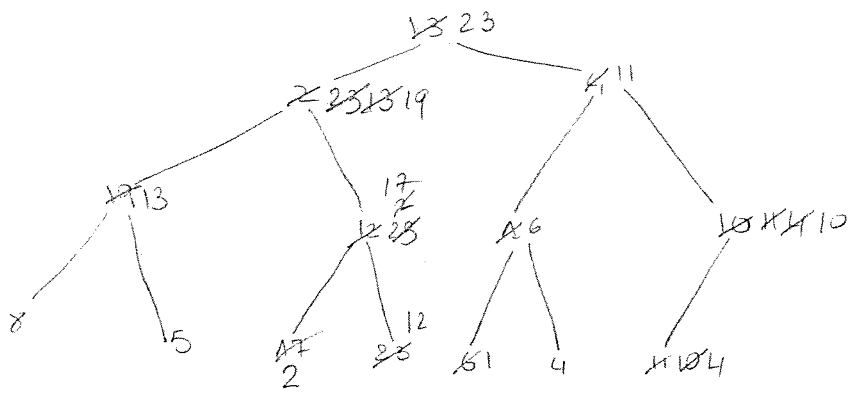
Problème 6 [10 points]. Dans le graphe ci-dessous, determiner un sous-ensemble S de sommets tel que le nombre d'arcs qui connectent les sommets de S aux sommets qui n'appartiennent pas à S est minimal. Noter que dans ce graphe, il y a plusieurs arcs entre certains sommets.



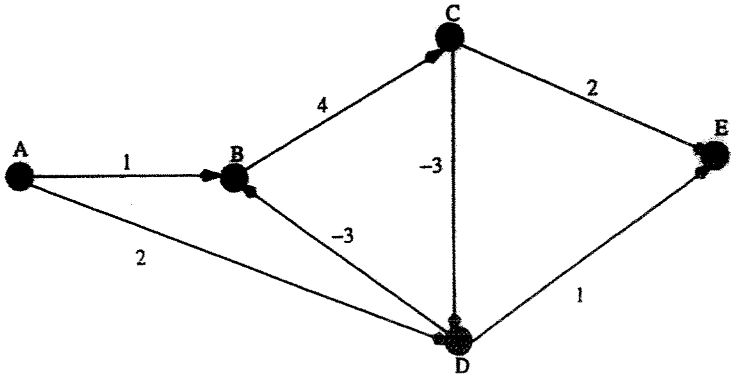
Problème* *7 [10 points]. Créer un heap à partir des nombres donnés dans l'ordre suivant en utilisant la méthode bottom-up heap :

$$13, 2, 4, 19, 12, 1, 10, 8, 5, 17, 23, 6, 4, 11$$

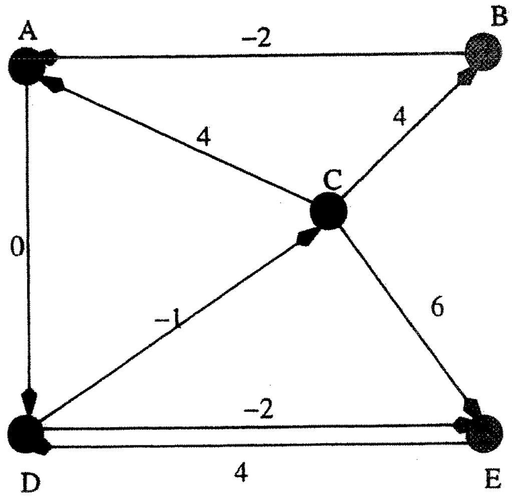
(Donner le heap final sans description supplémentaire)



Problème 8 [12 points]. Dans les deux graphes suivants, appliquer, quand cela est possible, soit l'algorithme Dijkstra soit l'algorithme Moore-Bellman-Ford. Pour chaque graphe, justifier le choix de l'algorithme.

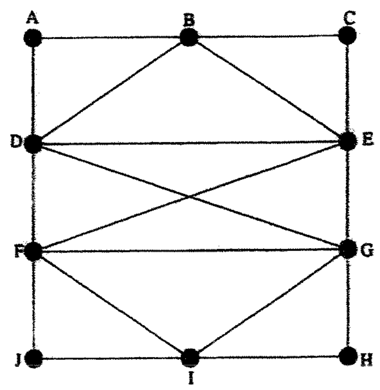


1.



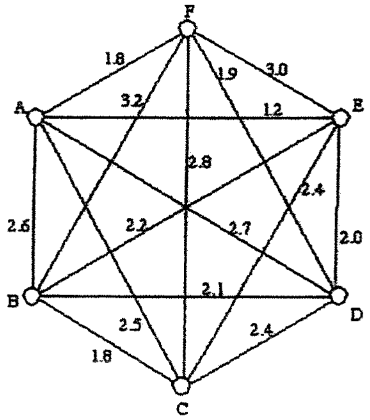
2.

Problème 9 [10 points]. Soit G le graphe suivant :



1. Donner un ordonnancement possible des sommets après avoir appliqué l'algorithme BFS sur G.
2. Donner un ordonnancement possible des sommets après avoir appliqué l'algorithme DFS sur G.
3. Donner un tri topologique des sommets.

Problème 10 [12 points]. Soit G le graphe suivant dont chaque arc est muni d'un poids :



1. Utiliser l'algoritme de Kruskal's pour trouver l'arbre minimum couvrant et indiquer les arcs choisis à chaque étape de l'algorithme. (Par exemple, le premier arc choisi pourrait être AE, puis AF, etc.) Dessiner l'arbre minimum couvrant trouvé.
2. Utiliser l'algorithme de Prim pour trouver l'arbre minimum couvrant et indiquer les arcs choisis à chaque étape de l'algorithme .