Mariage maximum

1. $M = \emptyset$
   Tant qu'il existe une arête e de $E - M$ ne partageant aucune extrémité avec une arête de M,
   ajouter e à M.
   Retourner M.
2. Non, et le graphe suivant donne un contre-exemple : $X = \{a, b\}$, $Y = \{1, 2\}$, les arêtes sont a1, b1 et b2. $\{b1\}$ est un mariage maximal non maximum.
3. Soit N l'ensemble des arêtes de M empruntées par le chemin augmentant, et O l'ensemble des autres arêtes (de $E - M$) empruntées par ce chemin. $O \cup (M - N)$ est un mariage de cardinalité $|M| + 1$.
4. On a $|C| \geq |M|$ puisque les arêtes de M doivent être couvertes par des sommets distincts.
5. Terminaison : à chaque exécution complète de la boucle «tant que» on marque un sommet de X non marqué, donc cette boucle ne peut être exécutée plus de $|X|$ fois.

   Par construction, pour tout sommet $x \in S$ il existe un chemin $\gamma(x)$ M-alterné d'un sommet de U à x (réduit à x si $x \in U$). Ceci se montre par récurrence : quand on ajoute w à S on peut prendre $\gamma(w) = \gamma(x)yw$. Il faut vérifier que y et w n'apparaissent pas dans $\gamma(x)$. C'est vrai pour y à cause de la condition $y \notin T$. C'est vrai pour w car avant d'ajouter w à S on a $w \notin S$. En effet, on ne peut pas avoir $w \in U$ puisque les sommets de U sont M-insaturés, et on ne peut pas non plus avoir $w \in S \setminus U$ puisque les sommets de $S \setminus U$ sont reliés par une arête de M à un sommet de T (et donc comme M est un mariage ils ne peuvent être reliés également à un sommet de $Y \setminus T$).

   Dans le cas où l'algorithme s'arrête en répondant «M n'est pas maximum», en prolongeant $\gamma(x)$ jusqu'à y on obtient un chemin M-augmentant donc effectivement M n'est pas maximum.

   Il reste donc à montrer que si l'algorithme s'arrête en retournant une couverture, $T \cup (X - S)$ est bien une couverture, c'est-à-dire qu'il n'existe pas d'arête entre S et $Y - T$. Tout sommet x de $S \setminus U$ est relié à un sommet de T par une arête de M (en fait par la dernière arête de $\gamma(x)$) donc par définition d'un mariage, x ne peut pas être relié également par une arête de M à un sommet de $Y - T$. En fait, aucun sommet de S n'est relié par une arête de M à un sommet de $Y - T$ puisque les sommets de U sont M-insaturés.

   Un sommet x de S ne peut pas non plus être relié à $Y - T$ par une arête de $E - M$ puisque dans la boucle Pour, tout sommet y qui vérifierait $y \notin T$ et $xy \in E \setminus M$ est ajouté à T. Donc effectivement x n'est relié à aucun sommet de $Y - T$.

6. Tout sommet de $X - S$ est une extrémité d'une arête de M puisque l'ensemble U des sommets M-insaturés de X est inclus dans S. De même en utilisant les notations de l'algorithme, tout sommet y de T est relié par une arête de M à un sommet w de S. Toutes les arêtes de M ainsi obtenues sont distinctes donc $|M| \geq |T| + |X - S|$.

   Mais on a vu que $|M| \leq |T| + |X - S|$, donc en fait $|M| = |T| + |X - S|$. Comme aucun mariage ne peut être plus grand que la couverture $T \cup (X - S)$, M est un mariage maximum.

7. Si pour chaque sommet $x \in T$ on se souvient du chemin $\gamma(x)$, on pourra construire un chemin M-augmentant dans le cas où M n'est pas maximum, et donc on pourra retourner un «mariage augmenté» de taille supérieure. Cela donne :

   $S = U$
   $T = \emptyset$
   tant qu'il y a dans S des sommets non marqués faire :
       Choisir un sommet $x \in S$ non marqué.
       Pour chaque voisin y de x tel que $y \notin T$ et $xy \notin M$ faire :
           Si y est insaturé, arrêter en retournant un mariage augmenté
   `/* non détaillé */`
           Sinon, soit $w \in X$ tel que $yw \in M$.
           Ajouter y à T ; precedent(y)=x
           Ajouter w à S ; precedent(w)=y
       Marquer x.
   Arrêter en répondant «$T \cup (X - S)$ est une couverture minimum
   et M un mariage maximum».

   Le mariage augmenté se construit à partir d'un chemin M-augmentant grâce à la réponse à la question 3, et celui s'obtient grâce au tableau des précédents.

   Il suffit d'appliquer itérativement cet algorithme (en partant par exemple du mariage vide) pour construire des mariages de plus en plus grand, jusqu'à atteindre un mariage maximum. Le nombre d'itérations est majoré par $\min(|X|, |Y|)$.

8. Si M n'est pas maximum l'agorithme de la question précédente construit un chemin augmentant, donc la réciproque est vraie également.

Mariages stables

1. Il n'y a que trois internats possibles.
   - $(a, b), (c, d)$ instable : b, c sont perturbateurs.
   - $(a, c), (b, d)$ instable : a, b sont perturbateurs.
   - $(a, d), (b, c)$ instable : a, c sont perturbateurs.
2. - itération 1 : a rejette y, c rejette z.

– itération 2 : c rejette y.

– itération 3 : x épouse a, y épouse b, z épouse d, w épouse c.

3. On remarque qu'une femme qui a demandé un homme en mariage continue à demander le même homme à toutes les itérations ultérieures jusqu'à être rejetée, ou mariée à la fin de l'algorithme. Tout homme demandé en mariage à une certaine itération est donc demandé également à toutes les itérations ultérieures (éventuellement par des femmes différentes). Ceci permet de montrer qu'à chaque étape toutes les femmes font une demande (a priori on pourrait imaginer qu'une femme soit rejetée par tous les hommes, et elle ne pourrait plus faire de demandes aux itérations ultérieures). En effet, supposons par l'absurde qu'une femme h est rejetée par tous les hommes. Aux itérations ultérieures tous les hommes reçoivent une demande puisqu'ils ont un jour été demandées par h. C'est impossible sinon il y aurait plus de femmes que d'hommes.

   Terminaison : au départ la somme des longueurs des listes de chaque femme est n^2. Si à la fin d'une itération la condition d'arrêt n'est pas vérifiée c'est qu'au moins un homme est demandé par plusieurs femmes (car comme on l'a vu, à chaque étape toutes les femmes font une demande). Dans ce cas au moins une femme est rejetée, et la somme des longueurs des listes des hommes demandables diminue d'autant. Il ne peut donc y avoir plus de n^2 itérations. L'algorithme termine donc toujours, et produit un mariage.

   Stabilité : Supposons que h et f' sont tels que f' préfère h à son conjoint. C'est donc que h a rejeté f'. Et comme h préfère son conjoint à toutes celles qu'il a rejetées, alors h préfère en particulier son conjoint à f', et il n'y a donc pas de danger d'adultère.

4. Supposons que M n'est pas optimal. C'est qu'il existe un mariage stable M' et un couple (f, h') marié dans M' tel que h' a rejeté f dans l'algorithme. Si h' a rejeté f c'est qu'il a été demandé en même temps par une femme f' qu'il préfère à f. Soit h'' le mari de f' dans M'. Par stabilité de M', f' préfère son mari h'' à h'.

   A partir du couple (f, h') nous avons donc construit un couple (f', h'') vérifiant les mêmes propriétés : f' est marié à h'' dans M' mais h'' a rejeté f' dans l'algorithme (puisque dans M, f' se retrouve au mieux marié à h'). De plus, le rejet de f' par h'' a eu lieu a une itération antérieure au rejet de f par h'. En effet f' ne demande h' en mariage qu'après avoir été rejetée par h''.

   On arrive donc à une contradiction en choisissant le couple (f, h') de sorte que le rejet de f par h' ait lieu le plus tôt possible pendant le déroulement de l'algorithme.

   La propriété de pessimalité du point de vue des hommes découle de celle d'optimalité du point de vue des femmes. Supposons en effet que h est marié à f dans M, et est marié dans un autre mariage stable M' a une femme f' qu'il aime moins que f. Dans M', f est mariée à un homme h' qu'elle préfère à h par stabilité de M'. Mais M ne serait alors pas optimal puisque f y est mariée à h.

Naturellement, on peut obtenir un mariage stable optimal pour les hommes et pessimal pour les femmes, avec l'algorithme inversé dans lequel ce sont les hommes qui font les propositions.

5. Notion de stabilité : c'est essentiellement la même. On dit que le mariage est stable s'il n'existe pas une femme f et un homme h tels que h préfère f à sa situation actuelle (célibataire ou marié à une femme moins bien classée que f) et réciproquement f préfère h à sa situation actuelle.

   L'algorithme aussi est très semblable :

   Répéter
   Chaque femme demande en mariage l'homme le plus haut sur sa liste qui ne l'a pas déjà rejetée.
   Tout homme qui reçoit $k \geq 2$ demandes en mariage rejette les demandes émanant des $k-1$ femmes les plus bas sur sa liste.
   Jusqu'à ce qu'aucun homme ne reçoive plus d'une demande.
   On marie alors chaque homme demandé à la femme qui l'a demandé.

   Propriétés de l'algorithme : soit m le nombre de femmes et n le nombre d'hommes. Le raisonnement de la question 3 montre que l'algorithme termine en au plus mn itérations. Premier cas : $m < n$. Le raisonnement de la question 3 montre que toutes les femmes font une demande à toutes les itérations. L'algorithme va donc marier toutes les femmes mais des hommes resteront célibataires. Dans le deuxième cas ($m > n$) des femmes devront rester célibataires. Mais une femme qui reste célibataire a fait des propositions a tous les hommes (et a été rejetée par tous). Comme on l'a vu à la question 3, tout homme qui reçoit un jour une proposition finit marié, donc l'algorithme marie tous les hommes.

   Stabilité : le même raisonnement montre que l'algorithme produit un mariage stable.

   Le même raisonnement montre que le mariage produit est optimal du point de vue des femmes (par stabilité de M', h'' existe toujours).

6. Soit M le mariage produit par l'algorithme et M' un autre mariage stable. Supposons d'abord que $m > n$. Par optimalité de M, toutes les femmes mariées dans M' sont mariées dans M. Par stabilité le nombre de femmes mariées est le même dans M et M' (il est égal à n). Ce sont donc les mêmes femmes qui sont mariées dans M ou M'. Si $m < n$, il suffit de faire le même raisonnement avec l'algorithme inversé dans lequel ce sont les hommes qui font les propositions.

7. D'après la question précédente on peut supposer que les mariages sont produits par l'algorithme de la question 5, et que la personne est un homme (si c'est une femme, on raisonne avec l'algorithme inverse dans lequel ce sont les hommes qui font les propositions). Avec cet algorithme, un homme est marié si et seulement si il est un jour demandé en mariage. Si un homme n'est pas demandé en mariage, sa liste de préférences n'intervient pas dans le déroulement de l'algorithme et il ne sert donc à rien de la modifier.