TP d'Algo/Complexité/Calculabilité

CIMBE Pierre-Alexandre
LAGNIEZ Jean-Marc
LESNYAK Viktor
RAFIK Ahmed

December 16, 2013

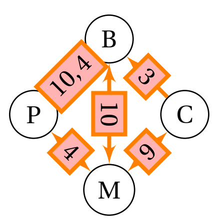
1 Partie théorique

1.1 Partie algorithmique

1.1.1 Exercice 2

Soit un graphe G=(V,E);
avec $\forall i, j \in V$**,**
et f(i,j)-est la fonction representant le flot de l'arc ou $\forall (i, j) \in E$**;**

1. Vrai, car si on enleve un arc (i,j), avec la f(i,j)=0,
   alors on n'influence pas la valeur du flot maximal.
   Ce qui correspondt a la definition de plus petit
   arc vital.
2. Vrai, car dans un flot maximum un arc (i,j) avec le f(i,j)
   minimum est le plus petit arc vital d'apres la
   definition de plus petit arc vital.
3. Faux, il faut montret le contre exemple(mais je galere a faire ce put.n de graph sous latex).



1.1.2 Exercice 3

1. Je suis sur que c'est faux ce que je vais raconter mais je me lance:

   Donc pour la partie trouver un sous-graphe d'un graphe $K2n+1$ qui est isomorphe a une double etoile on peut donner un
   example avec le graphe de petersen(voila le lien : $http://fr.wikipedia.org/wiki/Graphe\_de\_Petersen$);
2. Pour la partie de demonstration : en considerant le +1 (de la formule $K2n+1$) comme le sommet racine, on peut

dire que nous avons un graphe bipartie, et donc on peut trouver une equivalence avec le probleme d'affectation(le probleme de couplage maximum de poids minimum).

P.S: voila j'espere que ca a au moins un minimum de verité.

1.2 Partie complexité

1.3 Partie calculabilité

1.3.1 Exercice 7

1. **Comment enumérer les couples d'entiers?**
2. **Donner les fonctions de codage et de décodage f1 → x et f2 → y**
3. **Montrer que l'on peut coder les triplets. Géneraliser aux k-uplets.**
4. **Pensez-vous que l'on peut coder les éléments de l'intervalle [0,1]. Justifier.**

1. Soit (x,y) $\in \mathbb{N} * \mathbb{N}$, alors faire x + y et trié par ordre lexicographique
2. La fonction de codage est :

$$z = \frac{(x+y)(x+y+1)}{2} + y$$

Pour les fonction de décodage, posons t tel que

$$t = x + y$$

On va prendre t tel que si t augmente de 1 alors

$$\frac{t(t+1)}{2} > z$$

sinon on a

$$\frac{t(t+1)}{2} \leq z$$

La fonction de décodage de y est:

$$z = \frac{t(t+1)}{2} + y$$

$$y = z - \frac{t(t+1)}{2}$$

La fonction de décodage de x est:

$$x = t - y$$

$$x = -z + t + \frac{t(t+1)}{2}$$

$$x = -z + \frac{t(t+3)}{2}$$

3. Pour coder les triplets, il suffit de coder deux entier et coder le résultat et le dernier entier.

$$h(x, y, z) = c(x, c(y, z))$$

   On peut repeter se raisonement pour les k-uplets, ainsi on a

$$k(x1, x2...xk) = c(x1, c(x2, ...c(xk-1, xk))$$

4. On ne peut pas coder les éléments de l'intervalle [0,1] car l'ensemble n'est pas dénombrable. On utilise la diagonal de cantor sur cette ensemble.
   Supposons que l'on puisse numeroter $\mathbb{N} \to [0,1]$ et on en définie la suite S telle que tout éléments de [0,1] soit élément de la suite S. Et on définie un réel r tel que la partie entière est égal à 0 et que chaque décimal en position n est égal à sn(n)[1]+1 si sn(n) est différent de 9 et sn(n)-1 si sn(n) est égal à 9.
   Par construction, r n'est pas dans S sinon on aurait un Sn tel que

$$Sn(n) = r(n) = Sn(n) + 1$$

   ou

$$Sn(n) = r(n) = Sn(n) - 1$$

   C'est absurbe, ainsi ce n'est pas dénombrable.

1.3.2 Exercice 8

1. Les fonctions primitives récursives sont toutes les fonctions que l'on peut construire à partir des fonctions de base pas composition et récursion primitive.

   Exemple

   Soit les fonctions primitives:
   $O \in \mathbb{N}^0$, $\pi_i^k \in \mathbb{N}^k$ et SUC $\mathbb{N}^1$

$$O() = 0$$
$$\pi_i^k(x_1, x_2..., x_k) = x_i$$
$$SUC(x_1) = x_1 + 1$$

   Soit la fonction qu'on utilise pour la récursion primitive:
   $g \in \mathbb{N}^1$

$$g() = SUC(O())$$

   Soit la fonction recursive primitive:
   $f \in \mathbb{N}^1$

$$f(0) = g()$$
$$f(SUC(n)) = \pi_1^2(f(n), n)$$

2. yoloooooooo je ne sais pas

3. (a) Soit la fonction somme défini ainsi:
   Sum $\in \mathbb{N}^2$

$$Sum(0, y) = \pi_1^1(y) = y$$
$$Sum(Suc(x), y) = \pi_2^3(x, Sum(x, y), y)$$

[1] la nème décimal du nème élément de S

(b) Soit la fonction Mult défini ainsi:
Mult $\in \mathbb{N}^2$

$$Mult(O, y) = 0() = 0$$

$$Mult(1, y) = \pi_1^1(y) = y$$

$$Mult(Suc(x), y) = \pi_2^3(x, Sum(Mult(x, y), y), y)$$

(c) Soit la fonction puissance défini aisni:
$X^Y \in \mathbb{N}^2$

$$X^Y(x, 0) = Suc(0()) = 1$$

$$X^Y(x, Suc(y)) = \pi_2^3(x, Mult(X^Y(x, y), x), y)$$

(d) Soit la fonction prédecesseurs tel que:
Pred $\in \mathbb{N}^1$

$$Pred(0) = O() = 0$$

$$Pred(Suc(x)) = \pi_1^2(x, Pred(x))$$

(e) Soit la fonction soustraction tel que:
X-Y $\in \mathbb{N}^2$

$$X - Y(0, y) = 0() = 0$$

$$X - Y(x, 0) = \pi_1^1(x) = x$$

$$X - Y(x, y) = \pi_2^3(x, X - Y(Pred(x), Pred(y)), y))$$

(f) Soit la fonction sg tel que: sg $\in \mathbb{N}^1$

$$sg(0) = 0() = 0$$

$$sg(Suc(x)) = \pi_1^2(1, Suc(x))$$

(g) Soit la fonction $X > Y$ tel que :
X>Y $\in \mathbb{N}^2$

$$X > Y(0, y) = 0$$

$$X > Y(x, 0) = 1$$

$$X > Y(x, y) = \pi_2^3(x, X > Y(Pred(x), Pred(y)), y)$$

Soit la fonction $X \geq Y$ tel que :
X≥Y $\in \mathbb{N}^2$

$$X \geq Y(0, 0) = 1$$

$$X \geq Y(0, y) = 0$$

$$X \geq Y(x, 0) = 1$$

$$X \geq Y(x, y) = \pi_2^3(x, X > Y(Pred(x), Pred(y)), y)$$

4. (a) Voici la fonction d'Ackerman pour $0 \leq m \leq 3$ et $0 \leq n \leq 4$

m/n	0	1	2	3	4
0	1	2	3	4	5
1	2	3	4	5	6
2	3	5	7	9	11
3	5	13	28	58	118

(b) Fesons une preuve par récurrence

$$A_0(n) = Suc(n) = n+1$$

Hypothèse: $A_m(n)$ est primitive récursive
Montrons que $A_{m+1}(n)$ est primitive récursive

Si n = 0, on a que $A_{m+1}(n) = A_m(1)$. D'après l'hypothèse de réccurence, on a que $A_m(n)$ est primitive récursive. Donc $A_{m+1}(n)$ est primitf recursive

Si n > 0, on a que $A_{m+1}(n) = A_m(A_{m+1}(n))$.
Posons n' = $A_{m+1}(n)$. Donc on a $A_m(n')$. D'après l'hypothèse de réccurence, on a que $A_m(n)$ est primitive récursive pour tous $n \in \mathbb{N}$. Donc $A_{m+1}(n)$ est primitif recursive.

(c) Fesons une preuve par récurrence

$$A_0(n) = n+1$$

n+1 > n donc c'est vrai au premier rang
Hypothèse: $A_m(n) > n$
Montrons que $A_{m+1}(n) > n$

Maitenant, on applique une récurrence sur n
n = 0 : $A_{m+1}(1) > 1 > 0$ Hypothèse: $A_{m+1}(n) > n$
Montrons que: $A_{m+1}(n+1) > n+1$
On utilise les deux hypothèse de réccurence:

$$Am+1(n+1) = Am(Am+1(n)) > Am+1(n) > n$$

Ainsi

$$Am+1(n) \geq n+1$$

Donc

$$Am+1(n+1) > n+1$$

On peux donc conclure que $A_m(n) > n$

(d) Il faut montrer que $A_{m+1}(n) - A_m(n) \geq 0$
Fesons une preuve par récurrence sur m

$$A_0(n+1) - A_0(n) = n+1-n = 1$$

Hypothèse: $A_m(n+1) - A_m(n) > 0$
Montrons que $A_{m+1}(n+1) - A_{m+1}(n) > 0$
$A_{m+1}(n+1) = A_m(A_{m+1}(n)) > A_m(n)$
On peut conclure que
$A_m(n+1) - A_m(n) > 0$

(e) Pour n = 0: $A_{m+1}(0) = A_m(1)$. De plus, d'après la question précédente, on a que $A_m(1) > A_m(0)$

Pour n > 0: $A_{m+1}(n) = A_m(A_{m+1}(n-1))$. De plus on a que $A_m(n-1) > n - 1 \rightarrow A_m(n-1) \geq n$
Comme la fonction est strictement croissante, on a que $A_m(A_m(n-1)) \geq A_m(n)$

On peux en conclure que
$A_{m+1}(n) = A_m(A_{m+1}(n-1)) \geq A_m(n)$

(f) D'après les question précédente, on a montré que $A_{m+1}(n) \geq A_m(n)$ et que $A_m(n+1) > A_m(n)$. Ceci prouve que A_m^k est strictement croissante.

(g) Fesons une preuve pas récurrence sur k.
Au cas de base, on a bien $A_{m+1}(n) \geq A_m(n)$
Hypothèse: $A_{m+1}(n + k) \geq A_m^k(n)$
Montrons que : $A_{m+1}(n + k + 1) \geq A_m^{k+1}(n)$
D'après l'hypothèse de réccurence, on a

$$A_m^{k+1} = A_m(A_m^k(n)) \leq A_m(A_{m+1}(n+k))$$

De plus:

$$A_{m+1}(n+k+1) = A_m(A_{m+1}(n+k))$$

On peut conclure que:

$$A_m^{k+1} = A_m(A_m^k(n)) \leq A_{m+1}(n+k+1)$$

(h) Fesons une preuve par l'absurbe, soit la fonction d'Ackermann primitive récursive.

Sois la fonction

$$f : \mathbb{N} \to \mathbb{N} : n \to A(n, 2n)$$

Comme la fonction d'Ackerman est primitive récursive alors f est primitive récursive.