

Examen de mathématiques pour l'informatique SINF1250

Marco Saerens & Bertrand Lebichot
prenom.nom@uclouvain.be

June 2, 2014

Résumé

Il s'agit de l'examen du cours SINF1250 de second baccalauréat en sciences informatiques, UCL. L'examen se déroule en trois heures trente. Il y a cinq questions regroupant à la fois des concepts théoriques et pratiques.

Nous vous demandons de répondre à chaque question sur une feuille différente.

Question 1

Concernant la **logique mathématique** des propositions (30 min).

1. Définissez les notions de *tautologie* et d'*équivalence logique* entre deux propositions.
2. La proposition $[(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (q \rightarrow r)$ est-elle une tautologie ? Une contradiction ? Ou aucun des deux ? Recherchez la réponse en utilisant une table de vérité.
3. La proposition $[p \wedge (p \rightarrow q)] \rightarrow q$ est-elle une tautologie ? Une contradiction ? Ou aucun des deux ? Recherchez la réponse formellement en utilisant l'équivalence logique (sans utiliser les tables de vérité). Bien justifier, expliquer, et préciser, à chaque étape, la formule que vous utilisez.

Question 2

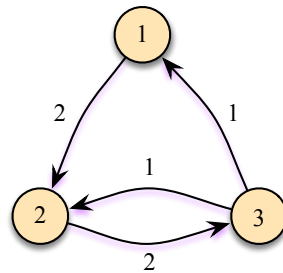
Concernant l'**analyse combinatoire** (30 min).

1. Classer les différents types de combinaisons et permutations à travers un arbre de décision indiquant dans quel contexte chaque technique doit être utilisée.

2. Définissez les notions de r -permutation et de r -combinaison (sans répétition) de r objets parmi n .
3. Démontrez la formule permettant de calculer le nombre de combinaisons de r objets parmi n (sans répétition). Bien détailler et expliquer votre raisonnement.
4. Combien de mots de six caractères (en lettres minuscules) de l'alphabet (26 lettres au total) contiennent
 - simplement six caractères ?
 - la lettre a (au moins une fois) ?
 - les lettres a et b dans des positions consécutives, avec a précédant immédiatement b , et chaque lettre du mot étant distincte ?
 - les lettres a et b , avec a précédant b dans le mot, et chaque lettre du mot étant distincte ?

Bien expliquer votre raisonnement pour chaque sous-question.

Question 3



Concernant la **théorie des graphes**, soit le graphe dirigé et pondéré (les poids sont notés à côté des arcs) ci-dessus (40 min).

1. Donnez sa matrices d'adjacence et calculez sa matrice de probabilités de transition.
2. Expliquez en détail (y compris les équations et dérivations mathématiques) ce que représente le score PageRank et son interprétation en terme de marche aléatoire sur un graphe (définir la notion de téléportation et la Google Matrix à l'aide des équations, mais par contre sans la personnalisation). Justifiez bien chaque étape.
3. Calculez le score PageRank de base (sans personnalisation) de chaque noeud du graphe sans inclure de téléportation ($\alpha = 1$). Détaillez et justifiez vos calculs.
4. Que se passerait-il au niveau du score PageRank si un des noeuds devenait absorbant (réponse en une phrase) ?

Question 4

Concernant les **équations de récurrence** linéaires (40 min).

1. Énoncez et démontrez le théorème fondamental permettant de résoudre une équation de récurrence linéaire non-homogène à coefficients constants. Quelle est la forme générale de sa solution ?
2. Résolvez l'équation de récurrence $a_{n+1} + 16a_{n-3} = 8a_{n-1}$ en déterminant sa solution générale. Bien expliquer votre raisonnement.
3. Un crapaud monte un escalier de $N \geq 2$ marches sans jamais revenir en arrière. A chaque étape, il atteint soit la marche suivante, soit il effectue un bond de deux marches (il passe une marche). Notons a_n le nombre de façon différentes d'arriver à la n -ième marche. Déduisez
 - Les conditions initiales du problème pour $n = 1$ et $n = 2$, sachant que $n = 0$ correspond au niveau du sol.
 - La relation de récurrence permettant de calculer ce nombre a_n , sans la résoudre. Justifiez de manière précise votre réponse.
 - Avez-vous déjà rencontré cette équation ? Si oui, donnez son nom.

Question 5

Concernant les **arbres** et la **programmation dynamique** (40 min).

1. Expliquez le but et la procédure utilisée par l'algorithme de codage de Huffman. Ensuite, détaillez son pseudo-code.
2. Démontrez la formule de programmation dynamique permettant de résoudre un problème à niveaux, sans oublier d'énoncer les hypothèses.
3. Calculez l'edit-distance entre les deux séquences "ADAND" et "DNAD-NAA" et détaillez le tableau de programmation dynamique. Utilisez des pénalités de +1 pour les erreurs (distance de Levenshtein).

Les énoncés de l'examen sont maintenant terminés. Nous vous souhaitons

Bonne chance !!
