

UQÀM Université du Québec à Montréal

INF-3105, Structures de données et algorithmes
— 2009

Examen final
— 2009

CONSIGNES

- **Les règlements de l'UQAM concernant le plagiat seront strictement appliqués.**
- Aucune sortie n'est permise durant l'examen.
- Il est important de bien expliquer vos choix s'il y a lieu.
- Aucun document n'est permis.
- La durée de l'examen est de 3 heures.
- Vous pouvez utiliser les versos comme brouillon ou comme espace supplémentaire.
- **Il est interdit de dégrafer le questionnaire.**
- Les téléphones cellulaires, calculatrices, ordinateurs, palm, baladeurs, iPods, etc. sont interdits.

#1 _____ / 0

#2 _____ / 0

#3 _____ / 0

#4 _____ / 0

#5 _____ / 0

#6 _____ / 0

#7 _____ / 0

IDENTIFICATION

NOM : _____

PRÉNOM : _____

CODE PERMANENT : _____

SIGNATURE : _____

GROUPE : _____

PROFESSEUR : _____

TOTAL

_____ / 0

commentaire :

Numéro 1. (0 pts)

Objectif(s) :

- Synthèse de la matière.
- Algorithme de Prim.

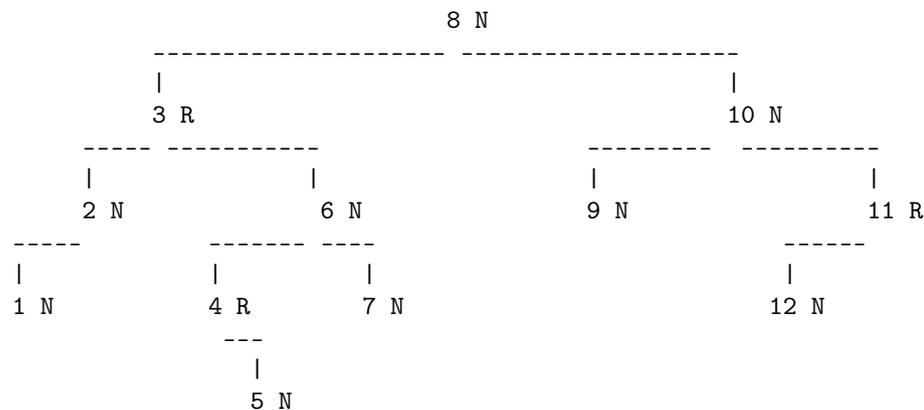
Question : Expliquez le fonctionnement de l'algorithme de Prim.

Numéro 2. (0 pts)

Objectif(s) :

- Application des connaissances.
- Arbre RN.

Question : Soit l'arbre AVL suivant :



a) (0 pts) Supprimez l'élément 9 et donnez l'arbre résultant.

b) (0 pts) Dans l'arbre de base, avant la suppression de 9, supprimez l'élément 1.

Numéro 3. (0 pts)

Objectif(s) :

- Application des connaissances.
- Arbre rouge-noir.

Question : Construisez l'arbre rouge-noir qui résulte de l'ajout des valeurs suivantes dans un arbre vide. Les valeurs sont ajoutées dans l'ordre où elles sont présentées.

50 30 55 62 24 35 90 46 74 94 1 24 51

Numéro 4. (0 pts)

Objectif(s) :

- Application des connaissances.
- Algorithme de Floyd.

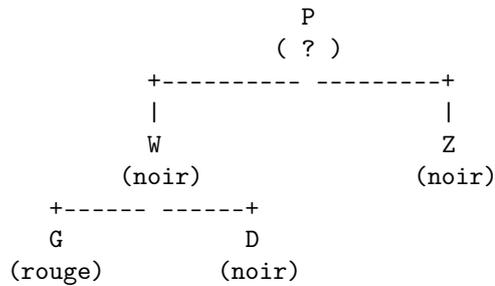
Question : Expliquez précisément le rôle que peut prendre un monceau dans l'algorithme de Kruskal.

Numéro 5. (0 pts)

Objectif(s) :

- Analyse de problème.
- Arbre Rouge-Noir.

Soit le quatrième cas de la suppression dans les arbres rouge-noir :



où P, W, G, D et Z sont des noeuds. Les couleurs sont entre parenthèse. La réduction à eux lieu à droite de P.

a) (0 pts) Décrivez les modifications à faire pour rééquilibrer l'arbre.

b) (0 pts) Expliquez le changement de nombre de noeud noir. Décrivez les différents nombres de noeud noir de chaque chemin des sous-arbres et noeuds avant et après modifications.

Numéro 6. (0 pts)

Objectif(s) :

- Application des connaissances.
- Représentation des graphes.
- stl.

Soit les déclarations suivantes :

```

#include<string>
#include<map>
#include<vector>

using namespace std;

template< typename T >
class Sommet {
public :
    string identification;
    T information;
};

template< typename T >
class ListeAdjacence {

```

```
private :
    vector< Sommet< T > > _liste;
public :
    void ajouterElement( const Sommet< T > & a_sommet );
    bool appartient( const Sommet< T > & a_sommet ) const;
};

template< typename T >
class Graphe {
private :
    map< Sommet< T >, ListeAdjacence< T > > description;
public :
    void ajouterArc( const Sommet< T > & a_sommet1,
                    const Sommet< T > & a_sommet2 );
    vector< Sommet< T > > voisins( const Sommet< T > & a_sommet ) const;
    bool sontVoisins( const Sommet< T > & a_sommet1,
                     const Sommet< T > & a_sommet2 );
};
```

a) (0 pts) Écrivez le code pour la fonction `ListeAdjacence::appartient`.

b) (0 pts) Écrivez le code pour la fonction `Graphe::sontVoisins`.

Numéro 7. (0 pts)

Objectif(s) :

- Synthèse de la matière.
- Monceau.

a) (0 pts) Expliquez comment un monceau descendant (maximum en tête) est représenté. Donnez les relations entre père et fils.

b) (0 pts) Expliquez comment un élément est ajouté dans un monceau et donnez l'ordre temporel de l'algorithme.

c) (0 pts) Expliquez comment un élément est enlevé dans un monceau et donnez l'ordre temporel de l'algorithme.