Laboratoire 2
Algorithmes de recherche,
Choix heuristique et Calcul de temps amorti

ELE440 – Algorithmes
Automne 2015

Pondération : 12 points

I - Objectifs

Ce laboratoire porte sur l'utilisation de l'analyse asymptotique pour le choix des algorithmes ainsi que sur les principaux algorithmes de recherche de données. Son but est de :

1. utiliser les résultats des analyses asymptotiques pour le choix heuristique des algorithmes ;

2. se familiariser avec la mesure de temps amorti dans l'évaluation des performances des algorithmes, et

3. se familiariser avec quelques-uns des principaux algorithmes de recherche de données, aussi bien quant à leur fonctionnement qu'à leur performance.

II - Implémentation (50 %)

Cette activité vise à modifier et adapter le programme du laboratoire 1 pour lui permettre de tester cinq (05) algorithmes de recherche dans des tableaux de données. Les détails du programme sont donnés ci-dessous.

1. Algorithmes de recherche :

Le programme doit contenir les cinq algorithmes de recherche suivants :

- recherche séquentielle
- recherche binaire
- recherche « *optimisée* »[1]
- table de hachage
- arbre de recherche

Chaque routine de recherche reçoit le tableau de données T1, sa taille N, le rang R et le taux de désordre D de ses données. Le rang R doit impérativement être supérieur ou égal à la taille N pour éviter les doublons. Chaque routine de recherche reçoit aussi le tableau T2 de la liste des données à rechercher et sa taille K. Finalement, chaque routine retourne un tableau T3 contenant l'index du tableau T1 de chaque donnée trouvée (index = -1 pour chaque donnée non trouvée).

[1] Voir description à la page 2 (note b).

Par exemple, soit le tableau de données T1 suivant :

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
T1 =	23	15	46	11	17	6	27	12	18	5	7	43	39	3	16

Et le tableau de la liste des données à rechercher T2 suivant :

	0	1	2	3	4	5
T2 =	6	43	28	15	13	11

Alors le tableau T3 retourné par la routine sera :

	0	1	2	3	4	5
T3 =	5	11	-1	1	-1	3

Rappel : Les algorithmes peuvent provenir de sources quelconques (telles que les livres, articles et sites web, etc.), pourvu que leur provenance soit clairement identifiée. Ne pas citer ses références sera considéré du plagiat !

Notes importantes :

a) Recherche binaire :

La recherche binaire exige que les données soient triées. Ainsi, votre algorithme de recherche binaire doit choisir automatiquement un des six (06) algorithmes de tri du laboratoire 1 pour trier les données avant de procéder aux recherches.

Votre algorithme doit choisir la méthode de tri la plus avantageuse en rapport avec les caractéristiques des données. Ainsi, si le degré de désordre D est faible, il peut être plus avantageux d'utiliser le tri par insertion, par contre si le rang R est petit, le tri pigeonnier pourrait être la méthode de choix.

Servez-vous de l'analyse des algorithmes de tri que vous avez déjà faite lors du laboratoire 1.

b) Recherche « *optimisée* » :

La méthode de recherche « *optimisée* » vise à choisir le meilleur algorithme de recherche, parmi les quatre autres algorithmes, selon les caractéristiques des données (N, R, D) et le nombre de recherches à faire (K).

Ainsi, si K est petit, il peut être plus avantageux de faire une recherche séquentielle, tandis que si K et D sont grands, la table de hachage ou l'arbre de recherche pourraient être plus intéressants.

Il vous incombe de créer la stratégie la plus efficace - voir l'ANNEXE 1 (p. 7).

c) Table de hachage et arbre de recherche :

Le choix de la fonction de hachage doit être fait intelligemment, selon les caractéristiques des données, de façon à obtenir un **facteur de charge** entre 0.5 et 1.

L'arbre de recherche doit être construit de façon à être bien équilibré.

2. Tableaux de données[2] *:*

- De la même manière qu'au laboratoire 1, le tableau de données T1 est généré automatiquement par le programme ou lu à partir d'un fichier. Par contre, contrairement au laboratoire 1, **aucun doublon ne doit être accepté** (le rang et la taille du tableau T1 doivent donc être appariés en conséquence).

- La liste des données à rechercher (i.e., tableau T2) est lue dans un fichier de type texte. La première valeur lue dans le fichier est le nombre de données à rechercher K (qui est aussi la taille du tableau T2) suivie des K données à rechercher. Lorsque la requête est effectuée à partir du clavier, vous pouvez limiter la lecture à une seule donnée.

3. Fonctionnalités générales :

- Le programme permet à l'usager de choisir la source des données (soient générées automatiquement ou bien lues à partir d'un fichier) et l'algorithme de recherche qui sera utilisé.

- À la fin de la recherche, le programme affiche à l'écran les statistiques suivantes :
  - Nom de la méthode de recherche
  - Nom de la méthode de tri (si nécessaire, voir recherche binaire)
  - Taille N, rang R et degré de désordre D des données (tableau T1)
  - Nombre de recherches faites et nombre de valeurs trouvées
  - Performance (évaluée à l'aide des baromètres suivants) :
    - $T_p(N)$: Temps de préparation (tri, hachage ou construction d'arbre)
    - $T_r(N)$: Temps total de recherche (pour les K recherches)
    - $T(N)$: Temps total où $T(N) = T_p(N) + T_r(N)$
    - $T_a(N)$: Temps amorti par recherche $T_a(N) = {}^{T(N)}/_{K}$

- De plus, le programme sauvegarde ces statistiques ainsi que les tableaux T1, T2 et T3 dans un fichier texte sous le format suivant :
  - Les statistiques décrites ci-dessus
  - Le tableau T1
  - Le tableau T2
  - Le tableau T3

[2] Plus de détails à propos du format des fichiers de données et de requête sont fournis dans l'ANNEXE 2 (p. 8).

III - Analyse des performances (30 %)

L'analyse de la performance des algorithmes de recherche est strictement théorique et utilise le principe des baromètres. L'analyse doit contenir les trois aspects suivants :

- *Temps de préparation* $T_p(N)$ *:* (10 %)

  Le temps de préparation à la recherche correspond au temps nécessaire pour préparer les données avant que la recherche commence. Par exemple, le temps nécessaire pour construire la table de hachage ou l'arbre de recherche. Pour la recherche binaire, cela correspond au temps nécessaire pour trier les données selon l'un des six algorithmes de tri. Pour ce dernier, vous pouvez simplement importer les résultats de vos analyses du laboratoire 1. Vous devez justifier vos calculs et résultats (sauf pour les tris).

- *Temps de recherche* $T_r(N)$ *:* (10 %)

  Le temps de recherche est le temps nécessaire pour rechercher une valeur, selon l'algorithme utilisé. Il n'inclut pas le temps de préparation décrit ci-dessus.

- *Temps de recherche amorti* $T_a(N)$ *:* (10 %)

  Le temps de recherche amorti est le temps moyen requis par recherche pour une liste de requêtes de taille K. Dans ce calcul, le temps de préparation des données est réparti (amorti) sur les K recherches. Ainsi, le temps de recherche amorti est obtenu par l'équation suivante :

$$T_a(N) = T_r(N) + \frac{T_p(N)}{K}$$

IV - Rapport (20 %)

Le rapport est écrit **dans vos mots** et doit contenir les sections suivantes (la longueur de chaque section est donnée à titre indicatif ; l'important est de fournir les informations demandées de manière précise et concise) :

1. *Introduction* (maximum 1 page)

Cette première section contient une description des buts du laboratoire et des objectifs visés. Elle sert à introduire les sections qui suivent.

2. *Les algorithmes de recherche* (environ 1 à 2 pages par algorithme)

La deuxième section explique le principe de fonctionnement de chaque algorithme et donne son pseudocode, **sans oublier de mentionner sa provenance**. Si des modifications ont été apportées aux algorithmes suggérés, il faut les expliquer et les justifier. Cette section donne aussi un court rapport sur les difficultés et autres observations intéressantes rencontrées lors de l'implémentation.

3. *L'analyse théorique* (maximum 1 page par algorithme)

Dans la troisième section, il faut fournir les détails de l'analyse théorique de chaque algorithme et justifie les conclusions de l'analyse. Cette section indique aussi les baromètres retenus (et non ceux qui ont été éliminés). Il faut distinguer clairement les formules exactes (ex. nombre d'exécutions, E) et les formules asymptotiques.

4. *Choix heuristique* (maximum 2 pages)

Pour la recherche binaire, il faut expliquer dans cette section comment sont choisis les algorithmes de tri en tenant compte des caractéristiques des données. Pour la recherche « *optimisée* », il faut justifier les motifs du choix de la stratégie de recherche selon les caractéristiques des données et le nombre de requêtes.

5. *Conclusion* (environ 1 à 2 pages)

Dans cette section, il faut présenter sa conclusion quant à l'atteinte des objectifs de départ ainsi que sur les avantages et les inconvénients de chaque méthode de recherche.

6. *Références*

Cette dernière section contient les références de vos sources. Dans le corps du rapport, vous devez également mettre un renvoi après chaque élément emprunté, ex. [1], [2]...

Remarque : Si une information provient de l'énoncé de laboratoire ou du matériel de cours, il n'est pas nécessaire de citer cette référence.

Si le document cité est un volume :

1. De Garmo, E.P., Sullivan, W.G. & Bontadelli, J.A. (1989). Engineering Economy (8e ed.). New York : MacMillan.

Si le document cité provient d'un site web :

2. École de technologie supérieure. Politique d'éthique de la recherche avec des êtres humains, [En ligne]. http://www.etsmtl.ca/SG/Politique/polethsh.pdf (Consulté le 14 novembre 2000).

Si le document cité est un article de périodique :

3. Gargour, C.S., Ramachandran, V., Bogdadi, G. (1991). Design of Active RC and Switched Capacitor Filters Having Variable Magnitude Characteristics Using a Unified Approach. J. of Computers and Electrical Engineering, 17(1), 11-12.

De plus, vous devez fournir au chargé de laboratoire le *code source de votre programme* ainsi que vos *fichiers de résultats* de l'analyse expérimentale en **version électronique**.

Note : Un soin particulier doit être accordé au français dans la rédaction de ce rapport. Les rapports mal écrits seront pénalisés jusqu'à concurrence de 10 %.

V - Échéancier

Semaine 1 :

- Modification du programme du laboratoire 1 :
  - Modification du squelette du programme (exemples : menu, conserver l'index initial des données lors des tris → tableau de structures)
  - Ajout de la lecture des requêtes.
- Recherche d'algorithmes de recherche (notes de cours, livres, Internet, etc.).
- Implémentation de la recherche séquentielle et de la recherche binaire.

Semaine 2 :

- Implémentation et validation des algorithmes de recherche.
- Implémentation des heuristiques pour le choix du tri avec la recherche binaire et pour le choix de la meilleure stratégie de recherche pour la recherche « *optimisée* ».

Semaine 3 :

- Finaliser l'implémentation et la validation des heuristiques.
- **Le fonctionnement complet du programme est démontré au chargé de laboratoire (voir la note ci-dessous).**

Semaine 4 :

- Remise du rapport de laboratoire au début de la séance de laboratoire.

Note importante concernant la démonstration de la troisième semaine :

Pour que la démonstration du code source de chaque groupe se passe dans les meilleures conditions possibles, voici quelques informations à propos de cette activité obligatoire :

- Une démonstration est requise par **groupe**.
- Chaque démonstration est **notée**. La note attribuée vaut 50% de la note globale du deuxième laboratoire.
- Au moins une personne doit être présente durant la séance laboratoire de la troisième semaine pour faire la démonstration de son groupe. L'absence de tous les membres d'un groupe pourrait valoir la note zéro.
- Lors de la démonstration, le fonctionnement du programme tel que décrit dans le protocole d'implémentation (voir la section II « *Implémentation* », pp. 1-3) est validé. Une attention particulière sera accordée à l'implémentation des cinq algorithmes de recherche ainsi qu'au programme d'affichage des statistiques.
- Le chargé du laboratoire vous fournira **deux fichiers de type texte** pour la validation de votre programme. Le premier est un fichier de données. Le second comporte les données à rechercher. Les deux fichiers ont le même format tel que décrit à l'annexe 2 de la page 8.

ANNEXE 1

EXEMPLE D'ANALYSE
Temps de calcul amorti pour K recherches

I - Recherche séquentielle

Temps de recherche d'une donnée : $T_r(N) \in O(N)$

Temps de préparation : $T_p(N) \in \Theta(1)$

Temps amorti pour K recherches : $T_a(N) = \frac{T_p(N)}{K} + T_r(N) \Rightarrow T_a(N) \in O(N)$

II - Recherche binaire

Temps de recherche d'une donnée : $T_r(N) \in \Theta(\log(N))$

Temps de préparation (tri par insertion) : $T_p(N) \in \Theta(N^2)$

Temps amorti pour K recherches : $T_a(N) = \frac{T_p(N)}{K} + T_r(N) \Rightarrow T_a(N) \in \Theta\left(\frac{N^2}{K} + \log(N)\right)$

Donc il sera plus avantageux d'utiliser la recherche binaire avec un tri par insertion qu'une recherche séquentielle lorsque :

(Temps de K recherches séquentielles) > (Temps du tri par insertion) + (Temps de K recherches binaires)

Soit, lorsque :

$$KN > N^2 + K \cdot \log(N) \Rightarrow K\big(N - \log(N)\big) > N^2$$

Donc lorsque :

$$K > \left(\frac{N^2}{N - \log(N)}\right) \approx N$$

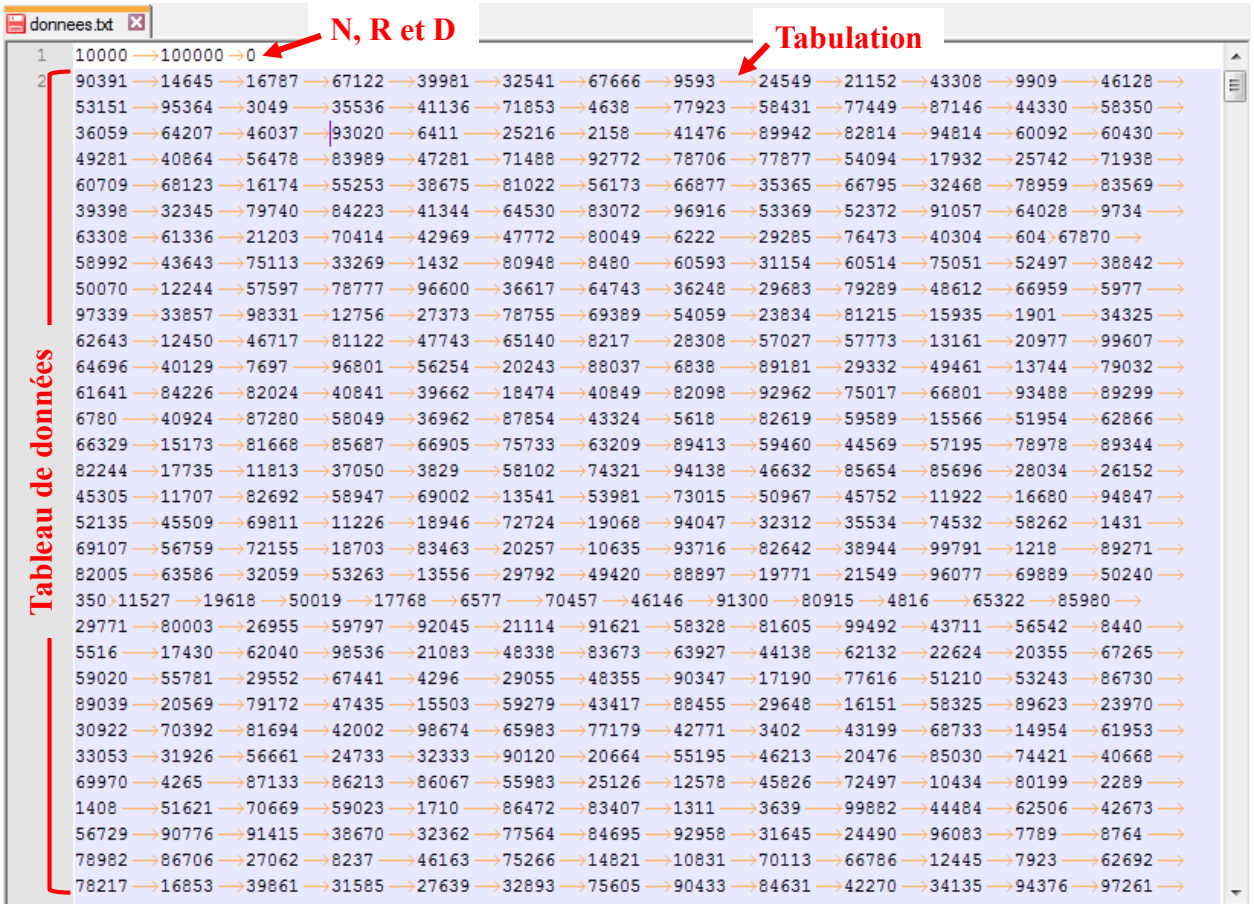
ANNEXE 2

FORMAT DES FICHIERS
Structure des fichiers de données et de requête

I - Fichier de données

Les données consommées par les algorithmes de recherche sont sauvegardées dans un fichier texte qui respecte le format suivant (voir la figure suivante) :

a. La première ligne comporte la taille *N*, le rang *R* et le degré de désordre *D* des données.
b. La deuxième ligne comporte le tableau de données.
c. Aucun doublon n'est accepté dans le tableau de données.
d. Les données sur les deux lignes sont séparées par le caractère de tabulation (i.e., « \t »).



II - Fichier de requête

La structure du fichier de requête est similaire à celle du fichier de données à l'exception de la première ligne qui ne comporte que la taille *K* des données à rechercher.

ANNEXE 3

GRILLE DE CORRECTION DU LABORATOIRE

Rapport + Analyse	
La pondération indiquée ci-dessous tient compte de la forme du rapport (20%) et de l'analyse de performance (30%).	
1. Introduction **0.5**	
2. Algorithmes **6**	
3. Analyse de performance **26** (= 6 + 20)	
4. Choix heuristiques **16** (= 6 + 10)	
5. Conclusion **1.5**	
Annexes	
Références	
Total partiel /**50**	
Points négatifs	
Rapport incomplet : • Une section manquante : -50% • Plus d'une section manquante : -10% additionnels par section.	
Références manquantes (max. -10%)	
Orthographe et grammaire de mauvaise qualité (max. -10%)	
Mauvaise présentation (max. -10%)	
Non-respect du gabarit (max. -10%)	
Non-respect du protocole de livraison (max. -10%)	
Retard (max. -10% par jour)	
Note Rapport + Analyse /**50**	
Programmes	
Implémentation[3] /**50**	
Points négatifs	
Non-respect du protocole de livraison[4] (max. -10%)	
Retard (-10% par jour)	
Note programmes /**50**	
Note finale laboratoire /**100**	

[3] L'implémentation est validée par une démonstration lors de la séance laboratoire de la troisième semaine.
[4] Une version électronique des programmes et des fichiers résultats doit être remise avec le rapport.