IFT 2125 Automne 2014
Introduction à l'algorithmique

Pierre McKenzie

EXAMEN INTRA

Le jeudi 16 octobre, 11:30–13:15

Directives :

- Aucune documentation n'est permise. *No documentation.*
- Répondez sur le questionnaire. *Answer on the question sheets, in the space provided within the French section of the exam.*
- *See the last two pages for an English translation. Do* not *write your answers on those two pages, which will be discarded.*

1. ______ /30
2. ______ /15
3. ______ /15
4. ______ /20
5. ______ /20

Total : ______ /100

Nom : ________________________________ **Code permanent :** ________________

Votre place dans la salle Z-310 *(your seat number in room Z-310)* : ________________

1. **(30 points)** Soit $f : \mathbb{N} \to \mathbb{R}^{\geq 0}$.

(a) (5 pts) Donnez la définition formelle complète de $O(f(n))$.

(b) (5 pts) Est-ce qu'on a forcément $O(2f(n)) = O(f(2n))$? (Expliquez.)

(c) (5 pts) Lorsque vous devez estimer le temps d'exécution en pire cas d'un algorithme, est-il préférable de déterminer son O, son Ω ou son Θ ? (Expliquez.)

(d) (5 pts) Est-ce que $O(n^4) \subseteq \Omega(n^2)$? (Justifiez.)

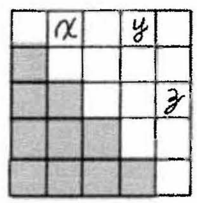
(e) (5 pts) Un algorithme A prend un temps $5m+3$ sur son entrée de longueur m. Un algorithme B sur son entrée de longueur n ne fait rien d'autre qu'appeler A itérativement avec les valeurs $m = 1, m = 2, m = 4, m = 8, \ldots, m = 2^n$. Donnez une fonction simple $t(n)$ telle que le temps d'exécution de B est dans $\Theta(t(n))$.

(f) (5 pts) Est-ce que $(\{1, 2, 3, 4, 5\}, I)$ est un matroïde lorsque I est l'ensemble des sous-ensembles de $\{1, 2, 3, 4, 5\}$ qui contiennent $\{1, 2, 3\}$? (Justifiez.)

2. **(15 points)** Cette question concerne des permutations de $\{1, 2, 3, 4, 5\}$.

   (a) (5 pts) Quel est le produit de permutations $(12) \times (124) \times (34)$?

   (b) (5 pts) Le tableau ci-dessous résulte de l'algorithme étudié en cours. Seules les entrées x, y et z du tableau diffèrent de la permutation identité à la fin de l'exécution complète de l'algorithme.

   Combien de permutations peuvent être engendrées en multipliant entre elles x, y et z de toutes les façons possibles ?

   (c) (5 pts) L'information fournie en (b) vous permet-elle d'identifier x, y et z de manière unique ? (Si oui, donnez x et y et z ; si non, justifiez.)

3. **(15 points)** Considérez le mot anaconda.

   (a) (6 pts) Tracez un arbre A servant à construire un code de Huffman de ce mot.

   (b) (3 pts) Donnez le code de ce mot selon votre arbre A.

   (c) (6 pts) Supposons qu'un arbre binaire B possède un sommet (autre qu'une feuille) qui possède un seul fils. Est-ce que le code du mot anaconda obtenu de B peut être plus court que votre code ci-dessus ? (Justifiez.)

4. **(20 points)** L'entreprise **FUTURE SHIP** propose des voyages dans l'espace à bord de sa capsule monoplace. Chaque client fournit à l'entreprise la durée (un nombre entier de jours) du séjour qu'il souhaite effectuer dans l'espace. L'entreprise ne possède qu'une capsule. La technologie WWIN™ utilisée pour la conception de la capsule possède l'inconvénient [1] que la capsule se désintègre dès qu'elle cumule 365 jours et une minute de vol.

   (a) (15 pts) Esquissez une méthode vorace

   fast_fire(souhaits) :

   ''' souhaits[i] est le souhait du client i et fast_fire retourne un tableau D où $D[i]$ est le jour du départ du client i, $0 \leq D[i] \leq 365$, 0 signifiant départ refusé '''

   (en pseudo-code ou en Python) permettant à **FUTURE SHIP** de maximiser le nombre de voyages effectués, sans perte de vie humaine.
   Vous pouvez supposer l'accès à une méthode de tri.

   (b) (5 pts) Identifiez trois composantes de votre méthode qui en font un algorithme vorace.

1. La technologie "we want it now" s'accompagne souvent d'effets de bord indésirables.

5. **(20 points)** Considérez la méthode Python

```
from math import floor,log2
def myst(a,b) :
    if (a < 4 and b < 4) :
        return (a+b)
    else :
        m = int (floor(log2(max(a,b)))+1)
        shift = m // 2
        r = (2 ** shift)
        s,t = a // r , a % r
        v,w = b // r , b % r
        return (r*myst(s,v) + myst(t,w))
```

(a) (6 pts) Combien de nouveaux appels à myst sont engendrés par l'appel
myst$(4,4)$?

myst$(16,8)$?

(b) (3 pts) Que calcule myst lorsque a et b sont des entiers naturels ?

(c) (3 pts) En supposant qu'une exécution de myst excluant les appels récursifs prend un temps dans $\Theta(m)$, posez une récurrence asymptotique décrivant l'ordre exact du temps $t(m)$ que prend myst(a,b) lorsque $\max(a,b)$ possède m chiffres binaires.

(d) (3 pts) Donnez sans preuve une fonction $f(m)$ simple telle que $t(m) \in \Theta(f(m) \mid m$ est puissance de 2).

(e) (5 pts) Que pouvez-vous dire de $\Theta(t(m))$? (Pourquoi ?)

BONNE CHANCE !

English translations. Please do not write your answers here.

1. **(30 points)** Let $f : \mathbb{N} \rightarrow \mathbb{R}^{\geq 0}$.
   (a) (5 pts) Give the complete formal definition of $O(f(n))$.
   (b) (5 pts) Does $O(2f(n))$ necessarily equal $O(f(2n))$? (Explain.)
   (c) (5 pts) When asked to estimate the worst case execution time of an algorithm, is it preferable to determine its O, its Ω or its Θ ? (Explain.)
   (d) (5 pts) Is $O(n^4) \subseteq \Omega(n^2)$? (Justify.)
   (e) (5 pts) An algorithm A takes time $5m + 3$ on an instance of size m. An algorithm B on an input of size n does nothing other that call A iteratively on instances of size $m = 1, m = 2, m = 4, m = 8, \ldots, m = 2^n$. Give a simple function $t(n)$ such that the execution time of B belongs to $\Theta(t(n))$.
   (f) (5 pts) Is $(\{1,2,3,4,5\}, I)$ a matroid when I is the set of subsets of $\{1,2,3,4,5\}$ that contain $\{1,2,3\}$? (Justify.)
2. **(15 points)** This question concerns permutations of the set $\{1,2,3,4,5\}$.
   (a) (5 pts) What is the product of the permutations $(12) \times (124) \times (34)$?
   (b) (5 pts) The table

	x		y	
				z

   results from the algorithm seen in class. The entries x, y and z are the only table entries that differ from the identity permutation at the end of the complete algorithm.
   (c) (5 pts) How many permutations can be generated by multiplying together x, y and z in all possible ways ?
   (d) (5 pts) Does the information provided in (b) uniquely determine x, y and z ? (If so, give x and y and z ; otherwise, explain.)
3. **(15 points)** Consider the word anaconda.
   (a) (6 pts) Draw the tree A used to construct a Huffman code for this word.
   (b) (3 pts) Give the code for this word according to your tree A.
   (c) (6 pts) Suppose that a binary tree B has a node (other than a leaf) with a single child. Can the code prescribed by such a tree B for the word anaconda be shorter than your code above ? (Justify.)
4. **(20 points) FUTURE SHIP** proposes space travel aboard its one-person shuttle. A customer who wishes to travel requests a duration (number of days) for his trip. The company owns a single shuttle. The WWIN™ technology used in the shuttle design phase has one drawback : the shuttle disintegrates after a cumulative period of precisely 365 days and one minute in space. [2]
   (a) (15 pts) Sketch a greedy method (in pseudo-code or Python)
   fast_fire(souhaits) :

2. The "we want it now" technology often entails similar undesirable features.

''' souhaits[i] is the wished duration of customer i ; fast_fire returns a table D where $D[i]$ is the departure day for customer i, $0 \leq D[i] \leq 365$, 0 meaning no departure '''

that **FUTURE SHIP** can use to maximize the number of space trips for its shuttle, with no loss of human life.

You may assume that a sorting method is available.

(b) (5 pts) Identify three features of your algorithm that makes it a greedy algorithm.

5. **(20 points)** Consider the Python method

```
from math import floor,log2
def myst(a,b) :
    if (a < 4 and b < 4) :
        return (a+b)
    else :
        m = int (floor(log2(max(a,b)))+1)
        shift = m // 2
        r = (2 ** shift)
        s,t = a // r , a % r
        v,w = b // r , b % r
        return (r*myst(s,v) + myst(t,w))
```

(a) (6 pts) How many further calls to myst does the following trigger : myst$(4,4)$? myst$(16,8)$?

(b) (3 pts) What does myst compute when a and b are natural numbers ?

(c) (3 pts) Assuming that one run of myst excluding recursive calls takes time in $\Theta(m)$, set up an asymptotic recurrence describing the exact order of the run time $t(m)$ of myst(a,b) when $\max(a, b)$ expressed in binary has m digits.

(d) (3 pts) Without proof, give a simple function $f(m)$ such that $t(m) \in \Theta(f(m) \mid m \text{ is a power of } 2)$.

(e) (5 pts) What can you say about $\Theta(t(m))$? (Why ?)

GOOD LUCK !