Université d'Ottawa
Faculté de génie

École d'ingénierie et de technologie de l'information (EITI)

uOttawa
L'Université canadienne
Canada's university

University of Ottawa
Faculty of Engineering

School of Electrical Engineering and Computer Science (EECS)

CSI 2510

Examen de mi-session

Aucune documentation permise – Pas de calculatrice

2 heures – 30% - 20 questions – 100 points

Q		Q	
1	6	11	3
2	5	12	4
3	2	13	8
4	3	14	8
5	4	15	6
6	6	16	6
7	6	17	8
8	4	18	6
9	2	19	5
10	6	20	2
TOTAL **100**			

Nom : SOLUTIONS

Numéro d'étudiant: ____________________

Signature : ____________________

[Question 1] (6 points)

Pour les fonctions suivantes, donner la meilleure complexité Grand O. Justifier vos réponses:

(1) $n^4 (1+2n)/n^2 + 1039n^3 + 1039n^2$ est O(n^3)

$n^2 + 2n^3 + 1039n^3 + 1039n^2$

(2) $-n + \log_2 (10)^n - \log_2(7n)$ est O(n)

$-n + n \log_2 10 - \log 7n$

$(\log 10 - 1)n - \log 7n$

(3) $n^2 + \sum_{i = 10, 11, .., n} (i^2)$ (for n > 10) est O(n^3)

pour n grand, l'absence des premiers termes est négligeable

$n^2 + (10)^2 + (11)^2 + (12)^2 + \ldots + n^2 \leq n^2 + (n-9)n^2$

[Question 2] **(5 points)**

Le pseudo-code ci-dessous prend en entrée un tableau A de n entiers positifs et utilise une pile S comme structure de donnée interne:

```
t ← 0

for (i=0; i<n-1; i++) do
     if (A[i] mod 2) = 1 then
        S.push(A[i])

while S.size() > 1 do {
   k = S.pop()
   t ← t + k*k
}

return t
```

si impair empile A[i]

(1) Quelle sera la sortie de cet algorithme pour le tableau suivant? [3]

A = {2, 6, 3, 5, 9, 1, 34, 92, 9}

le dernier élément est empilé

$\cancel{3^2+5^2+9^2}$

le dernier élément n'est pas dépilé

$\cancel{9+25+81 = 115}$

$9^2 + 5^2 + 9^2 + 1^2 = 25 + 81 + 1 = \cancel{107}$

$+81 \qquad 188$

(2) Quelle est la complexité Grand O de cet algorithme en fonction de n? [2]

a) $O(n^2)$
b) $O(n \log n)$
c) $O(n)$
d) $O(n^3)$
e) $O(1)$

[Question 3] (2 points)

Quelle sera, <u>au pire cas</u>, la complexité Grand O en fonction de n pour l'algorithme ci-dessous lorsqu'un tableau 2D de dimension n x n est utilisé en entrée? Justifier votre réponse

```
sum = 0;
for (i=0; i<n-1; i++) do
    for (j=0; j<100; j++) do
        for (k=0; k< i*j ; k++) do
            sum = sum + A[i][j]*k
return sum
```

pire cas n fois

100 fois

pire cas 100·n fois

combien de fois cet énoncé est exécuté ?

$$\sum_{i=0}^{n-1} 1i + 2i + 3i + \ldots + 99i$$

$$= \sum_{i=0}^{n-1} i\underbrace{(1+2+\ldots 99)}_{\text{constante } K}$$

$$= K \sum_{i=0}^{n-1} i = K\frac{n(n-1)}{2}$$

$O(n^2)$

donc $O(n^2)$

[Question 4] (3 points)

Démontrer que f(n) = $\sum_{i=1,\ldots,n} (n^2 - i)$ est O(n^3) en utilisant la définition de Grand O.

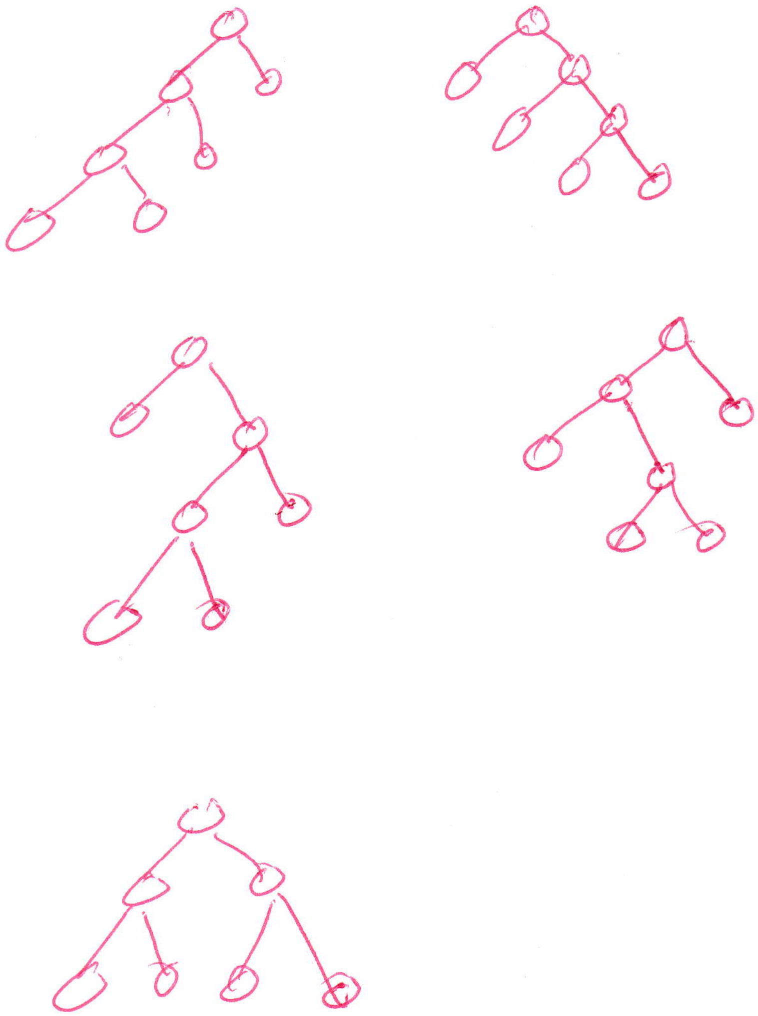
$$= n^2 - 1 + n^2 - 2 + n^2 - 3 + \ldots + n^2 - n$$

$$= n(n^2) - \frac{n(n+1)}{2}$$

$O(n^3)$

[Question 5] (4 points)

Dessiner tous les arbres binaires pleins possibles ayant 7 noeuds.



[Question 6] (6 points)

(a) Dans la représentation tableau d'un arbre monceau min, les éléments sont toujours en ordre décroissant.

VRAI ou FAUX

(b) L'élément de clé maximum dans un monceau-min est toujours un noeud externe.

VRAI ou FAUX

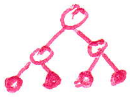
(c) La hauteur de tout arbre binaire ayant n noeuds est toujours O(log n)

VRAI ou FAUX

(d) Le nombre de noeuds externes dans un arbre quelconque est toujours plus grand que le nombre de noeuds internes

VRAI ou FAUX

(e) Le nombre de noeuds externes dans un arbre binaire parfait est toujours plus grand que le nombre de noeuds internes



VRAI ou FAUX

(f) Afin de trier des éléments en ordre décroissant avec le tri par monceau, quel type d'arbre devrait-on utiliser?

Monceau-min ou Monceau-max

[Question 7] (6 points)

Quelle est la complexité Grand O <u>au pire cas</u> de la recherche du plus petit élément dans chacune des structures de données suivantes?

(a) Monceau-max

~~O(n)~~ O(n) c'est l'une des feuilles

(b) Monceau-min

O(1) à la racine

(c) Séquence triée en ordre décroissant dans une liste chainée simple

O(n) à la fin

(d) Séquence non-triée dans une liste chainée double

O(n) en moyenne n/2

(e) Arbre binaire de recherche

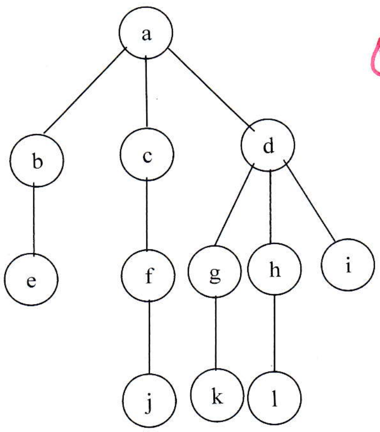
O(n) cas dégénéré

(f) Tableau trié en ordre croissant

O(1) le premier

[Question 8] (4 points)

Soit l'arbre ci-dessous, quel sera le résultat du parcours post-ordre de cet arbre?



e b j f c k g l h i d a

[Question 9] (2 points)

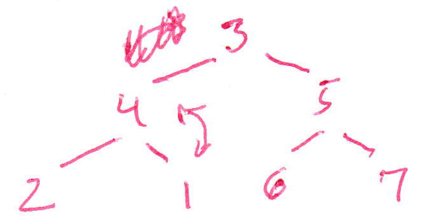
Combien de noeuds peut avoir un arbre binaire complet de hauteur h?

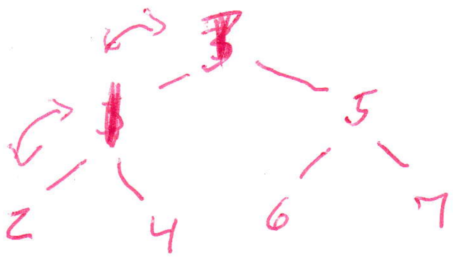
a. 2^h noeuds
b. 2^{h+1} noeuds
c. 2^{h-1} noeuds
d. entre 2^{h-1} et 2^h noeuds
e. entre 2^h et 2^{h+1} noeuds

entre 2^h et $2^{h+1} - 1$

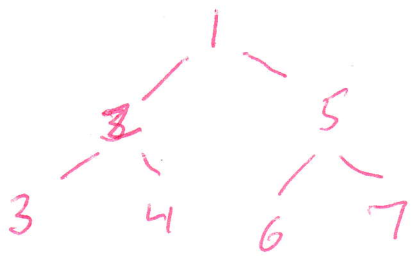
[Question 10] (6 points)

Soit la sequence suivante contenue dans un tableau `[3, 4, 5, 2, 1, 6, 7]`. Montrer étape par étape comment ce tableau peut être convertit en monceau-min en utilisant l'algorithme de construction <u>ascendante</u> vu en classe.





[3, 1, 5, 2, 4, 6, 7]



[1, 2, 5, 3, 4, 6, 7]

[Question 11] (3 points)

L'algorithme suivant utilise un monceau-min A représenté à l'aide d'un tableau. Quelle est la complexité Grand O de cet algorithme au pire cas en fonction de la taille du tableau (n)?
`Heapify (A)` est un algorithme permettant de convertir une séquence en un monceau-min de façon optimale.

```
sum = 0;
Heapify (A)
for (i=0; i<n-1; i++) do
     a = removeMin(A)
     sum = sum + a;
return sum
```

$O(n)$

$O(\log n)$ exécuté n fois

$O(n \log n)$

[Question 12] (4 points)

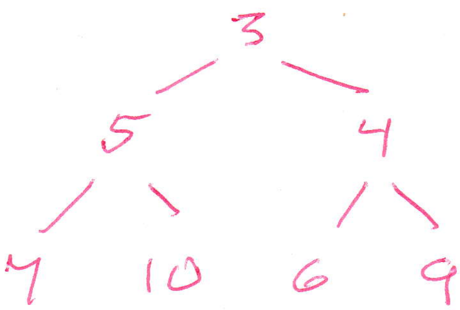
Voici un arbre binaire complet représenté à l'aide du tableau [3, 5, 4, 7, 10, 6, 9].

(a) Donner la valeur de l'enfant de droite du noeud de valeur 5?

10

(b) Donner la liste de tous les noeuds externes?

7, 10, 6, 9



[Question 13] (8 points)

(a) Combien y-a-t-il de noeuds externes dans un arbre binaire complet comptant 3567 noeuds?

arbre parfait de 2047 noeuds

1520 noeuds au dernier niveau + ~~1287~~ (1024 − (1520/2)) avant dernier niveau = ~~2807~~ 1764 noeuds externes

(b) Quelle est la hauteur d'un arbre binaire parfait comptant 4096 noeuds? 5

$2^{h+1} - 1 = 4095$ $h = 11$

(c) Quelle est la hauteur d'un monceau-min comptant 340 noeuds?

arbre parfait de 255 noeuds   8

(d) Combien de noeuds, <u>au maximum</u>, peut avoir un monceau-max de hauteur 8?

$2^{h+1} - 1 = 511$ noeuds

(e) Combien de nœuds internes y-a-t-il dans un monceau-max comptant 500 nœuds?

123+127 = 250

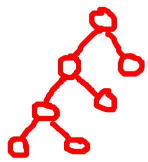
arbre parfait de 255 noeuds

245 noeuds au dernier niveau donc 123 des 128 noeuds de l'avant dernier niveau sont internes

(f) Combien y-a-t-il de nœuds externes, au minimum, dans un arbre binaire plein comptant 7 nœuds?

~~123 + 127 = 250~~

4 noeuds



(g) Quelle est la hauteur maximum d'un arbre binaire quelconque comptant 500 nœuds?

499

(h) Combien y-a-t-il de nœuds internes dans un arbre parfait de hauteur 4?

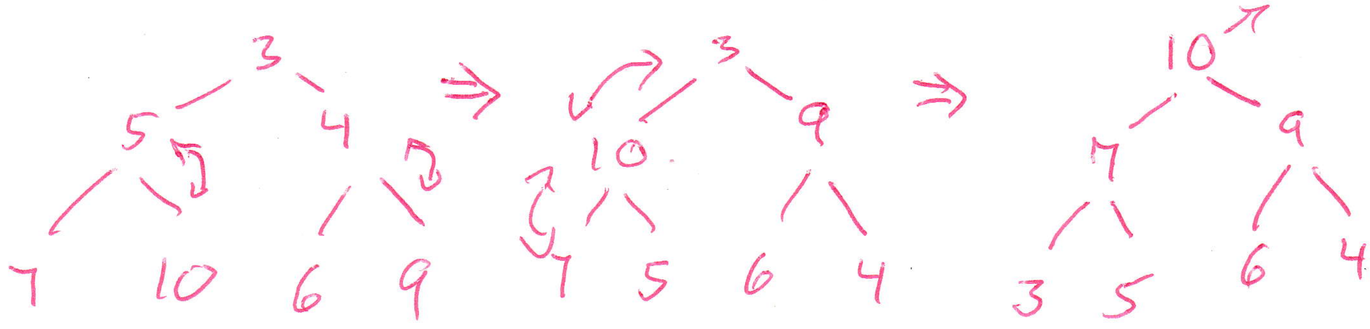
$(2^{h+1} - 1) - 2^h$

$31 - 16 = 15$

[Question 14] (8 points) il faut donc un monceau-max

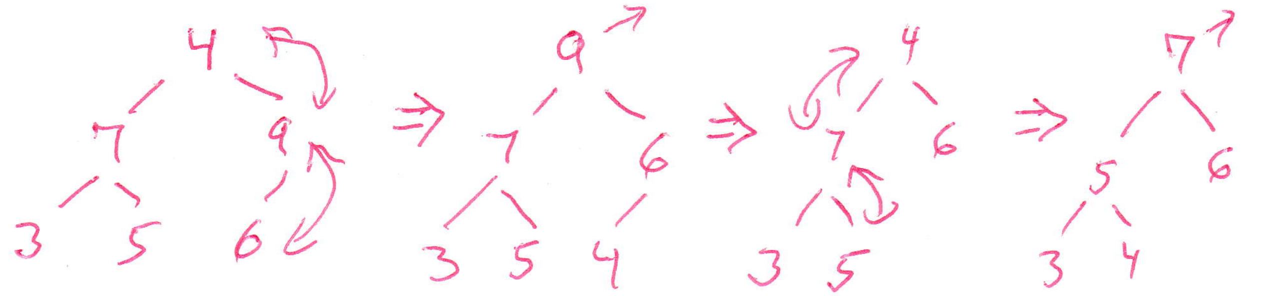
Soit la séquence suivante `[3, 5, 4, 7, 10, 6, 9]`. Montrer, étape par étape, l'application du tri par monceau sur place. Le tri doit se faire en ordre croissant.

1° Construction ascendante du monceau

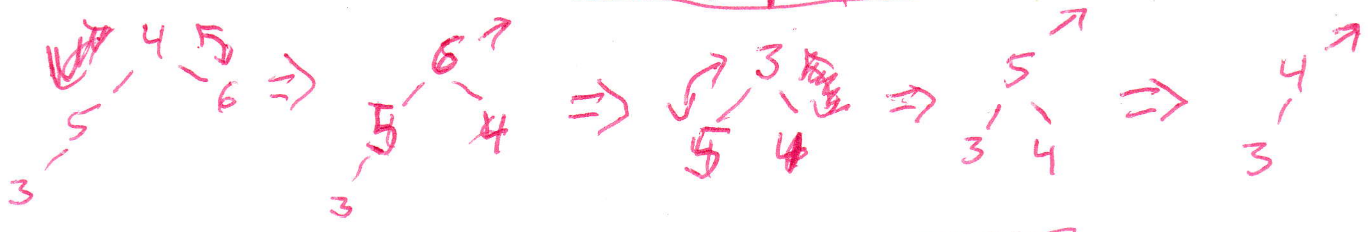


10	7	9	3	5	6	4

2° Remove max



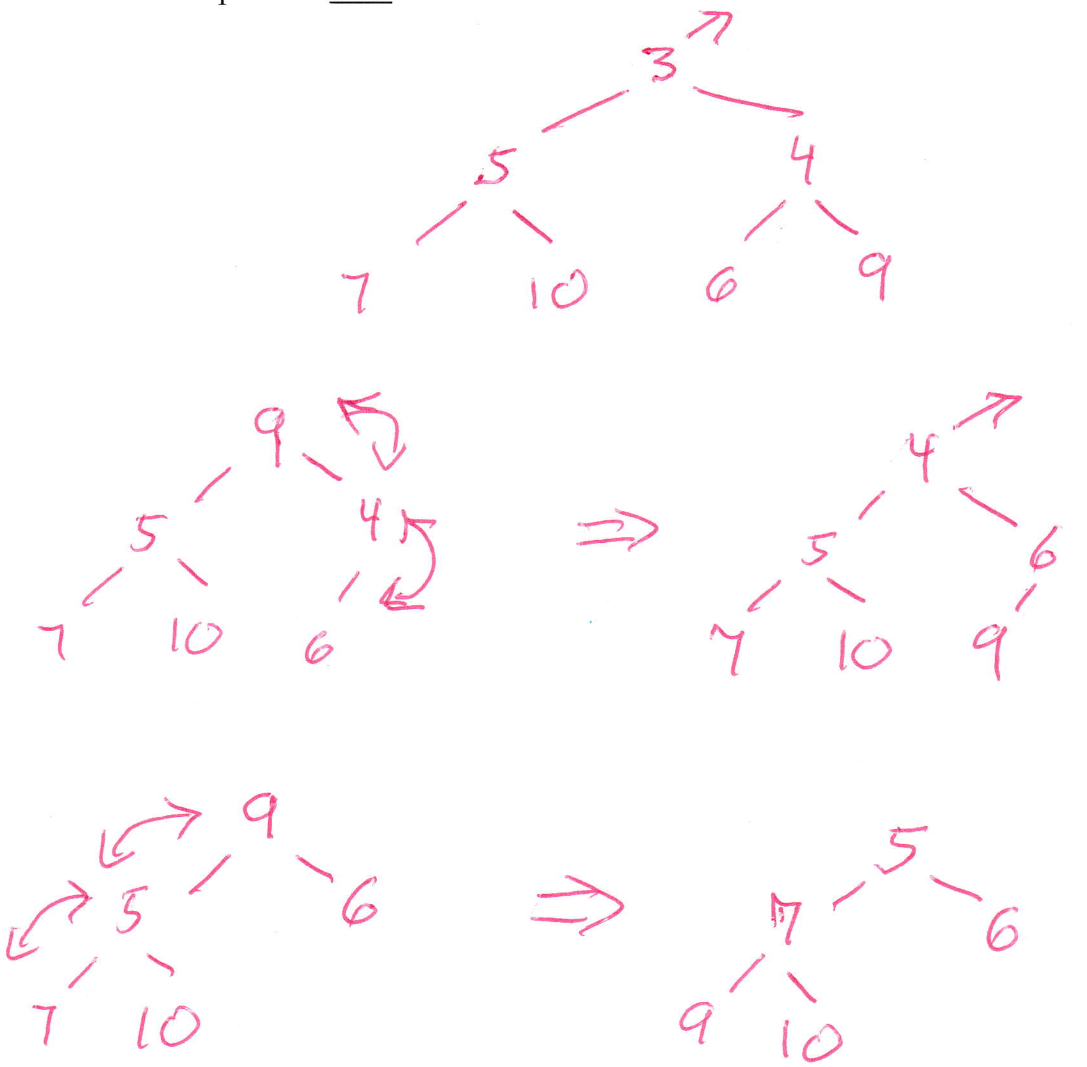
7	5	6	3	4	9	10



3	4	5	6	7	9	10

[Question 15] (6 points)

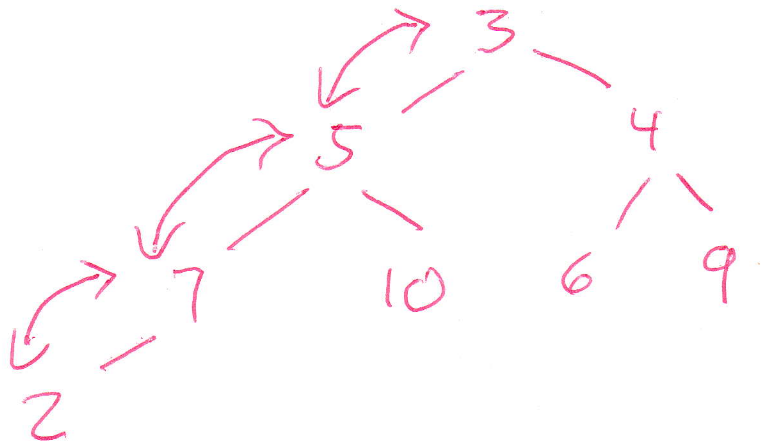
Soit l'arbre monceau-min représenté avec le tableau suivant `[3, 5, 4, 7, 10, 6, 9]`. Montrer, étape par étape ce qui se produit si 2 opérations de retrait de l'élément minimum (*removeMin*) sont effectuées. Vous pouvez illustrer ces étapes en dessinant l'arbre binaire correspondant. Donner l'état du tableau après ces deux retraits.

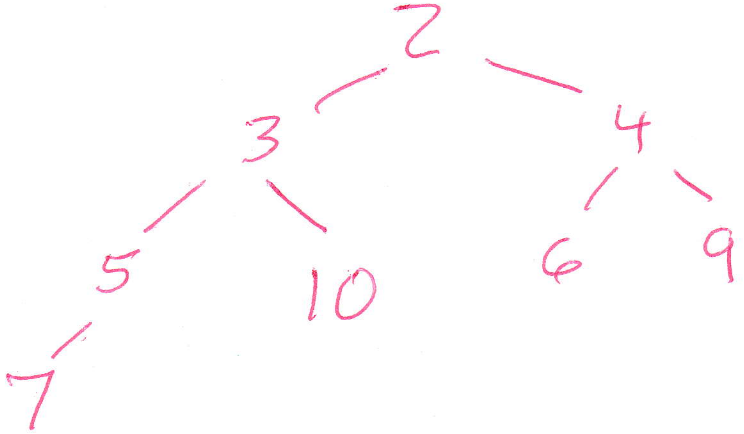


5	7	6	9	10

[Question 16] (6 points)

Soit l'arbre monceau-min représenté avec le tableau suivant `[3, 5, 4, 7, 10, 6, 9]`. Montrer, étape par étape ce qui se produit si le nombre 2 est inséré dans cet arbre. Vous pouvez illustrer ces étapes en dessinant l'arbre binaire correspondant. Donner l'état du tableau après cette insertion.



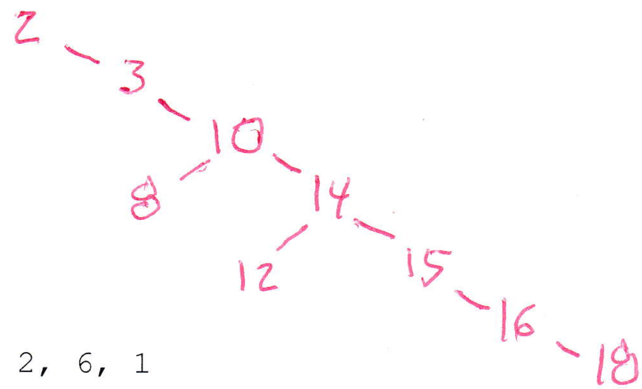


2	3	4	5	10	6	9	7

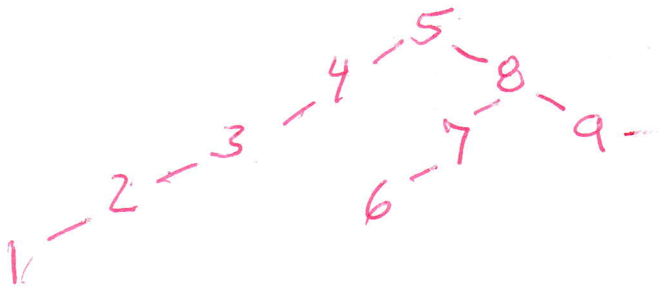
[Question 17] (8 points)

(1) Dessiner les arbres binaires de recherche obtenus si les séquences suivantes sont insérées dans l'arbre. Pour chaque séquence, l'insertion se fait initialement dans un arbre vide. [6]

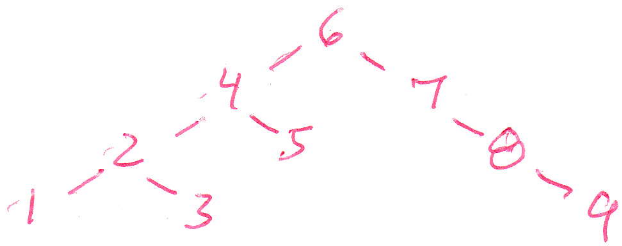
a. 2, 3, 10, 8, 14, 12, 15, 16, 18



b. 5, 8, 4, 9, 3, 7, 2, 6, 1



c. 6, 4, 7, 2, 5, 8, 3, 1, 9



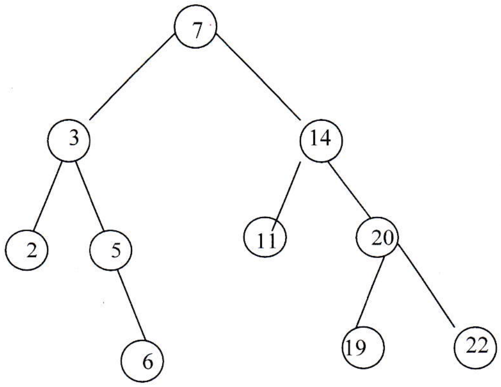
(2) Lequel de ces arbres donnera la recherchéela plus efficace en moyenne? [2]

c. nbre moyen d'éléments visités pour une recherche

1 + 2 + 2 + 3 + 3 + 3 + 4 + 4 + 4

[Question 18] (6 points)

Soit l'arbre binaire de recherche ci-dessous.



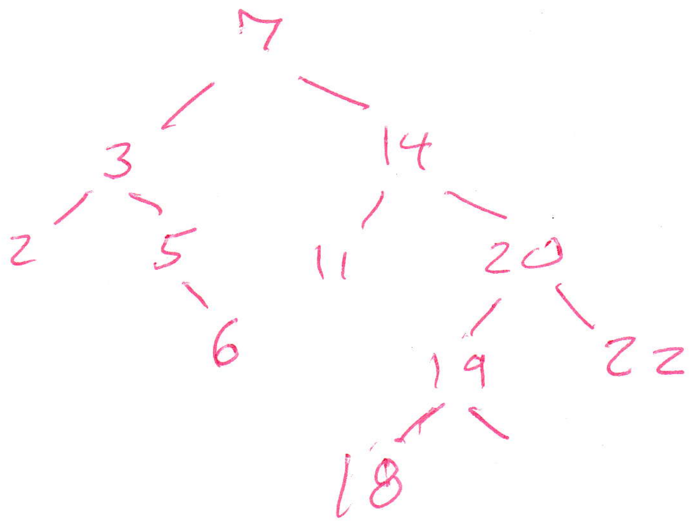
(a) Combien de comparaisons seront requise afin de trouver le nombre 21?

comparaisons avec 4 éléments

(b) Combien de comparaisons seront requise afin de trouver le nombre 4?

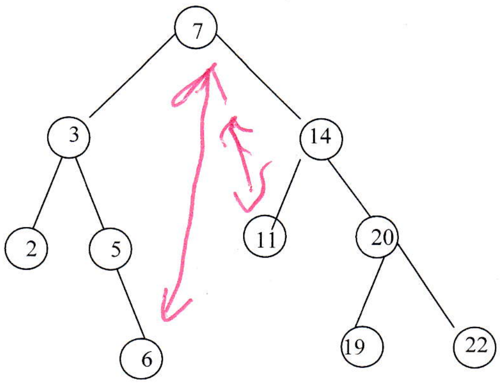
comparaisons avec 3 éléments

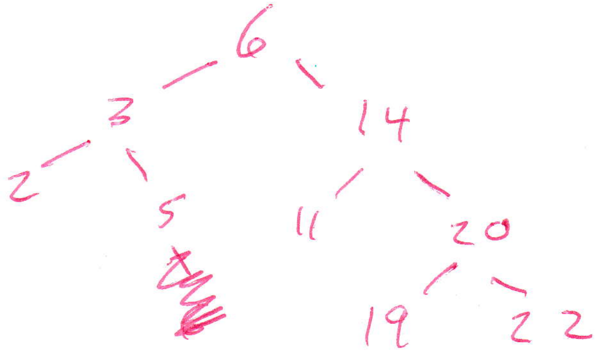
(c) Dessiner l'arbre résultant après l'insertion de la valeur 18.



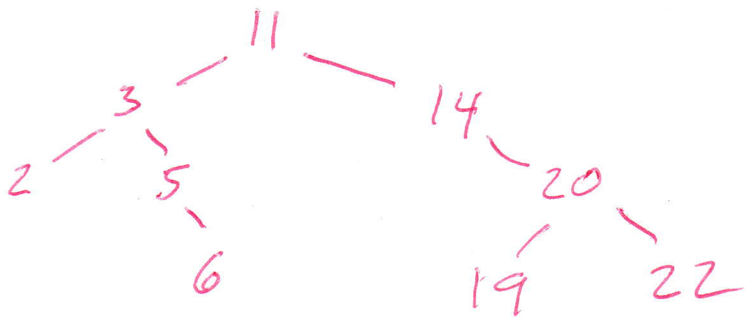
[Question 19] (5 points)

Retirer le noeud de valeur 7 dans l’arbre binaire de recherche ci-dessous. Bien expliquer les étapes suivies





OU



[Question 20] (2 points)

Lequel/Lesquels des algorithmes suivants permet de visiter les noeuds d'un arbre binaire de recherche en ordre croissant de clés.

(a)
```
Algorithm mySorting(T,v)
     visit(v)
if v is internal:
     mySorting (T,T.LeftChild(v))
     mySorting (T,T.RightChild(v))
```

préordre

(b)
```
Algorithm mySorting(T,v)
if v is internal:
     mySorting (T,T.LeftChild(v))
     mySorting (T,T.RightChild(v))
  visit(v)
```

postordre

(c)
```
Algorithm mySorting(T,v)
if v is internal:
     mySorting (T,T.LeftChild(v))
  visit(v)
if v is internal:
     mySorting (T,T.RightChild(v))
```

in ordre

(d)
```
Algorithm mySorting(T,v)
for (i = 0; i < n; i++) {
   vmin= removeMin(v);
       visit(vmin);
}
```

(e)
```
Algorithm mySorting(T,v)
for (i = 0; i < n; i++) {
   vmax= removeMax(v);
       visit(vmax);
}
```