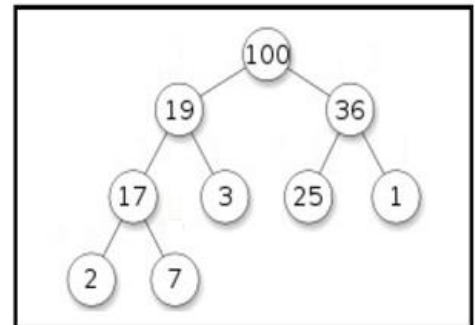


**TDP N° 2**  
**Tri par Fusion**  
**Tri par Tas**

**Partie 01 : TD**

**Exercice 0:**

1. Soit l'arbre presque-complet ci-contre, en justifiant votre réponse, afficher :
  - a. Le résultat du parcours Infixe (GRD)
  - b. Le résultat du parcours préfixe (RGD)
  - c. Le résultat du parcours postfixe (GDR)
  - d. Le résultat du parcours en largeur



**Exercice 1:** Construction de Tas Max

Construire un **Tas Max** en ajoutant respectivement les éléments suivants dans l'ordre d'apparition: **73 - 44 - 59 - 51 - 63 - 32 - 80 - 74 - 53 - 85**

**Indication :** à la 1<sup>ère</sup> itération, le tableau contient un seul élément, à la 2<sup>ème</sup> itération, le tableau contient 2 éléments, à la 3<sup>ème</sup> itération, le tableau contient 3 éléments ... etc

**N.B.:** à chaque itération, les éléments du tableau non remplis restent vides

**Tableau 1. Construction d'un Tas Max**

Iter 1										
Iter 2										
Iter 3										
Iter 4										
Iter 5										
Iter 6										
Iter 7										
Iter 8										
Iter 9										
Iter 10										

**Exercice 2** : Tri par Tas Max

**NB.** La transformation de l'arbre de tas max vers un tableau se fait en réalisant un parcours horizontal (par niveau).

Soit T, le Tas Max contenant les éléments contenus dans la ligne **Init** du tableau 1 ci-dessous.

A partir du Tas max T, réaliser un tri par tas max du tableau 1. Le tri est un tri dans l'ordre croissant.

Répondre en remplissant le tableau 1, en précisant pour chaque ligne le contenu du tableau après chaque itération de tri.

**Tableau 1. Tri croissant du Tas Max T**

Init	74	61	56	49	54	32	44	25	38	35
Iter 1										
Iter 2										
Iter 3										
Iter 4										
Iter 5										
Iter 6										
Iter 7										
Iter 8										
Iter 9										

**Partie 02 : TP**

**Exercice 3: Tri par Fusion**

Générer aléatoirement un tableau T de taille n puis le trier via l'algorithme de tri par fusion (vu au cours).

Faire le même traitement appliqué au TP 1 (changer la taille du tableau et ploter le temps d'exécution moyen pour en conclure la complexité de l'algorithme)

**Exercice 4: Tri par Tas**

1. Générer aléatoirement un tableau T de taille n
2. Transformer T en un tas max
3. Trier dans un ordre croissant, le tas résultant dans la phase 2.
4. Faire le même travail (phases 1, 2 et 3) en variant **le plus possible** la taille n et en **calculant la complexité** de la phase 2 et 3 de chaque valeur de la taille n.
5. Refaire les mêmes questions 1,2,3 et 4 pour faire un tri dans un ordre croissant dans un tableau en utilisant un **tas min.** (comparer les temps d'exécutions pour tas max et tas min)