

# RAPPORT DES PARCOURS DU TP2

Année universitaire: 2022\_2023

Encadreur: Mr.Zegour D.Eddine.

Binome: Sid Ahmed Sarah \_\_\_ Mendjel Chahrazed

Groupe: 08 , Section: C

**Explication:** nous avons lancé le programme C en créons les arbres avec des nombres de nœuds croissant de 10-10000, tout en notons le pourcentage de stabilité de l'arbre crée par les cinq parcours. (Les données sont afficher dans le fichier DATA\_OUT).

\_\_\_ Pourcentage de Stabilité? - Le pourcentage de stabilité est la somme des différences de nombres de nœuds dans un niveau par rapport au maximum de nœuds que ce même niveau peu contenir division par la hauteur max de l'arbre.

L'équation est la suivante:  $per = ((\sum_{k=2}^{niv} n/2^k)/h)*100$  . \*Stabilité d'un arbre\*

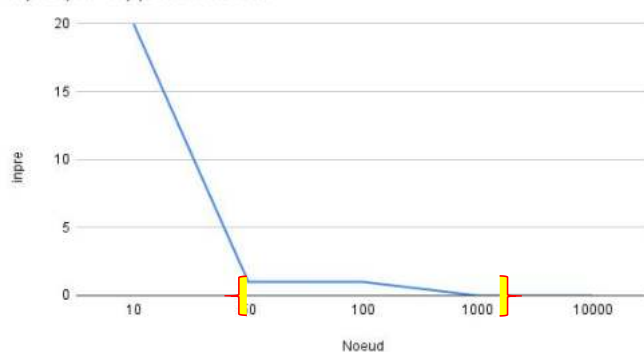
En transférant les données stocké dans le fichier vers excel, nous avons tracé les graphes correspondant à chaque parcours, comme montré ci-dessous:

## ANALYSE DES RESULTATS:

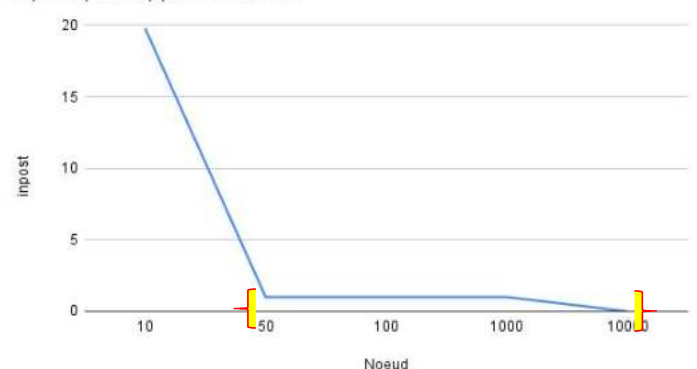
Comme nous pouvons le voir (et comme nous l'avons anticipé), les parcours Inordre-pre et Inordre-post donnent des resultats fort instable, avec des hauteurs presque toujours proche ou égales au nombres de nœuds, ce qui veut dire que l'arbre est plus ou moins une liste chaînée. (Pas du tout optimal). À peine nous augmentant le nombre de nœuds vers 50 est le pourcentage de stabilité prend des valeurs proche de 0.

D'autre caractéristique: Complexité du programme plus élevé (+++).

inpre par rapport à Noeud



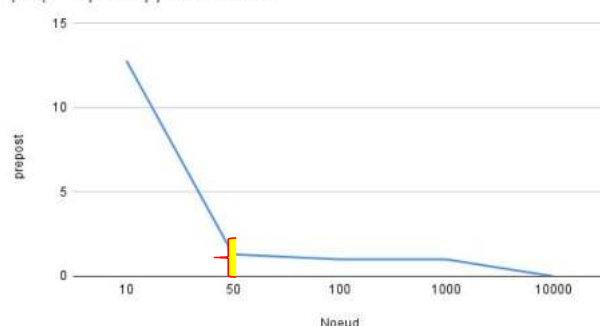
inpost par rapport à Noeud



-----

Pour le Preordre-post, il sa vert plus ou moins mieux que les précédant or bien loin d'être idéale pour la fusion de deux arbres. Car dépassant nœud=50, le pourcentage de stabilité de vient équivalent à 1-0. (Proche d'une liste linéaire chaînée de nœuds, non optimal aussi.).

prepost par rapport à Noeud

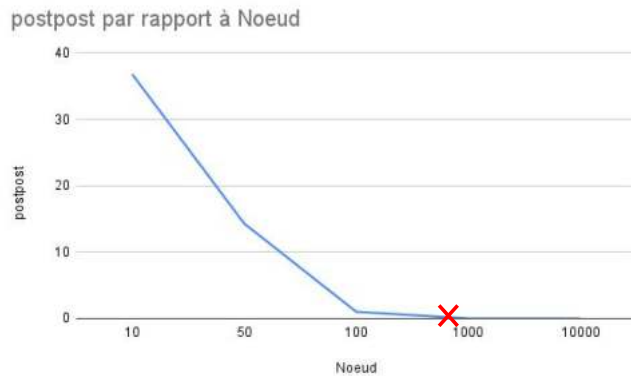


-Complexité et temps d'exécution réduit.

---

Pour le parcours Postpost, les hauteurs des arbres sont plus ou moins idéalisées en fonction du nombre de nœuds, or arrivant à un nombre de nœuds  $>100$  le pourcentage de stabilité commence à se réduire radicalement, jusqu'à arriver à une valeur équivalente à 0. Comme il apparaît sur la courbe ci-dessous. Par ailleurs, il est bien mieux que les trois autres parcours précédents.

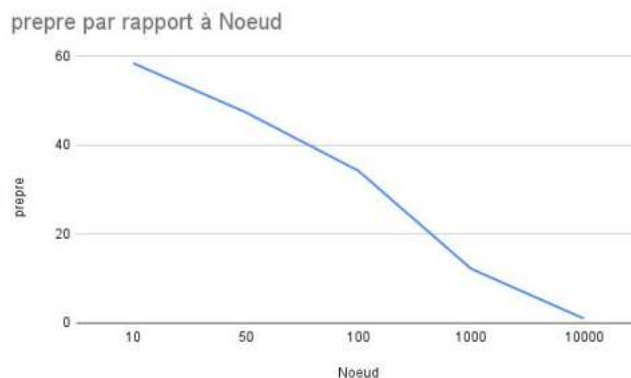
D'autres caractéristiques: -Temps et complexité d'exécution plus rapides et réduits.



---

Dernier parcours, Preordre-Preordre, les hauteurs des arbres sont bien plus réduites par rapport aux autres parcours (ex:  $N=100$ , Hauteur-max=34-38), le pourcentage de stabilité de l'arbre est très élevé d'où très optimale à la complétude de l'arbre (45% à 58%), bien que, l'accroissement du nombre de nœuds insérés dans un arbre associe aussi le décroissement du pourcentage de complétude de l'arbre mais cela pour des nuages de données volumineuses (presque  $N=100000$ ).

D'autres avantages: -Le temps d'exécution de ce parcours est plus réduit.



## **CONCLUSION:**

Classement	Complexité	Temps	Stabilité	Complétude	Hauteur	Results
Inordre-Pre	High	High	Low	Low	High	Bad
Inordre-Post	High	High	Low	Low	High	Bad
Pre-Post	Stable	Stable	Low	Low	Stable	Stable
Pre-Pre	Low	Low	High	High	Low	Ideal
Post-Post	Low	Low	Low	Stable	Stable	Stable

De ce fait, après analyse des cas et exécution répétitive du programme C sur plusieurs arbres différents, nous avons arrivé au classement des parcours comme ci-dessous.

Par ce tableau, nous concluons par notre recherche, que le parcours idéale pour la fusion de deux arbre en un arbre binaire de recherche incluons des doubles serait le parcours **Preordre-Preordre (Prepre).**