

# Compte rendu Projet AP3

## Compte rendu des expérimentations ABR

Le déséquilibre d'un ABR fait par la fonction `bst_rnd_create` :

La fonction `bst_rnd_create(x, tree)` crée un arbre de taille `x` et dont les éléments sont compris entre 1 et 100 (valeur arbitraire mise directement dans le code de la fonction). Pour calculer le déséquilibre d'un arbre nous avons fait les fonctions :

`dif(abr)` : qui prend un arbre et renvoie la différence entre la hauteur du fils droit et celle du fils gauche.

`bst_generator(n)` : qui prend un int, crée un arbre avec `bst_rnd_create(n)` et renvoie le déséquilibre de cet arbre

`average_cal(n,x)` : qui prend 2 int et en renvoie 3

Cette fonction crée un nombre `x` d'arbre et récupère la différence de hauteur de chaque arbre. Calcule la moyenne de cette différence et renvoie la moyenne, la différence minimum et la différence maximum.

Pour des arbres contenant 100 valeurs, avec 100 000 arbres nous arrivons à une moyenne de 4.8 de déséquilibre un minimum à 0 et un maximum à 21.

Le déséquilibre d'un ABR fait avec des liste ordonnées :

la fonction `crea_list_ordo(x,n,list)` crée une list ordonnée avec des valeur aléatoires un arbre est ensuite créer avec cette liste

Comme pour les abr fait par la fonction `bst_rnd_create` nous avons fait des fonction permettant de créer un grand nombre d'arbre et de renvoyer la moyenne du déséquilibre.

pour un arbre contenant 100 valeurs, avec 100 000 arbres nous arrivons à un déséquilibre moyen de 48,5 un minimum de 0 et un maximum de 98