

PRALG

Rapport TP1: Arbres

Question 2.1 La suite [12 8 4 9 23 17 15] correspond à un parcours en profondeur first de l'arbre.

Question 3.1 Les erreurs pour les fonctions de `IntTree` sont:

- `IntTree(int d)`
- `~IntTree()`
- `int getData() const`
- `void setData(int d)`
- `int nbSons() const`
- `void setSon(int pos, IntTree* newSon):`
 - Le *pos* est **out of range** sur la taille du vecteur.
- `IntTree* getSon(int pos):`
 - Le *pos* est **out of range** sur la taille du vecteur.
- `const IntTree* getSon(int pos) const:`
 - Le *pos* est **out of range** sur la taille du vecteur.
- `void addAsLastSon(IntTree* newSon):`
- `void removeLastSon()`
- `void display(std::string prefix, std::string indent)`

Question 3.2 Le signalement d'erreur par retour de valeur nécessite de tester la valeur d'un paramètre qui ne serait pas en cohérence avec l'opération à effectuer (par exemple un `int` négatif alors qu'on veut calculer sa racine carrée). On constate que les erreurs listées ci-dessus ne s'y prêtent pas.

Question 3.3 Le signalement d'erreur par exception se fait lorsqu'un paramètre, bien qu'*a priori* cohérent avec ce qui est demandé, n'est en fait pas conforme pour d'autres raisons. On constate que les erreurs ci-dessus peuvent toutes être traitées par l'exception.

Question 3.4 Le mode de signalement d'erreur doit être unique dans un programme. Avec ce que l'on vient de dire, il est évident que nous choisirons le mode de signalement d'erreurs par exception.