

Objectifs

À la fin de cette séance, vous devriez être capable de :

- écrire des fonctions de manipulation pour un type abstrait dynamique;
- jongler avec des ABRI ...

Exercice 1 :

On étudie dans ce TD une structure de données qui permet de gérer les réservations d'un équipement pendant certaines périodes de temps. Pour cela, on considère des arbres binaires dont les nœuds sont étiquetés par des intervalles (par exemple d'entiers). Un nœud contenant l'intervalle $[a..b]$ représente alors une réservation de l'instant a à l'instant b .

Pour faciliter l'utilisation de la structure à la manière des arbres binaires de recherche, nous définissons la relation suivante entre intervalles :

$$[a..b] < [c..d] \Leftrightarrow b \leq c$$

Pourquoi cette relation est-elle bien adaptée pour établir un système de réservations et pour construire des arbres binaires de recherche d'intervalles ?

1. Mise en route

- Dessinez un ABRI d'au moins 6 nœuds. Est-il possible que des intervalles se chevauchent ? Est-il possible que des intervalles se « touchent » ?
- Définissez le type abstrait arbre, et les primitives d'accès et de construction qui vous seront utiles.
- Écrivez une fonction qui calcule la somme des longueurs des intervalles d'un ABRI.
- À quelle condition est-il possible d'insérer un intervalle $[a..b]$ dans un ABRI ? Écrivez une fonction qui insère un intervalle dans un ABRI. Précisez la convention adoptée pour indiquer si l'insertion a été effectivement possible.

2. Parcours

- Écrivez une fonction qui vérifie si un arbre binaire d'intervalles est un ABRI.
- Écrivez un fonction qui calcule la longueur du plus grand "trou" entre deux intervalles (d'un ABRI).

3. Transformation

- Écrivez une fonction qui transforme un ABRI en "tassant" tous les intervalles à gauche : les intervalles seront alors tous contigus.

Membres du groupe et responsabilités

	Appliquer des méthodes algorithmiques connues (algorithme récursif, parcours d'arbre)	Gérer le temps
	Évaluer l'efficacité des algorithmes	Réguler les prises de parole
	Rechercher des cas problématiques Déterminer des invariants	S'assurer que tous comprennent et sont en accord avec le rendu
	Représenter schématiquement les structures de données et leur traitement	Rédiger le rendu

Écrivez une fonction qui vérifie si un arbre binaire d'intervalles est un ABRI.

Soignez votre rédaction.