



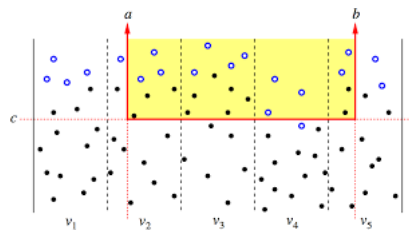
U-MONS

RAPPORT DE PROJET DE STRUCTURES DE DONNÉES 2

---

## Priority Search Tree and Windowing

---



**Directeurs :**  
G.Devillez et V.Bruyère

**Groupe :**  
M.Salemi et A.Lecocq

15 avril 2017

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Priority Search Tree</b>	<b>2</b>
2.1	Objectif de la structure . . . . .	2
2.2	Définition . . . . .	2
2.3	Construction . . . . .	2
2.3.1	Tri des données . . . . .	2
2.3.2	Création de l'arbre . . . . .	2
2.3.3	Création d'un nœud . . . . .	3
2.4	Windowing . . . . .	3
2.5	Mise en place du problème . . . . .	3
<b>3</b>	<b>Idées et Mise en pratique</b>	<b>3</b>
<b>4</b>	<b>Diagramme de Classes</b>	<b>3</b>
<b>5</b>	<b>Algorithmes et explications</b>	<b>3</b>
<b>6</b>	<b>Illustrations</b>	<b>3</b>
<b>7</b>	<b>Conclusion</b>	<b>3</b>

# 1 Introduction

Dans le cadre du cours de structures de données 2, nous avons été amenés à réaliser un projet en Java. Ce projet a pour objectif de créer et manipuler une structure de données non vue au cours. Cette nouvelle structure se base sur un arbre de recherche à priorité (Priority Search Tree en anglais ou PST). De la documentation nous a été fournie afin de nous familiariser avec cette dernière structure qui elle non plus n'a pas été vue au cours.

Un PST partage certaines caractéristiques avec les tas et les Arbres Binaires de Recherche ou ABR. Ces deux dernières structures de données ayant été étudiées en profondeur lors des séances de cours, nous les considérons ici comme connues.

## 2 Priority Search Tree

### 2.1 Objectif de la structure

Un PST est une structure de données de type arbre binaire (chaque nœud comporte au plus deux fils). Cette structure de données organise des points de l'espace défini par deux coordonnées  $X$  et  $Y$ . L'organisation des données permet d'effectuer efficacement la recherche des points présents dans une fenêtre de l'espace (sans avoir à parcourir l'ensemble des points).

### 2.2 Définition

Un PST est une structure de données mixte. Chaque nœud est constitué d'un point et d'un nombre appelé la médiane. Si l'on considère uniquement les coordonnées  $X$ , un PST est un tas avec le minimum à la racine. Si l'on considère uniquement les coordonnées  $Y$ , le PST est un ABR avec une petite particularité. Au lieu de trier les fils par rapport à la donnée du nœud courant, un PST trie les fils en fonction de la médiane du nœud courant. Ainsi, tout nœud  $n$  d'un PST respecte les contraintes suivantes :

- son fils gauche (s'il existe ainsi que ses descendants s'ils existent) aura sa coordonnée  $X$  plus grande que celle du nœud  $n$  et sa coordonnée  $Y$  plus petite que la médiane du nœud  $n$  ;
- son fils droit (s'il existe ainsi que ses descendants s'ils existent) aura sa coordonnée  $X$  plus grande que celle du nœud  $n$  et sa coordonnée  $Y$  plus grande que la médiane du nœud  $n$ .

### 2.3 Construction

#### 2.3.1 Tri des données

La construction d'un PST est plus simple si l'on construit ce dernier à partir d'une liste de points triés selon la coordonnée  $Y$ . La première étape est donc de trier les points.

#### 2.3.2 Création de l'arbre

Pour créer un arbre, créons le nœud racine à partir de la liste des points triés.

### 2.3.3 Création d'un nœud

La coordonnée X d'un nœud devant être plus petite ou égale à celle de ses fils, commençons par rechercher le point avec la plus petite coordonnée X. Attribuons ce point au nœud courant. Séparons le reste des points en deux. Attribuons comme médiane du nœud courant, la moyenne de la coordonnée Y du dernier point de la partie 1 et du premier point de la partie 2. Les points étant triés selon la coordonnée Y, la première partie est disposée d'une coordonnée Y plus petite ou égale à la médiane du nœud alors que la deuxième partie aura une coordonnée Y plus grande ou égale à la médiane du nœud. Dans le cas où il ne reste aucun point après le retrait du minimum en X, la valeur de la moyenne n'a pas d'importance. Dans le cas où il ne reste qu'un point, la coordonnée Y du point restant peut être attribuée à la médiane du nœud.

## 2.4 Windowing

Le windowing est une technique très répandue qui consiste à sélectionner certaines fenêtres parmi une énorme quantité de données. Un exemple pratique très répandu est l'affichage d'une carte sur un gps, le gps se déplaçant rapidement n'affichera pas toutes les routes qui se trouvent dans le monde (énorme quantité de données) mais seulement celles qui nous entourent au moment où nous roulons avec notre véhicule.

## 2.5 Mise en place du problème

Pour ce projet, il nous a été demandé de résoudre un problème de windowing utilisant la structure de données "Priority Search Tree". Il nous est donc fourni un fichier.txt qui contient un ensemble de segments que nous devons pouvoir afficher et sélectionner à travers la dite technique.

Il faut donc pouvoir gérer différents types de fenêtres lors du windowing :

1. À remplir
- 2.

## 3 Idées et Mise en pratique

## 4 Diagramme de Classes

## 5 Algorithmes et explications

## 6 Illustrations

## 7 Conclusion