PRACTICA 5 DE LABORATORIO

DOCENTE	CARRERA	CURSO
Vicente Machaca Arceda	Escuela Profesional de Ciencias de la Computación	Estructuras de Datos Avanzadas

PRÁCTICA N°	TEMA	DURACIÓN (HORAS)
05	K-d Tree	10

1. OBJETIVOS

- Implementar la estructura K-d Tree

2. TEMAS A TRATAR

- K-d Tree

3. MARCO TEÓRICO

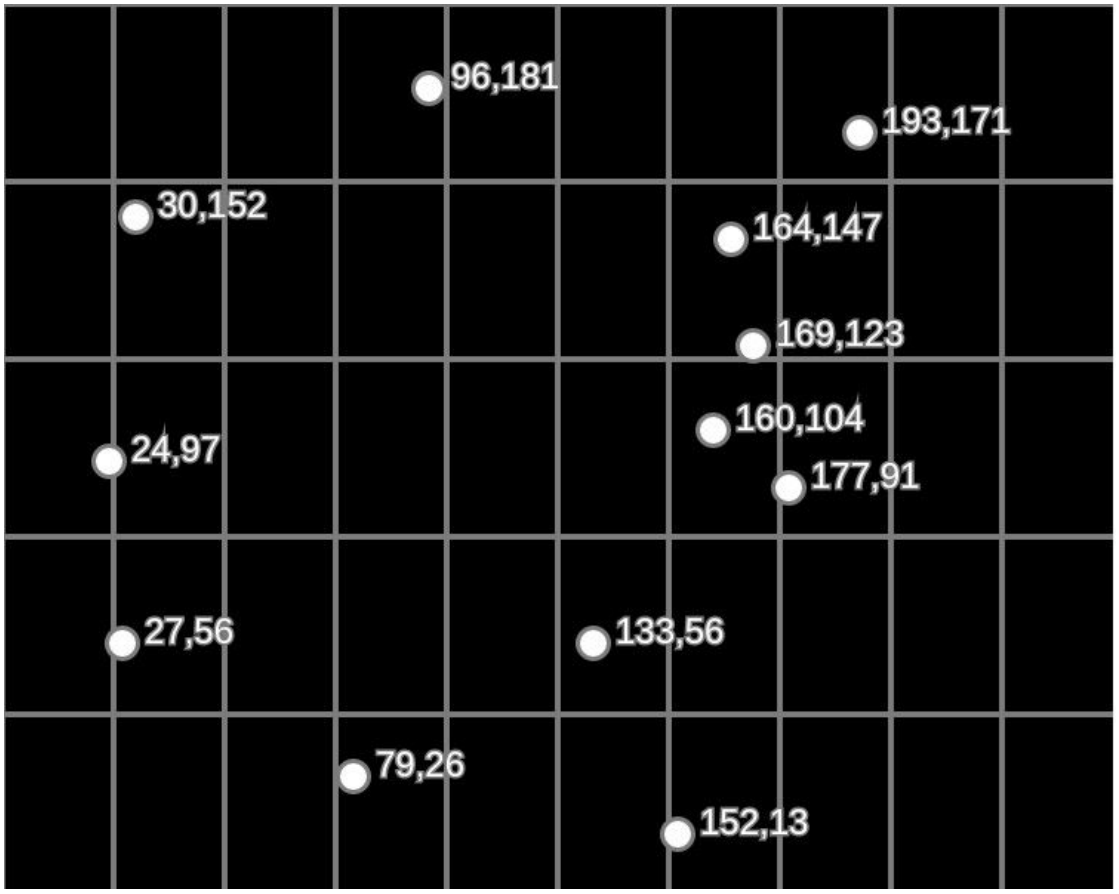
K-D TREE
(Extraido de: https://www.geeksforgeeks.org/k-dimensional-tree/)

A K-D Tree(also called as K-Dimensional Tree) is a binary search tree where data in each node is a K-Dimensional point in space. In short, it is a space partitioning(details below) data structure for organizing points in a K-Dimensional space.

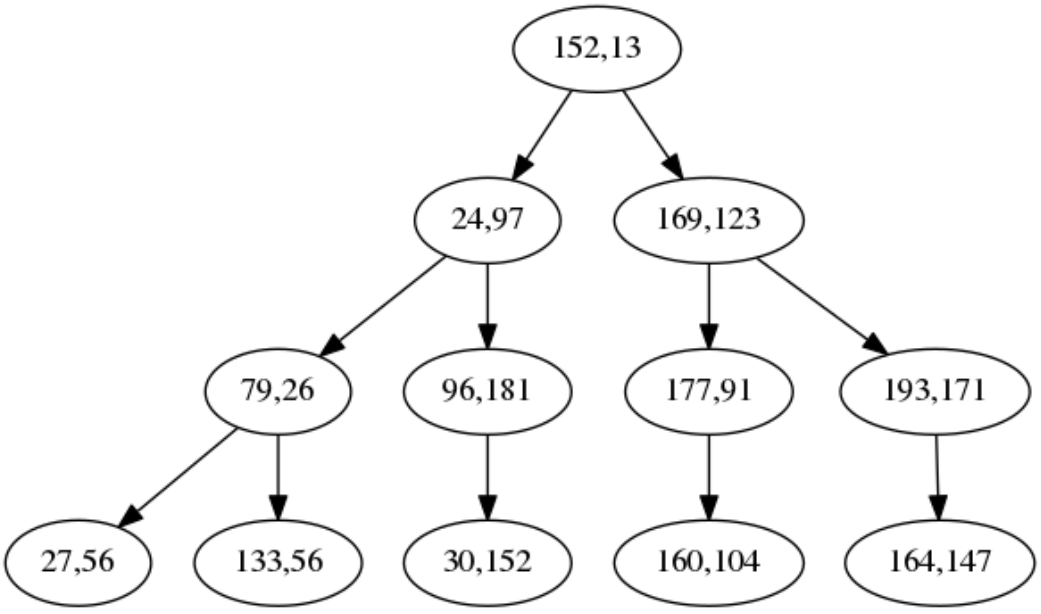
A non-leaf node in K-D tree divides the space into two parts, called as half-spaces.

Points to the left of this space are represented by the left subtree of that node and points to the right of the space are represented by the right subtree. We will soon be explaining the concept on how the space is divided and tree is formed.

For example for the next points:



We build the next tree:



4. EJERCICIOS

We will develop a K-d Tree in Javascript, follow the next steps:

1. **Copy the next code in main.html.**

```
<html>

<head>
<title>Kd tree</title>
<script src="p5.min.js"></script>
<script src="kdtree.js"></script>
<script src="sketch.js"></script>

</head>
<body>
</body>

</html>
```

2. **Create a kdtree.js file and copy:**

```
k = 2;

class Node{
   constructor(point, axis){
       this.point = point;
       this.left = null;
       this.right = null;
       this.axis = axis;
   }
}

function getHeight(node) {}
function generate_dot(node){}
function build_kdtree(points, depth = 0){}
```

3. **Complete the functions:**
   a. ***build_kdtree*:** Return a Node that represent the K-d Tree.
   b. ***getHeight:*** Return the tree's height
   c. ***generate_dot*:** Return a text in dot format , for example:

```
digraph G {
```

```
"152,13" -> "24,97";
"24,97" -> "79,26";
"79,26" -> "27,56";
"79,26" -> "133,56";
"24,97" -> "96,181";
"96,181" -> "30,152";
"152,13" -> "169,123";
"169,123" -> "177,91";
"177,91" -> "160,104";
"169,123" -> "193,171";
"193,171" -> "164,147";
}
```

4. **Create a file sketch.js and copy the next code:**

```
function setup(){
   var width = 250;
   var height = 200;
   createCanvas(width,height);

   background(0);
   for (var x = 0; x < width; x += width / 10) {
        for (var y = 0; y < height; y += height / 5) {
            stroke(125, 125, 125);
            strokeWeight(1);
            line(x, 0, x, height);
            line(0, y, width, y);
        }
    }

   var data = [];
   for(let i = 0; i < 12; i++){
       var x = Math.floor(Math.random() * height);
       var y = Math.floor(Math.random() * height);
       data.push([x, y]);

       fill(255, 255, 255);
       circle(x, height - y, 7); //200-y para q se dibuje apropiadamente
       textSize(8);
```

```
        text(x + ',' + y, x + 5, height - y);//200-y para q se dibuje
apropiadamente
    }

    var root = build_kdtree(data);
    console.log(root);
}
```

5. **Show your results and comment.**

REFERENCIAS

Maneewongvatana, S., & Mount, D. M. (1999, December). It's okay to be skinny, if your friends are fat. In *Center for Geometric Computing 4th Annual Workshop on Computational Geometry* (Vol. 2, pp. 1-8).