Assignment 13

1. Consider the multi-way search tree of Slide 5 in the Lecture 13 notes, is it a (2,4) tree? Justify your answer.

2. Consider the following sequence of keys:
   (5, 16, 22, 45, 2, 10, 18, 30, 50, 12, 1, 25, 7)
   Insert items with this set of keys, in the order given, into an empty (2,4)-tree.

R-3.10 A certain Professor Amongus claims that a (2,4) tree storing a set of items will always have the same structure, regardless of the order in which the items are inserted. Show that Professor Amongus is wrong.

Using the EulerTour template below, override the methods so the EulerTour finds the list of integers in the tree between integers **low** and **high**. Return an empty list if there are no elements between **high** and **low**.

```
class EulerTour {
        visitExternal(T, p, result) { }
        visitPreOrder(T, p, result) { }
        visitInOrder(T, p, result) { }
        visitPostOrder(T, p, result) { }
        eulerTour(T, p) {
                let result = new Array(3);
                if (this._tree.isExternal(p)) {
                        this. visitExternal(T, p, result);
                } else {
                        this.visitPreOrder(T, p, result);
                        result[0] = this.eulerTour(T, T.leftChild(p));
                        this.visitInOrder(T, p, result);
                        result[2] = eulerTour(T.rightChild(p));
                        this.visitPostOrder(T, p, result);
                }
                return result[1];
        }
}
class FindElements extends EulerTour {  // insert your code here
        constructor() {


        }
        findElements(T, low, high) {


        }
}
```