```javascript
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

// O(n^2) time | O(n^2) space - where n is the number of
// nodes in each array, respectively
function sameBsts(arrayOne, arrayTwo) {
  if (arrayOne.length !== arrayTwo.length) return false;

  if (arrayOne.length === 0 && arrayTwo.length === 0) return true;

  if (arrayOne[0] !== arrayTwo[0]) return false;

  const leftOne = getSmaller(arrayOne);
  const leftTwo = getSmaller(arrayTwo);
  const rightOne = getBiggerOrEqual(arrayOne);
  const rightTwo = getBiggerOrEqual(arrayTwo);

  return sameBsts(leftOne, leftTwo) && sameBsts(rightOne, rightTwo);
}

function getSmaller(array) {
  const smaller = [];
  for (let i = 1; i < array.length; i++) {
    if (array[i] < array[0]) smaller.push(array[i]);
  }
  return smaller;
}

function getBiggerOrEqual(array) {
  const biggerOrEqual = [];
  for (let i = 1; i < array.length; i++) {
    if (array[i] >= array[0]) biggerOrEqual.push(array[i]);
  }
  return biggerOrEqual;
}

exports.sameBsts = sameBsts;
```