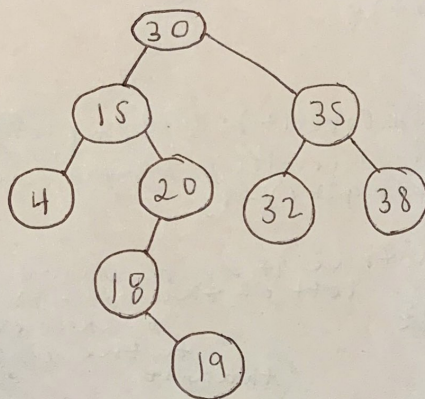


- ② Show tree for following data lines and describe what the resulting tree is



This tree is a binary search tree because it nodes added on the tree from the root are separated left and right based on if they're greater or lesser than the root value. This is not a full tree because not every node has zero or two children. Lastly this isn't a complete tree because not all the levels besides the last one is completely filled.

- ④ Write a function that tests whether a binary tree is a binary search tree

```
int Is-binary-search-tree (int root, int small, int large)
```

```
{
    if (root == null) {
        return "no tree";
    }
```

```
    int left_child = small; // first node on left side of given tree
    int right_child = large; // first node on right side of given tree
```

```
    while (tree.length() > 0) {
```



```
if (left_child > root) {
    return false;
}
```

```
else {
    if (right_child < root) {
        return false;
    }
```

```
    left_child = left_child.level(+1); // moving left child value to the next one
    right_child = right_child.level(+1); // same concept ↑
    tree.length() = tree.length() - 1;
}
```

~~while (tree.length() > 0) {~~ // return true if every value at the left of the root is less than it and every value to the right is greater than it

```
while (tree.length() > 0) {
    root = left_child;
```

~~if (left\_child > root) {~~

```
    left_child = left_child.level(+1);
```

```
    if (left_child > root) {
        return false;
    }
```

```
    if (right_child < root) {
        return false;
    }
```

```
    root = left_child;
```

```
    left_child = left_child.level(+1);
```

```
    tree.length() = tree.length() - 1;
```

```
}
```

```
while (tree.length() > 0) {
```

```
    root = right_child;
```

```
    right_child = right_child.level(+1);
```

```
    if (left_child > root) {
        return false;
    }
```

```
    if (right_child < root) {
        return false;
    }
```

```
    root = right_child;
```

```
    right_child = right_child.level(+1);
```

```
    tree.length() = tree.length() - 1;
```

```
    return true;
}
```

```
}
```

// checking if left side of tree is binary search

// checking if right side of tree is binary search