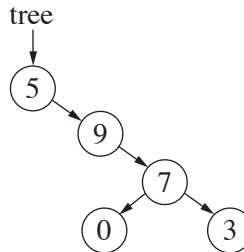


**לפותרים בשפת Java**

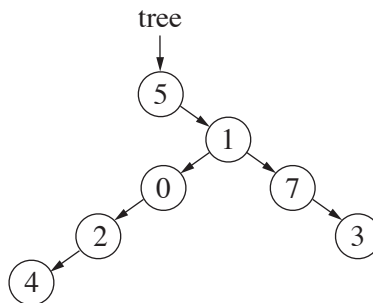
א. נתונה הפעולה foo שלפניכם:

```
public static int foo (BinNode<Integer> tree) {
    if (tree.getLeft() == null && tree.getRight() == null)
        return 0;
    if (tree.getLeft() == null)
        return foo (tree.getRight()) + 1;
    if (tree.getRight() == null)
        return foo (tree.getLeft()) - 1;
    return foo (tree.getLeft()) + foo (tree.getRight());
}
```

(1) בצעו מעקב אחרי הפעולה foo והעץ שלפניכם, וכתבו מה הפעולה מחזירה. יש להציג את המעקב.



(2) בצעו מעקב אחרי הפעולה foo והעץ שלפניכם, וכתבו מה הפעולה מחזירה. יש להציג את המעקב.



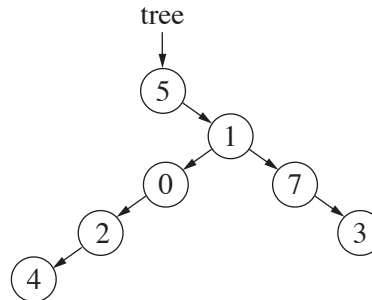
(3) בעבור עץ שיש לו 6 צמתים, מהו המספר הכי גבוה שהפעולה foo יכולה להחזיר? הציגו את העץ.

(4) בעבור עץ שיש לו 6 צמתים, מהו המספר הכי נמוך שהפעולה foo יכולה להחזיר? הציגו את העץ.

(5) כתבו בקצרה מה הפעולה foo מחזירה בעבור עץ כלשהו שאינו ריק.

```
public static boolean goo (BinNode<Integer> tree) {
    if (tree == null)
        return true;
    if (foo (tree) != 0)
        return false;
    return goo (tree.getLeft()) && goo (tree.getRight());
}
```

(1) בצעו מעקב אחרי הפעולה goo והעץ שלפניכם, וכתבו מה הפעולה מחזירה. יש להציג את המעקב. בסעיף זה אין צורך להציג מעקב אחרי הפעולה foo.



(2) הציגו עץ שיש לו מעל 3 צמתים שבעבורו הפעולה goo תחזיר תוצאה הפוכה מן התוצאה שהתקבלה בתת-סעיף ב(1).