

מבוא למדעי המחשב 202-1-101-1

סמסטר א' תשס"ד
פתרון מבחן מועד א'

1 תשובה

```
for (int i=0; i < arr.length & !ans ; i++) {  
    ans = true;  
    for (int j=0; j<arr.length & ans; j++)  
        if (i != j & arr[i] == arr[j] ) ans = false;  
}
```

2 תשובה

- 1) B.g
- 2) B.g
- 3) B.f
- 4) B.f
- 5) B.g
- 6) B.g

3 תשובה

```
static void oddPartitions(int n, int max, String suffix)  
{  
    if(n == 0)  
        System.out.println(suffix);  
    else  
        for (int i=1; i<=Math.min(n, max); i=i+2)  
            oddPartitions(n - i, i, i + " " + suffix);  
} // method oddPartitions
```

תשובה 4

נוסת ראשון:

```
public int min() {
    int r = get_root(), ml=r, mr=r;
    if(!left().isEmpty()) ml = left().min();
    if(!right().isEmpty()) mr = right().min();

    return Math.min(r,Math.min(ml,mr));
}

public boolean isBST() {
    boolean ans = true;
    if(isEmpty()) return true;
    if(!left().isEmpty()) ans = left().isBST() & get_root() > left().max();
    if(!right().isEmpty()) ans &= right().isBST() & get_root() < right().min();

    return ans;
}
```

נוסת שני:

```
public boolean isBST() {
    if(isEmpty()) return true;
    return isBST(_root);
}

private boolean isBST(BTN curr) {
    boolean ans = true;
    BTN ls = curr.left_node(), rs = curr.right_node();
    if(ls!=null)
        ans = isBST(ls) && curr.data() > max(ls);

    if(rs!=null)
        ans &= isBST(rs) && curr.data() < min(rs);

    return ans;
}

public int min() {
    return min(_root);
}

private int min(BTN curr) {
    int ans = curr.data(), ml = ans, mr=ans;
    BTN ls = curr.left_node(), rs = curr.right_node();
    if(ls!=null) ml = min(ls);
    if(rs!=null) mr = min(rs);
    return Math.min(Math.min(ans,ml),mr);
}
```