

שאלה 1, סעיף א (8 נק')

`public static int fibIter(int n){`

<code>int a0 = 0, a1 = 1;</code>
<code>int tmp;</code>
<code>while (n &gt; 0) {</code>
<code>tmp = a1;</code>
<code>a1 = a0 + a1;</code>
<code>a0 = tmp;</code>
<code>n = n - 1;</code>
<code>}</code>
<code>return a0;</code>

`}`

שאלה 1, סעיף ב (12 נק')

`public static int fibCount(int n, int m){`

<code>return fibIter(n-m+1);</code>

`}`

5	4	3	2	1
: סופי				

שאלה 2 סעיף א (5 נק')

**public static void hanoi(int n, char source, char target, char helper){**

if (n>0){
hanoi(n-1,source,helper,target);
System.out.println("move "+source+" to "+target);
hanoi(n-1,helper,target,source);
}

שאלה 2 סעיף ב (15 נק')

**public static void circularHanoi(int n, char source, char target, char helper){**

if (n>0){
if (target == next(source)){
hanoi(n-1,source,helper,target);
System.out.println("move "+source+" to "+target);
hanoi(n-1,helper,target,source);
} else{
hanoi(n,source,helper,target);
hanoi(n,helper,target,source);
}
}

}

שאלה 3, סעיף א (8 נק') יש להשלים (במידת הצורך) את שורת ההכרזה של הרכיב

`public abstract class Cell _____ {`

<code>protected String id;</code>
<code>public Cell(String id) {</code>
<code>    this.id = id;</code>
<code>}</code>
<code>public abstract double evaluate();</code>
<code>public String toString() {</code>
<code>    return id + " : ";</code>
<code>}</code>

`} // Cell`

שאלה 3, סעיף ב (5 נק') יש להשלים (במידת הצורך) את שורת ההכרזה של הרכיב

```
public class ValueCell extends Cell {
```

private double value;
public ValueCell(String id, double value) {
super(id);
this.value = value;
}
public double evaluate() {
return value;
}
public String toString() {
return super.toString() + value;
}
public void setValue(double value) {
this.value = value;
}

```
} // ValueCell
```

שאלה 3, סעיף ג (12 נק') יש להשלים (במידת הצורך) את שורת ההכרזה של הרכיבים

```
public abstract class FunctionCell extends Cell {
```

protected Cell[] cells;
public FunctionCell(String id, Cell[] cells) {
super(id);
this.cells = cells;
}
public String toString() {
String ans = super.toString() + getFuncName();
ans = ans + "(" + cells[0].id;
for (int i = 1; i < cells.length; i = i + 1)
ans = ans + " " + cells[i].id;
ans = ans + ")";
return ans;
}
public abstract String getFuncName();

```
}// FunctionCell
```

המשך בעמוד הבא

שאלה 3 סעיף ג - המשך

```
public class SumCell extends FunctionCell {
```

private static final String FUNC_NAME = "SUM";
public SumCell(String id, Cell[] cells) {
super(id, cells);
}
public double evaluate() {
double sum = 0;
for (int i = 0; i < cells.length; i = i + 1)
sum = sum + cells[i].evaluate();
return sum;
}
public String getFuncName() {
return FUNC_NAME;
}

```
} // SumCell
```

שאלה 4 (15 נק')

סעיף א. (7 נקודות)

```
public static boolean equalQueues(Queue q1, Queue q2) {
    Object marker = new Object();
    boolean res = true;
    if (q1 != q2) {
```

Object o1 = null, o2 = null;
q1.enqueue(marker);
q2.enqueue(marker);
while((q1 != null)    (q2 != null)) {
If (q1 != null) o1 = q1.dequeue();
If (q2 != null) o2 = q2.dequeue();
res = res && o1.equals(o2);
if (o1 != marker) q1.enqueue(o1);
else q1 = null;
if (o2 != marker) q2.enqueue(o2);
else q2 = null;
}

המשך בעמוד הבא

\

שאלה 4, סעיף ב. (8 נקודות) השלימו

```
public static void reverse(Queue q) {  
    Object o;
```

if (! q.isEmpty()) {
o = q.dequeue();
reverse(q);
q.enqueue(o);
}

```
} //reverse
```



שאלה 5, סעיף א (5 נק') :

```
public Iterator iterator() {
```

```
    return new PreOrderIterator(this);
```

```
}
```

שאלה 5, סעיף ב (8 נק') :

```
public class PreOrderIterator implements Iterator {
```

```
    private Stack s = new SomeStack();
```

```
    private Queue q = new SomeQueue();
```

השלימו בנאי\ם, שדות נוספים ושיטות עזר אם אתם נזקקים להם

```
public PreOrderIterator(BinaryNode t) {
```

```
    s.push(t);
```

```
}
```

המשך בעמוד הבא

שאלה 5 סעיף ב - המשך

```
public boolean hasNext() {
```

```
    return !s.isEmpty();
```

```
}// hasNext
```

```
public Object next() {
```

```
    if (! hasNext()) throw new NoSuchElementException();
```

```
    BinaryNode current = (BinaryNode)s.pop();
```

```
    if (current.right != null)
```

```
        s.push(current.right);
```

```
    if (current.left != null)
```

```
        s.push(current.left);
```

```
    return current.data;
```

```
}// next
```

```
}// PreOrderIterator
```

שאלה 5, סעיף ג (7 נק')

```
public void printPreOrder() {
```

Iterator iter = iterator();
while(iter.hasNext())
System.out.print(iter.next()+" ");
System.out.println();

```
}// printPreOrder
```

```
public void printFilteredPreOrder(Filter filter) {
```

Iterator iter = iterator();
while(iter.hasNext()) {
Object nextVal = iter.next();
if(filter.accept(nextVal))
System.out.print(nextVal+" ");
}
System.out.println();

```
}// printFilteredPreOrder
```

בהצלחה !