

## שאלה 1 (20 נק')

### סעיף א' (5 נקודות)

השיטה הבאה מתווספת למחלקה `BinaryNode`:

```
public void swapChildren(){
    BinaryNode temp = left;
    left = right;
    right = temp;
}
```

### סעיף ב' (10 נקודות)

השיטה הבאה מתווספת למחלקה `BinaryNode`:

```
public void mirrorSubTree(){
    swapChildren();
    if (left != null)
        left.mirrorSubTree();
    if (right != null)
        right.mirrorSubTree();
}
```

### סעיף ג' (5 נקודות)

השיטה הבאה מתווספת למחלקה `BinaryTree`:

```
public void mirrorTree(){
    if (root != null)
        root.mirrorSubTree();
}
```

## שאלה 2 (20 נק')

סעיף א (5 נקודות)

השיטה בודקת האם שני התורים זהים (אורך זהה ואותם האיברים באותו הסדר). במהלך הפעולה מוצאים איברים מהתורים.

סעיף ב (10 נקודות)

השיטה הופכת את סדר האיברים בתור

סעיף ג (5 נקודות)

בתור בו ישנם פחות מ-5 איברים יש לכתוב רק את האיברים הרצויים. יש לסמן X על המרווחים הנוותרים.

q1 = [1, 2, 3, 4, 5]

q2 = [5, 4, 3, 2, 1]

## שאלה 3 (20 נק')

סעיף א (10 נקודות)

```
public abstract class DerivedFunction implements Function {  
    public double square(double x) {  
        double ans = valueAt(x);  
        return ans * ans;  
    }  
    public abstract Function derived();  
}
```

```
public class LinearFunction extends DerivedFunction {
    protected double a,b;

    public LinearFunction(double a, double b) {
        this.a = a;
        this.b = b;
    }
    public Function derived() {
        return new ConstantFunction(a);
    }
    public double valueAt(double x) {
        return (a * x) + b;
    }

    public String toString() {
        String ans = "y = ";

        if (a == 0)
            ans = ans + b;
        else {
            ans = ans + a + "x";
            if (b < 0) ans = ans + b;
            else if (b > 0) ans = ans + " + " + b;
        }
        return ans;
    }
}
```

```
public class ConstantFunction extends LinearFunction {  
    public ConstantFunction(double c) {  
        super(0,c);  
    }  
}
```

שאלה 4 (20 נק')

סעיף א (5 נקודות)

```
public int size() {  
    int ans = 0;  
    Link currentLink = first;  
    while (currentLink != null) {  
        ans = ans + 1;  
        currentLink = currentLink.getNext();  
    }  
    return ans;  
}
```

```
public void moveLastToHead(){
    if (first != null && first.getNext() != null) {
        Link currentLink = first;
        while (currentLink.getNext().getNext() != null)
            currentLink = currentLink.getNext();
        currentLink.getNext().setNext(first);
        first = currentLink.getNext();
        currentLink.setNext(null);
    }
}
```

```
public void rotateRight (int k){
    int n = size();
    if(n > 1){ // the size of the linked list is more than 1
        Link currentLink = first;
        for (int i = 0; i < (n - (k + 1));i++) //move to the link before "new first"
            currentLink = currentLink.getNext();

        Link newFirst = currentLink.getNext();
        while (newFirst.getNext() != null)
            newFirst = newFirst.getNext();
        newFirst.setNext(first);
        first = currentLink.getNext();//update "first" element
        currentLink.setNext(null);
    }
}
```

## שאלה 5 (20 נקודות)

```
public static void subsetsOfSize(int n, int k, String s) {  
    if (k <= n) {  
        if (k == 0) System.out.println(s);  
        else {  
            subsetsOfSize(n-1, k, s);  
            subsetsOfSize(n-1, k-1, n+s);  
        }  
    }  
}
```

## שאלה 6 (10 נק')

### סעיף א (3 נקודות)

יש להקיף בעיגול את בחירתכם. נא לבחור או ב-"מחלקה חדשה" או לציין את שם המחלקה הקיימת.

א. מחלקה חדשה

ב. מחלקה קיימת: **Population**

### סעיף ב (3 נקודות)

יש להקיף בעיגול את בחירתכם. נא לבחור או ב-"שיטה חדשה" או לציין את שם השיטה הקיימת.

א. שיטה חדשה

ב. שיטה קיימת: **nextGeneration**

### סעיף ג (4 נקודות)

העברת individuals[0] למערך אוכלוסיית הדור הבא.

# בהצלחה!