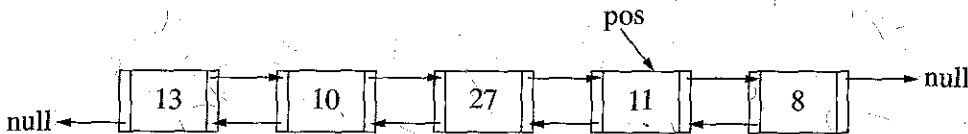


**שים לב:** לשאלה זו שני נוסחים:  
 נוסח אחד ב-Java בעמוד 8-9, ונוסח אחר ב-C# בעמ' 10-11.  
 עבוד על פי השפה שלמדת.

### לפותרים ב-Java

נגדיר **רשימה דו-כיוונית** כאוסף סדר של חוליות מטיפוס `BinNode<Integer>` המקושרות כך:  
 לכל זוג חוליות `p1, p2`, ברשימה, אם מתקיים `p1.getRight() == p2`, אז מתקיים גם  
`p2.getLeft() == p1`. ברשימה דו-כיוונית יש לפחות שתי חוליות.  
 כלומר: כל חוליה ברשימה – חוץ מהחוליה שבקצה הימני של הרשימה והחוליה שבקצה השמאלי  
 של הרשימה – מצביעה על החוליה שלפניה ועל החוליה שאחריה.  
 לפיכך דוגמה ל**רשימה דו-כיוונית** ומשתנה `pos` מטיפוס `BinNode<Integer>` המצביע על  
 חוליה כלשהי ברשימה דו-כיוונית.



הפעולה `firstLeft` מקבלת מצביע `pos` שונה מ-`null` מטיפוס `BinNode<Integer>` המצביע  
 על חוליה כלשהי ברשימה דו-כיוונית ומחזירה את החוליה השמאלית ביותר ברשימה.  
 הפעולה `firstRight` מקבלת מצביע `pos` שונה מ-`null` מטיפוס `BinNode<Integer>` המצביע  
 על חוליה כלשהי ברשימה דו-כיוונית ומחזירה את החוליה הימנית ביותר ברשימה.  
 א. לפיכך שלד של הפעולה `firstLeft`.  
 העתק אותו למחברתך והשלם אותו, כך שהפעולה תבצע את הנדרש.

```
public static BinNode<Integer> firstLeft(BinNode<Integer> pos)
```

```
{
    while (_____)
        pos = _____;
    return _____;
}
```

/המשך בעמוד 9/

ב. לפניך הפעולה `what(BinNode<Integer> pos)` המקבלת מצביע לחוליית כלשהי.

ברשימה דו-כיוונית ומחזירה ערך בוליאני. הרשימה הדו-כיוונית מכילה לפחות 3 חוליות.

(1) עקוב אחר ביצוע הפעולה בעבור המשתנה `pos` והרשימה שבדוגמה המוצגת

בתחילת השאלה.

במעקב הראה את הרשימה הדו-כיוונית ואת ערכי המשתנים `pos, left, right, sum`.

```
public static boolean what(BinNode<Integer> pos)
```

```
{
```

```
    BinNode<Integer> left = firstLeft(pos);
```

```
    BinNode<Integer> right = firstRight(pos);
```

```
    int sum = left.getValue() + right.getValue();
```

```
    left = left.getRight();
```

```
    right = right.getLeft();
```

```
    while ((left != right) && (left.getRight() != right) &&
```

```
           (left.getValue() + right.getValue() == sum))
```

```
    {
```

```
        left = left.getRight();
```

```
        right = right.getLeft();
```

```
    }
```

```
    if (left == right)
```

```
        return right.getValue() == sum;
```

```
    if (left.getRight() == right)
```

```
        return left.getValue() + right.getValue() == sum;
```

```
    return false;
```

```
}
```

(2) קבע אם אפשר או אי אפשר להחליף את 3 השורות האחרונות שבפעולה — השורות

המוקפות במסגרת — בהוראה:

```
return left.getValue() + right.getValue() == sum;
```

נמק את קביעתך.

/המשך בעמוד 10/