



מס' שאלון - 455

5

באפריל 2017

סמסטר 2017א

מס' מועד 90

20441 / 4

שאלון בחינת גמר

20441 - מבוא למדעי המחשב ושפת Java

משך בחינה: 3 שעות

בשאלון זה 13 עמודים

מבנה הבחינה:

קראו בעיון את ההנחיות שלהלן:

- \* בבחינה יש שש שאלות.
- \* כל התכניות צריכות להיות מתועדות היטב.
- יש לכתוב תחילה בקצרה את האלגוריתם וכל הסבר נוסף הדרוש להבנת התכנית.
- יש לבחור בשמות משמעותיים למשתנים, לפונקציות ולקבועים שבתכנית.
- תכנית שלא תתועד כנדרש לעיל תקבל לכל היותר 85 % מהניקוד.
- \* יש להקפיד לכתוב את התכניות בצורה מבנית ויעילה.
- תכנית לא יעילה לא תקבל את מלוא הנקודות.
- \* אם ברצונכם להשתמש בתשובתכם בשיטה או במחלקה הכתובה בחוברת השקפים, אין צורך שתעתיקו את השיטה או את המחלקה למחברת הבחינה. מספיק להפנות למקום הנכון, ובלבד שההפניה תהיה מדויקת (פרמטרים, מיקום וכו').
- \* אין להשתמש במחלקות קיימות ב-Java, חוץ מאלו המפורטות בשאלות הבחינה.
- \* יש לשמור על סדר; תכנית הכתובה בצורה בלתי מסודרת עלולה לגרוע מהציון.
- \* בכתיבת התכניות יש להשתמש אך ורק במרכיבי השפה שנלמדו בקורס זה
- אין להשתמש במשתנים גלובליים!
- \* את התשובות לשאלות 3 - 6 יש לכתוב על גבי השאלון. לא נבדוק תשובות שייכתבו במקום אחר!
- \* אפשר לתעד בעברית. אין צורך בתיעוד API.

## חומר עזר:

ספר הלימוד: java software solutions מאת: lewis/loftus  
חוברת השקפים של הקורס של ד"ר אמיר גורן ותמר וילנר.  
יחידות 1-6, 7-12. מותרות הערות בכתב יד, ע"ג הספרים.  
אין להכניס חומר מודפס או כל חומר אחר מכל סוג שהוא.

בהצלחה !!!

## החזירו

למשגיח את השאלון

וכל עזר אחר שקיבלתם בתוך מחברת התשובות





## חלק א – עליכם לענות על כל השאלות בחלק זה במחברת הבחינה

### שאלה 1 - 25 נקודות

אנו מגדירים פריט (Item) כעצם (אובייקט) שיש לו שתי תכונות: משקל (`_weight`) וערך (`_value`).

להלן ההגדרה של המחלקה Item:

```
public class Item {
    private int _weight;
    private int _value;

    public Item(int w, int v) {
        משקל _weight = w;
        ערך _value = v;
    }

    public int getWeight() { return _weight; }
    public int getValue() { return _value; }
}
```

נתון מערך חד-ממדי items מלא באובייקטים מהמחלקה Item, לכל אובייקט במערך יש משקל וערך.

כמו כן, נתון **תרמיל גב** (knapsack) שהקיבולת (capacity) שלו היא  $w$ , כלומר, אי אפשר להכניס לתרמיל פריטים במשקל כולל שהוא גדול מ- $w$ .

עלינו להחליט אילו פריטים כדאי להכניס לתרמיל הגב, כך שהערך של הפריטים האלו יהיה מקסימלי (וכמובן לא להכניס יותר מהקיבולת של התרמיל).

**לדוגמא**, אם המערך מלא בפריטים הבאים:

0	1	2	3
<code>_weight = 10</code>	<code>_weight = 20</code>	<code>_weight = 30</code>	<code>_weight = 15</code>
<code>_value = 60</code>	<code>_value = 100</code>	<code>_value = 120</code>	<code>_value = 200</code>

- אם קיבולת התרמיל היא  $w = 30$ , כדאי לקחת את הפריטים 0 ו-3 כדי שהערך יהיה 260. שימו לב שלמרות שיש שני פריטים (0 ו-1) ששוקלים ביחד 30, בעוד שהפריטים 0 ו-3 שוקלים רק 25, בכל זאת כדאי לקחת את הפריטים 0 ו-3 כי ערכם הכולל הוא 260 בעוד ששני הפריטים שבמקומות 0 ו-1 ערכם הוא רק 160.
- אם קיבולת התרמיל היא  $w = 50$ , אז כדאי להכניס את הפריטים שבתאים 0, 1 ו-3, וכך הערך המקסימלי יהיה 360.



עליכם לכתוב שיטה **רקורסיבית** המקבלת מערך של אובייקטים מהמחלקה Item ומשקל w (מספר שלם), ומחזירה מה הערך המקסימלי שאפשר להשיג מהאובייקטים שבמערך, כך שמשקלם הכולל לא יעלה על w. שימו לב שאין צורך לכתוב מהם הפריטים שנלקחים אלא רק את ערכם הכולל.

**חתימת השיטה היא:**

```
public static int knapSack(Item[] items, int w)
```

אפשר להניח שהמערך מלא בנתונים חוקיים. אין צורך לבדוק זאת.

**השיטה שתכתבו צריכה להיות רקורסיבית ללא שימוש בלולאות כלל. כך גם כל שיטות העזר שתכתבו (אם תכתבו) לא יכולות להכיל לולאות.**

**אין צורך לדאוג ליעילות השיטה!**

**אל תשכחו לתעד את מה שכתבתם!**



## שאלה 2 - 25 נקודות

כתבו שיטה יעילה המקבלת כפרמטר מערך חד-ממדי arr המלא במספרים שלמים חיוביים הממוינים בסדר לא יורד. השיטה מחזירה מהו המספר החיובי הקטן ביותר שלא יכול להיות סכום של קבוצת מספרים מהמערך.

חתימת השיטה היא:

```
public static int findSmallest (int [] arr)
```

דוגמאות:

• עבור המערך arr הבא:

0	1	2	3	4	5
1	2	6	10	11	15

מהו המספר הקטן ביותר שלא יכול להיות סכום של חלק מאיברי המערך?  
1 יכול להיות כי הוא הסכום של התא arr[0]. 2 יכול להיות כי הוא הסכום של התא arr[1]. 3 יכול להיות כי הוא הסכום של התאים arr[0] ו-arr[1]. 4 לא יכול להיות סכום של חלק מאיברי המערך (לאו דווקא רצופים). לכן השיטה תחזיר 4.

- עבור המערך arr הבא:  $arr[] = \{1, 1, 1, 1\}$  השיטה תחזיר 5
- עבור המערך arr הבא:  $arr[] = \{1, 1, 3, 4\}$  השיטה תחזיר 10
- עבור המערך arr הבא:  $arr[] = \{1, 3, 5, 10, 20, 40\}$  השיטה תחזיר 2
- עבור המערך arr הבא:  $arr[] = \{1, 2, 4, 10, 11, 15\}$  השיטה תחזיר 8

אפשר להניח שהמערך מלא במספרים שלמים חיוביים והוא ממוין בסדר לא יורד. אין צורך לבדוק זאת.

שימו לב:

- השיטה שתכתבו צריכה להיות יעילה ככל הניתן, גם מבחינת סיבוכיות הזמן וגם מבחינת סיבוכיות המקום. תשובה שאינה יעילה מספיק כלומר, שתהיה בסיבוכיות גדולה יותר מזו הנדרשת לפתרון הבעיה תקבל מעט נקודות בלבד.

- כתבו מה סיבוכיות הזמן וסיבוכיות המקום של השיטה שכתבתם.
- אל תשכחו לתעד את מה שכתבתם!



**חלק ב - את התשובות לשאלות 3-6 יש לכתוב על גבי השאלון.  
לא נבדוק תשובות שייכתבו במקום אחר!**

**שאלה 3 - 12 נקודות**

נניח שהמחלקה Node שלהלן מממשת צומת של עץ חיפוש בינרי.

```
public class Node
{
    private int _number;
    private Node _leftSon, _rightSon;

    public Node (int number)
    {
        _number = number;
        _leftSon = null;
        _rightSon = null;
    }

    public int  getNumber()      {return _number; }
    public Node getLeftSon()     {return _leftSon; }
    public Node getRightSon()    {return _rightSon; }

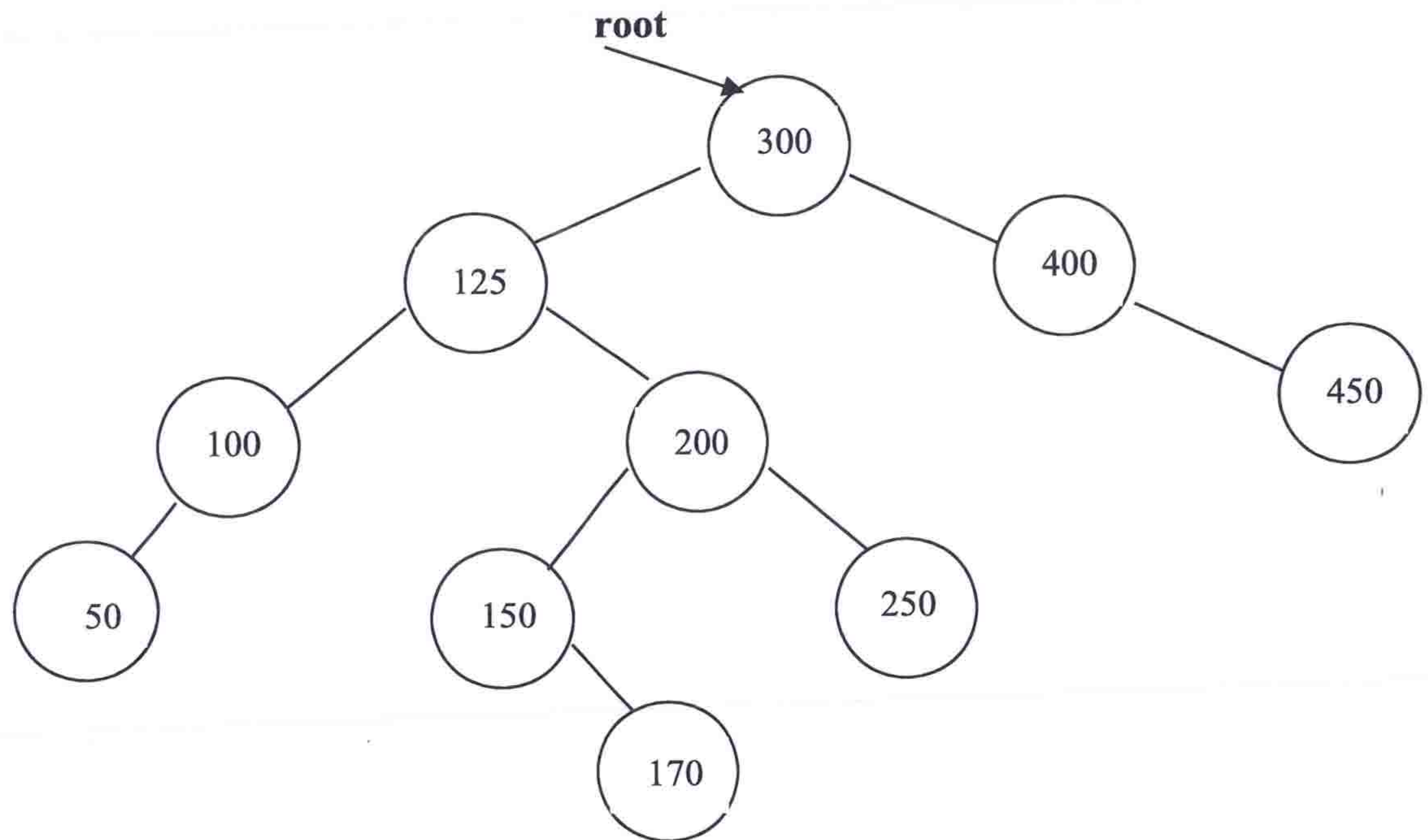
    public void setNumber(int number) { _number = number; }
    public void setLeftSon(Node node) { _leftSon = node; }
    public void setRightSon(Node node) { _rightSon = node; }
}
```

נתונה השיטה what הבאה שמקבלת כפרמטר את node מטיפוס Node שמצביע לשורש של עץ חיפוש בינרי, ומספר שלם num.

```
public static void what(Node node, int num)
{
    if (node!=null) {
        if (node.getNumber() > num)
        {
            what (node.getLeftSon(), num);
            System.out.println(node.getNumber());
            what (node.getRightSon(), num);
        }
        if (node.getNumber() <= num)
            what (node.getRightSon(), num);
    }
}
```



נתון עץ חיפוש בינרי הבא, ששורשו הוא root



ענו על שני הסעיפים הבאים:

5 נק' (i) מה יודפס לאחר שנפעיל את השיטה what עם שני הפרמטרים השורש root שלעיל, והמספר num = 150? (הערה: מספר השורות הריקות לא בהכרח תואם את מספר השורות שיופיעו בפלט.)

התשובה היא:

---

---

---

---

---

---

---

---

7 נק' (ii) מה מבצעת השיטה what באופן כללי? הסבירו בקצרה מה עושה השיטה ולא כיצד היא מבצעת זאת. שימו לב, עליכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. התייחסו למקרי הקצה השונים.

התשובה היא:

---

---

---

---



## שאלה 4 - 12 נקודות

נתונה המחלקה הבאה:

```
public abstract class A1 {
    private int x;
    public A1(int x) {
        this.x = x;
    }
    public abstract void f(int y);
}
```

בכל אחד מהסעיפים הבאים ישנה הגדרה של מחלקה נוספת. עבור כל סעיף כתבו האם קוד המחלקה עובר קומפילציה, או לא (כתבו מדוע). 2 נקודות לכל סעיף.

א.

```
public class B1 extends A1 {
    public B1(int x) {
        super(x);
    }
}
```

✓

ב.

```
public abstract class B2 extends A1 {
    public B2(int x) {
        super(x);
    }
}
```

✓

ג.

```
public abstract class B3 extends A1 {
}
```

✗



.ד

```
public class B4 extends A1 {  
    public B4() {  
        super(2);  
    }  
    public void f(int x) { x = 7; }  
}
```

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

.ה

```
public abstract class B5 extends A1 {  
    public B5() {  
        super(2);  
    }  
    public void f(double x) { super.x = 8.5; }  
}
```

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

.ו

```
public class B6 extends A1 {  
    public B6() {  
        super(2);  
    }  
    public void f(int x) {  
        A1 a1 = new B6();  
        System.out.println(a1);  
    }  
}
```

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_



## שאלה 5 - 12 נקודות

נתונה המחלקה IntNode הבאה, המייצגת איבר ברשימה:

```
public class IntNode {
    private int _value;
    private IntNode _next;

    public IntNode(int val, IntNode n) {
        _value = val;
        _next = n;
    }

    public int getValue() {
        return _value;
    }
    public IntNode getNext() {
        return _next;
    }

    public void setValue(int v) {
        _value = v;
    }
    public void setNext(IntNode node) {
        _next = node;
    }
}
```

נתונה רשימה מקושרת של מספרים שלמים, הממומשת בעזרת המחלקה IntList שלהלן:

```
public class IntList
{
    private IntNode _head;

    public IntList() {
        _head = null;
    }

    public boolean f(int num)
    {
        IntNode p=_head;
        while (p != null)
        {
            if (p.getValue() == num)
                return true;
            p = p.getNext();
        }
        return false;
    }
}
```



```

public int g( int num)
{
    int t1 = num;
    int t2 = num;

    while (f(t1))
        t1++;
    while (f(t2))
        t2--;

    return (t1-num) + (num-t2) -1;
}

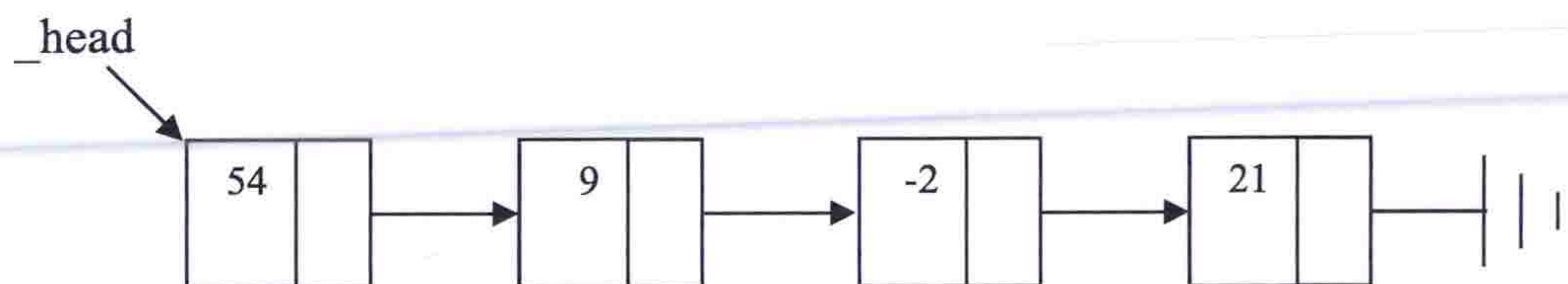
public int what()
{
    int m = 0;
    IntNode p=_head;
    while (p != null)
    {
        int res = g(p.getValue());
        if (res > m)
            m = res;
        p = p.getNext();
    }
    return m;
}

. . . // other methods
}

```

אתם יכולים להניח שהרשימה מלאה במספרים שלמים, ואורכה הוא n.

בטענות להלן, נסמן את איברי הרשימה כמספרים מופרדים בפסיקים, בתוך סוגריים מסולסלים. כך לדוגמא, נסמן { 54 , 9 , -2 , 21 } את הרשימה שלהלן:





סעיף א (4 נקודות)

הקיפו בעיגול את כל הטענות הנכונות שכתובות להלן:

- א) אם נפעיל את השיטה  $g$  על הרשימה  $\{9, 7, 2, 8, 3\}$  והמספר  $num = 4$ , נקבל את התשובה 2.
- ב) אם נפעיל את השיטה  $g$  פעם אחת על הרשימה  $\{9, 7, 2, 8, 3\}$  והמספר  $num = 8$  ופעם שניה על הרשימה  $\{2, 3, 7, 8, 9\}$  והמספר  $num = 7$  נקבל אותה תשובה.
- ג) אם נפעיל את השיטה  $what$  פעם אחת על הרשימה  $\{2, 7, 2, -8, 3\}$  ופעם שניה על הרשימה  $\{12, 3, -1, 5, 9, -3, 16, 7\}$  נקבל אותה תשובה.
- ד) אם נפעיל את השיטה  $what$  פעם אחת על הרשימה  $\{2, 5, -8, 1, -1, 7, -3, 6, -2\}$  ופעם שניה על הרשימה  $\{10, 7, 3, -1, 5, 9, -3, 8\}$  נקבל אותה תשובה.

סעיף ב (8 נקודות)

מה מבצעת השיטה  $what$  באופן כללי? הסבירו בקצרה מה השיטה עושה ולא כיצד היא מבצעת זאת.

שימו לב, עליכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כלומר, עליכם לכתוב מה המשמעות של הערך שמוחזר מהשיטה  $what$ , כשהיא מופעלת על רשימה כלשהי.

---

---

---

---

---

---

X



## שאלה 6-14 נקודות

נתונות שתי השיטות הסטטיות f ו-what הבאות. המערך arr מלא במספרים שלמים ממוינים  
בסדר עולה ממש.

```
public static int f (int [] arr, double x)
{
    int i= 0, j = arr.length-1;
    if (arr.length < 1 || arr[0] > x)
        return -1;
    while (i<=j)
    {
        int m = (i+j)/2;
        if (arr[m] == x)
            return m;
        if (arr[m] < x)
            i = m+1;
        else
            j = m-1;
    }
    return j;
}

//-----
public static boolean what (int [] arr, int k, double x)
{
    int le, ri, sum;
    le = f (arr, x);
    if (le == -1)
        return false;
    ri = le;
    sum = arr[le];
    for (int ind=1; ind<k; ind++)
    {
        if (ri < arr.length-1 && sum < x*ind)
        {
            ri++;
            sum += arr[ri];
        }
        else if (le>0)
        {
            le--;
            sum += arr[le];
        }
    }
    if (sum == x*k)
        return true;
    return false;
}
```



### סעיף א (3 נקודות)

מה מבצעת השיטה  $f$  באופן כללי? הסבירו בקצרה מה השיטה עושה ולא כיצד היא מבצעת זאת.  
שימו לב, עליכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כלומר, עליכם לכתוב מה המשמעות של הערך שמוחזר מהשיטה  $f$ .

### סעיף ב (3 נקודות)

מה מחזירה השיטה  $what$  כאשר היא מקבלת את הפרמטרים הבאים:

$$arr = \{-5, 2, 3, 4, 15\}, k = 3, x = 3$$

### סעיף ג (3 נקודות)

מה מחזירה השיטה  $what$  כאשר היא מקבלת את הפרמטרים הבאים:

$$arr = \{-5, 2, 3, 4, 15\}, k = 3, x = 6$$

### סעיף ד (5 נקודות)

מה מבצעת השיטה  $what$  באופן כללי? הסבירו בקצרה מה השיטה עושה ולא כיצד היא מבצעת זאת.

שימו לב, עליכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כלומר, עליכם לכתוב מה המשמעות של הערך שמוחזר מהשיטה  $what$ .

בהצלחה