

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

שאלה 1 - 25 נקודות

נתונים:

- מערך דו-ממדי mat של מספרים שלמים וחיוביים (בגודל n שורות ו- m עמודות).
- מערך חד-ממדי arr של מספרים שלמים וחיוביים (בגודל m)
- מספר שלם k

אנחנו רוצים לבדוק האם קיימות לכל היותר k שורות של המערך mat ($1 \leq k \leq n$) כך שאיחוד השורות האלו מכיל את כל המספרים שבמערך arr .

לדוגמא,

אם המערכים mat ו- arr הם כדלהלן:

	0	1	2
0	1	5	7
1	3	2	9
2	1	2	3

mat

0	1	2
7	2	3

arr

עבור $k = 2$ נקבל תשובה חיובית כי שורה מספר 0 ושורה מספר 2 במערך mat המכילות את המספרים 1, 5, 7, 2, 3 מכילות את כל מספרי המערך arr . גם שורות 0 ו- 1 במערך המכילות את המספרים 1, 5, 7, 3, 2, 9 מכילות את כל מספרי המערך arr . כמו כן ברור שכל $k > 2$ מקיים את התנאי.

אולם עבור $k = 1$ נקבל שהתשובה שלילית כיוון שאין שורה אחת במערך mat שמכילה את כל איברי המערך arr .

עליכם לכתוב שיטה **בוליאנית** ו**קורסיבית** המקבלת כפרמטרים מערך דו-ממדי mat , מערך חד-ממדי arr ומספר שלם k . השיטה צריכה להחזיר את הערך `true` אם יש k שורות (או פחות) במערך mat שאיחודן מכיל את מספרי המערך arr , ו- `false` אחרת.

חתימת השיטה היא:

```
public static boolean covers(int[][] mat, int[] arr, int k)
```


אתם יכולים להניח שהמערכים מלאים במספרים שלמים חיוביים, וכן שהם בגדלים המתאימים. אין צורך לבדוק זאת.

לנוחותכם, הגדרנו כמה שיטות שאתם יכולים (אך לא חייבים) להשתמש בהן. אין צורך לממש אותן:

1. `private static boolean allZero(int[] x)`

השיטה מחזירה true אם כל איברי המערך x שווים ל-0, ו-false אחרת.

2. `private static void copy(int[] dest, int[] src)`

השיטה מעתיקה את כל איברי המערך src למערך dest. אפשר להניח שהאזורים שווים

3. `private static void xZero(int[] x, int[] y)`

השיטה מאפסת את כל איברי המערך x אשר מופיעים במערך y.

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

נגדיר שלשה פיתגורית (Pythagorean triple) כשלשה של מספרים שלמים חיוביים a, b ו- c המקיימים את משפט פיתגורס. כלומר המקיימים את המשוואה $a^2 + b^2 = c^2$. עליכם לכתוב שיטה בוליאנית יעילה המקבלת כפרמטר מערך חד-ממדי `arr` המלא במספרים שלמים הממוינים בסדר עולה ממש ומחזירה `true` אם במערך קיימת שלשה פיתגורית ו- `false` אם לא. שימו לב, אין צורך להדפיס את השלשה הזו, אלא רק להחזיר אם יש כזו או לא.

חתימת השיטה היא:

```
public static boolean isPythagorean (int[] arr)
```

לדוגמא,

- עבור המערך `arr` הבא:

0	1	2	3	4
1	3	4	5	6

השיטה תחזיר `true` שכן יש שלשה פיתגורית $(3, 4, 5)$ שכן $3^2 + 4^2 = 9 + 16 = 25 = 5^2$. המספרים לא חייבים להיות בתאים רצופים. בשלשה $(3, 4, 5)$ הם כן רצופים כי המערך ממוין.

- עבור המערך `arr` הבא:

0	1	2	3	4
4	5	6	10	12

השיטה תחזיר `false` שכן אין במערך אף שלשה פיתגורית.

אפשר להניח שהמערך מלא במספרים חיוביים שלמים והוא ממוין בסדר עולה ממש. 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. רמז: הסיבוכיות הנדרשת כאן היא ריבועית.

- מותר לשנות את הערכים במערך!
- 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

א. נניח שהמחלקה Node שלהלן מממשת צומת של עץ בינרי.

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

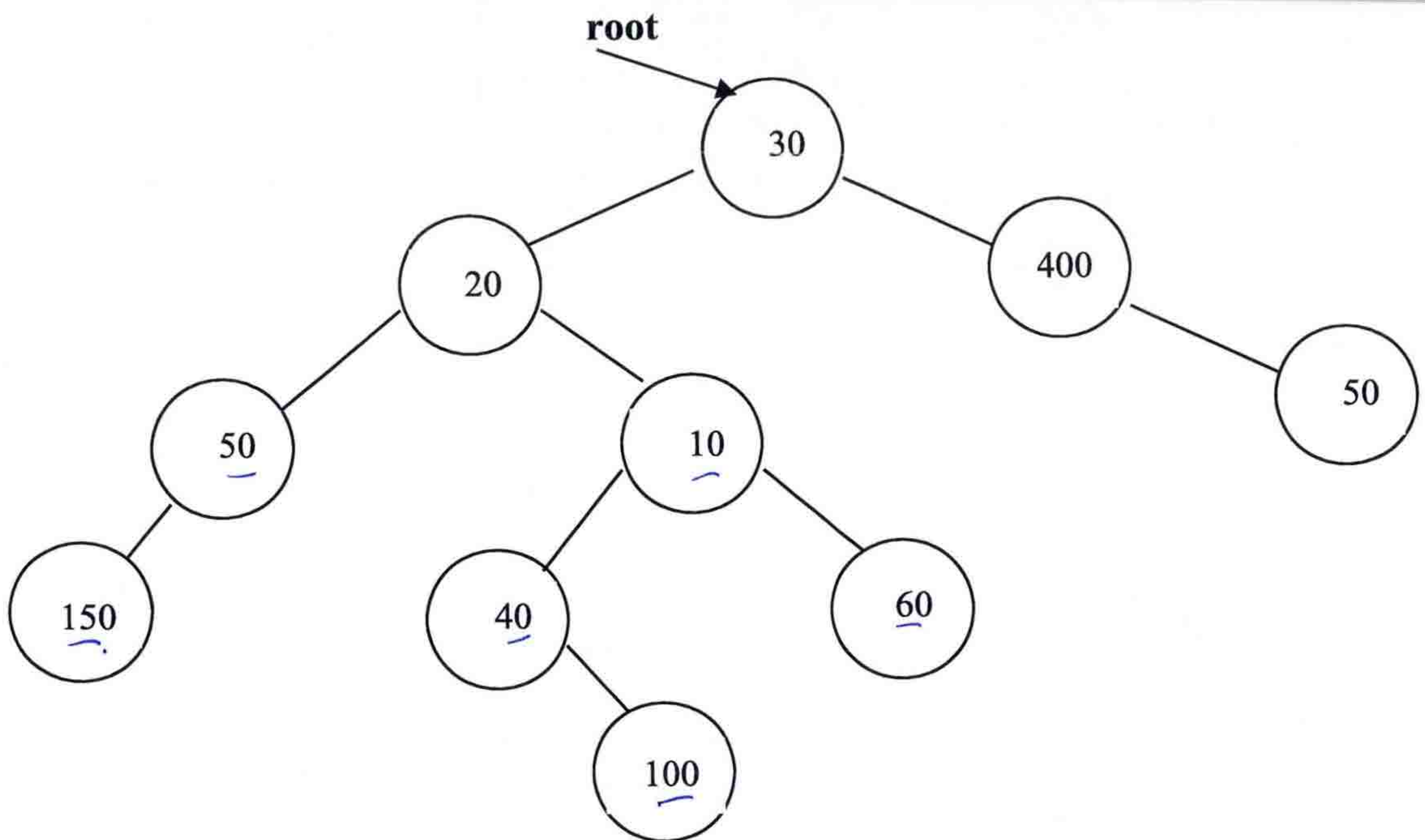
    public Node (int number)
    {
        _number = number;
        _leftSon = null;
        _rightSon = null;
    }

    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 n מטיפוס Node שמצביע לשורש של עץ בינרי.

```
public static int what (Node n)
{
    if (n == null)
        return 0;
    else
    {
        int x = what (n.getLeftSon());
        int y = what (n.getRightSon());
        int num = n.getNumber() + x + y;
        n.setNumber(num);
        return num;
    }
}
```

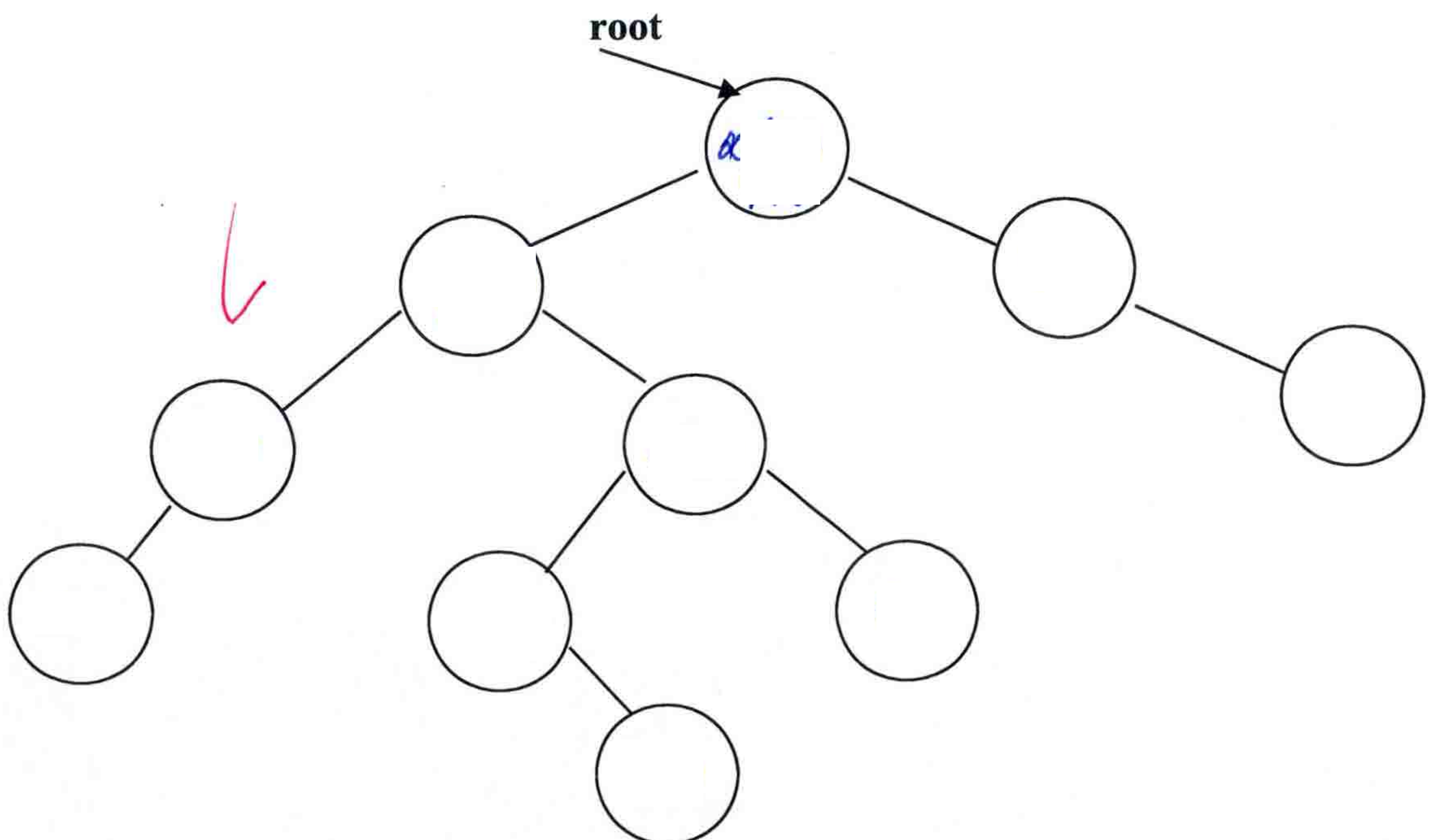
ענו על שלושת הסעיפים הבאים:

2 נק' (i) איזה ערך תחזיר השיטה what אם נקרא לה עם השורש root שלעיל ?

התשובה היא:

4 נק' (ii) ציירו את העץ המתקבל ששורשו root לאחר הקריאה לשיטה what.

התשובה היא:



6) (נק') (iii) 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

נתונות המחלקות הבאות (כל מחלקה בקובץ נפרד, כמובן):

```
public class AA {
    private String str1;

    public AA() {
        this.str1 = "Hello";
        System.out.println("in AA's" + "constructor\n"+ str1);
    }

    public String getStr1() {
        return str1;
    }

    public void h() {
        System.out.println("in AA's h");
    }

    public void f(){
        System.out.println("in AA's f");
    }

    public void g() {
        System.out.println("in AA's g");
    }

    public String toString() {
        return str1;
    }
}
```



```

public class BB extends AA{
    private String str2;

    public BB() {
        str2 = "Good Job!";
        System.out.println("in BB's" + "constructor\n" + str2);
    }

    public BB(AA a) {
        if(a instanceof BB){
            str2 = ((BB) a).str2;
            System.out.println(str2);
            a.f();
        }
        else
            str2 = "Done";
    }

    public void h() {
        super.h();
        System.out.println("in BB's h");
    }

    public void f() {
        System.out.println("in BB's f");
    }

    public String toString() {
        return str2;
    }
}

```

ונתון ה-main הבא :

```

public class Main {
    public static void main(String[] args) {
        // ****
    }
}

```

בכל אחד מהסעיפים הבאים נתון קוד. עבור כל סעיף כתבו מה יקרה אם נכתוב את הקוד בתוך ה-main במקום השורה המסומנת ב ****
 כתבו האם תהיה שגיאת קומפילציה (כתבו מדוע) או שתהיה שגיאת ריצה (כתבו מדוע) או שכתבו מה יהיה הפלט על המסך.
 מספר השורות הריקות לא בהכרח תואם את הדרוש בפלט.
2 נקודות לכל סעיף.

א.

```
AA a1 = new AA();
```

_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

ב.

```
AA a2 = new BB();  
BB b1 = (BB) a2;  
b1.g();
```

_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

ג.

```
AA a3 = new BB(b1); // הוגדר בטעיף ב b1  
System.out.println(a3 instanceof BB ? "Almost done" : "Just  
starting");
```

_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

ד.

```
AA a4 = new BB(new BB());  
a4.h();
```

—
—
—
—
—
—

—
—
—
—
—
—

ה.

```
AA a5 = new AA();  
BB b2 = new BB(a5);  
System.out.println(b2);
```

.
.
.
.

—
—
—
—
—

⊗

ו.

```
AA a6 = new AA();  
Object obj = a6;  
((BB)obj).f();
```

—
—
—
—

—
—

⊗

שאלה 5 - 14 נקודות

לפניכם המחלקות `IntNode` ו-`IntList`. כל אחת בקובץ נפרד, כמובן.

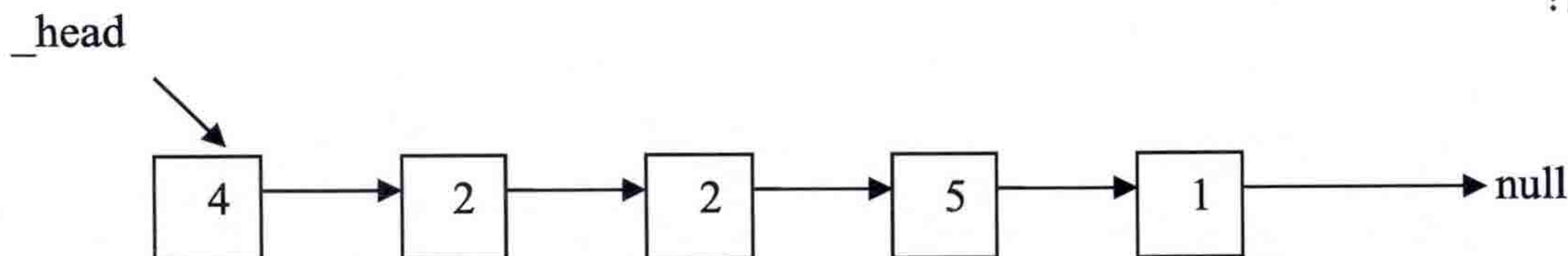
```
public class IntNode
{
    private int _data;
    private IntNode _next;

    public IntNode (int data, IntNode node)
    {
        _data = data;
        _next = node;
    }
    public IntNode getNext()
    {
        return _next;
    }
    public void setNext(IntNode node)
    {
        _next = node;
    }
    public int getData()
    {
        return _data;
    }
    public void setData(int data)
    {
        _data = data;
    }
}

//-----

public class IntList
{
    private IntNode _head;
    public IntList()
    {
        _head = null;
    }
}
```

נתונה רשימה מקושרת של קודקודים מסוג `IntNode`, כאשר `_head` מצביע על האיבר הראשון ברשימה:



אנחנו נכתוב אותה כך `{4, 2, 2, 5, 1}`. בשאלה זו הניחו שכל המספרים ברשימה חיוביים.

תזכורת: השיטה `Math.max(x, y)` מחזירה את המקסימום בין `x` ו-`y`.

השיטה הבאה (method1) נמצאת בתוך המחלקה IntList:

```
public int method1()
{
    if (_head == null)
        return 0;
    int m1 = 0; 3
    int m2 = _head.getData(); 4 3 2 6
    IntNode q = _head;
    for (IntNode p = _head.getNext(); p != null;
        q = q.getNext(), p = p.getNext())
    {
        if (q.getData() < p.getData())
            m2 = m2 + p.getData();
        else
        {
            m1 = Math.max(m1, m2);
            m2 = p.getData();
        }
    }
    return Math.max(m1, m2);
}
```

סעיף א: (3 נקודות)

נפעיל את השיטה method1 על הרשימה {1, 2, 3, 2, 5, 1, 7}. איזה ערך תחזיר השיטה?

סעיף א: (4 נקודות)

נפעיל את השיטה method1 על הרשימה {1, 2, 2, 4}. איזה ערך תחזיר השיטה?

סעיף ב: (7 נקודות)

מה מבצעת השיטה method1 באופן כללי? הסבירו בקצרה מה השיטה עושה ולא כיצד היא מבצעת זאת. הניחו שכל המספרים ברשימה חיוביים.

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. מה משמעותו של הערך המוחזר מהשיטה method1? התייחסו למקרי קצה.

שאלה 6-12 נקודות

נתונות שתי המחלקות Stack ו-Queue המייצגות מחסנית ותור של מספרים שלמים (בהתאמה).
ממשק המחלקה Stack:

```
public class Stack
{
    public Stack()           // יוצר מחסנית ריקה
    public boolean isEmpty() // בודק אם המחסנית ריקה
    public void push(int num) // דוחף לראש המחסנית את num
    public int pop()         // מוציא את האיבר בראש המחסנית
    public int top()         // מחזיר את האיבר בראש המחסנית
                           // בלי להוציא אותו מהמחסנית
}
```

ממשק המחלקה Queue:

```
public class Queue
{
    public Queue()           // יוצר תור ריק
    public boolean isEmpty() // בודק אם התור ריק
    public void enqueue(int num) // דוחף לזנב התור את num
    public int dequeue()      // מוציא את האיבר בראש התור
    public int first()        // מחזיר את האיבר בראש התור
                           // בלי להוציא אותו מהתור
}
```

סעיף א

נתונה השיטה הסטטית fun1 הבאה:

```
public static void fun1(Queue q)
{
    Stack s = new Stack();
    while (!q.isEmpty())
        s.push(q.dequeue());
    while (!s.isEmpty())
        q.enqueue(s.pop());
}
```

מה מבצעת השיטה fun1 בהינתן לה תור q? התייחסו למקרי קצה. 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. (3 נקודות)


```

public static int fun2 (Queue q)
{
    Stack s = new Stack();
    int count = 0;
    while (!q.isEmpty())
    {
        if (s.isEmpty() || s.top() <= q.first())
            s.push(q.dequeue());
        else
            q.enqueue(s.pop());
        count++;
    }
    return count;
}

```

מה מבצעת השיטה fun2 בהינתן לה תור q? התייחסו למקרי קצה. 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. (5 נקודות)

אם מספר האיברים בתור הוא k, מה גודלו המקסימלי של המספר count המוחזר מהשיטה fun2? (4 נקודות)