

הקורס "מבוא למדעי המחשב ושפת Java"

קראו בעיון את ההנחיות שלהלן:

- המבחן בכללותו בנוי מחמישה חלקים. בכל חלק יש שאלות שונות. עליכם לענות על כל השאלות בכל החלקים.
- כל השיטות שאתם כותבים צריכות להיות מתועדות היטב. יש לכתוב תחילה בקצרה את האלגוריתם וכל הסבר נוסף הדרוש להבנת התכנית. יש לבחור בשמות משמעותיים למשתנים, לשיטות עזר (אם יש) ולקבועים. תכנית שלא תתועד כנדרש לעיל תקבל לכל היותר 85% מהניקוד.
- אפשר לתעד בעברית. אין צורך בתיעוד API.
- יש להקפיד לכתוב את התכניות בצורה מבנית ויעילה. תכנית לא יעילה לא תקבל את מלוא הנקודות.
- אם ברצונכם להשתמש בתשובותיכם בשיטה או במחלקה הכתובה בחוברות השקפים, אין צורך שתעתיקו את השיטה או את המחלקה לתשובה שלכם. 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!

מבנה הבחינה:

בבחינה יש חמישה חלקים:

- החלקים הראשון והשני של הבחינה כוללים שאלות שאתם צריכים לכתוב בהן שיטות. בכל שאלה, עליכם להקליד את השיטה בקובץ Word או notepad (ולא ++notepad!), ולשמור אותו כקובץ pdf. את הקובץ הזה תצטרכו להעלות לאתר במקום המיועד לשם כך.
- הקובץ חייב להיות מסוג pdf. שימו לב שהקובץ צריך להיות מוקלד! לא יתקבלו סריקות של קבצים, תמונות וכד'.
- שימו לב שלא תוכלו להעלות קובץ מסוג אחר. גם לא !java
- אסור להשתמש בסביבת עבודה כלשהי של Java! (לא BlueJ, לא eclipse, ולא אף סביבת עבודה או הרצה של תכניות ב-Java).

- שימו לב שאסור להקליד בתוך החלון שיש בשאלה! אם תקלידו בתוך החלון הזה, התשובה שלכם עלולה להימחק על ידי המחשב של האו"פ, שאם מזהה שנכתבה תוכנה, הוא חושד שהיא מזיקה ולכן מוחק אותה ולא נוכל לקרוא את התשובה שלכם. התשובה שלכם צריכה להיות אך ורק בקובץ pdf.
- בשלושת החלקים האחרים תצטרכו לענות על שאלות שהן שאלות רב-ברירה (אמריקאיות), או שאלות השלמה (גם הן רב-ברירה).
- אל תיבהלו מהמספר הרב של ה"שאלות", במודל זה, כל סעיף נחשב כשאלה נפרדת, ולכן מספר השאלות גדול.

אנו מאחלים לכם בהצלחה בבחינה.

במסלול של רכבת יש n תחנות.

מחירי הכרטיסים של הנסיעות מתחנה i לתחנה j (כאשר j גדול ממש i) נתונים במערך דו-ממדי.

לדוגמא, מערך המחירים של הכרטיסים הוא זה:

שימו לב שהערכים בתאים בהם i גדול מ- j הם -1 כי מותר לנסוע רק בכיוון אחד (רק מ- i ל- j שמקיימים $i < j$, וכאשר הם שווים, המחיר הוא 0 , כמובן (נשארים בתחנה...))

	0	1	2	3
0	0	15	80	90
1	-1	0	40	50
2	-1	-1	0	70
3	-1	-1	-1	0

עליכם לכתוב שיטה סטטית רקורסיבית המקבלת מערך דו-ממדי, מלא במספרים שלמים המהווים מערך מחירים של כרטיסים. השיטה צריכה להחזיר את המחיר המינימלי לנסיעה מהתחנה הראשונה 0 לאחרונה $n-1$.

הנחות חשובות:

- המערך תקין, ואכן כל התאים באלכסון הראשי הם 0 ומתחת לאלכסון הראשי הם -1 .
 - כל הנסיעות מתחנה i לתחנה j אפשריות. אין תאים ריקים.
- אין צורך לבדוק את תקינות המערך.

בדוגמא לעיל, המחיר המינימלי הוא 65 , אפשר להגיע אליו אם נוסעים מתחנה 0 לתחנה 1 ומתחנה 1 לתחנה 3 .

חתימת השיטה היא:

```
public static int minPrice(int[][] mat)
```

השיטה צריכה להיות רקורסיבית ללא שימוש בלולאות כלל. גם כל שיטות העזר שתכתבו (אם תכתבו) לא יכולות להכיל לולאות. אפשר להשתמש בהעמסת-יתר (overloading). מותר לשנות את המערך במהלך השיטה, אבל בסופה הוא צריך לחזור למצבו המקורי.

אין צורך לדאוג ליעילות השיטה, אבל כמובן שצריך לשים לב לא לעשות קריאות רקורסיביות מיותרות! אסור להשתמש במשתנים גלובליים (סטטיים)!

אל תשכחו לתעד את מה שכתבתם!

את התשובה יש להקליד בקובץ, לשמור אותו כקובץ pdf ולהעלות את הקובץ כאן למטה. לא ייבדקו פתרונות סרוקים או מצולמים.
ניתן להעלות קבצים מסוג pdf בלבד.

[targil1.pdf](#) 

כתבו שיטה סטטית בוליאנית יעילה המקבלת כפרמטרים, מערך חד-ממדי `arr` המלא במספרים שלמים הממוינים בסדר עולה, ומספר ממשי `x`. השיטה בודקת אם במערך `arr` קיים רצף של איברים עוקבים, שהממוצע שלהם הוא `x`.

אם קיים רצף כזה כנדרש, השיטה תחזיר `true` ותדפיס את האינדקסים של הרצף המבוקש.

אם לא קיים רצף כזה, השיטה תחזיר `false` ותדפיס הודעה שאין רצף כזה.

חתימת השיטה היא:

```
public static boolean findAverage (int [] arr, double x)
```

לדוגמא, עבור המערך `arr` הבא:

0	1	2	3	4	5
2	3	8	14	15	35

- אם $x = 8$, השיטה תחזיר `true` ותדפיס את האינדקסים 2 --- 2 שכן הממוצע של הרצף המכיל את האיבר `arr[2]` הוא 8
- אם $x = 2.5$, השיטה תחזיר `true` ותדפיס את האינדקסים 0 --- 1 שכן הממוצע של הרצף המכיל את הרצף מ- `arr[0]` עד `arr[1]` הוא 2.5, כיוון ש- $(2 + 3)/2 = 2.5$
- אם $x = 10$, השיטה תחזיר `true` ותדפיס את האינדקסים 1 --- 4 שכן הממוצע של הרצף המכיל את הרצף מ- `arr[1]` עד `arr[4]` הוא 10, כיוון ש- $(3+8+14+15)/4 = 10$
- אם $x = 8.5$, השיטה תחזיר `false` ותדפיס הודעה שאין רצף מתאים, שכן אף רצף במערך לא נותן ממוצע של 8.5. (שימו לב כי הממוצע של 3 ו- 14 הוא 8.5, אולם הם לא נמצאים ברצף במערך).



אתם יכולים להניח שהמערך מלא במספרים שלמים והוא ממוין בסדר עולה. 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!

את התשובות יש להקליד בקובץ, לשמור את הקובץ כ- pdf ולהעלות את הקובץ כאן למטה. לא ייבדקו פתרונות סרוקים או מצולמים.
ניתן להעלות קבצים מסוג pdf בלבד.

TARGIL2.pdf 

מידע

סעיף א

ענו על השאלה הבאה:

נתון עץ בינרי ששורשו root ויש בו 10 צמתים (כולל השורש והעלים).

כשנסרוק את העץ בסדר סופי (Post Order) נקבל את הפלט הבא (משמאל לימין):

7 17 26 35 48 56 64 73 89 97

כשנסרוק את העץ בסדר תוכי (In Order) נקבל את הפלט הבא (משמאל לימין):

7 35 17 26 97 48 73 64 56 89

עליכם לבחור מבין התשובות הבאות מה יהיה הפלט כשנסרוק את העץ בסדר תחילי (Pre Order).

שימו לב: המספרים בתשובות מסודרים משמאל לימין.

יש לבחור תשובה אחת:

97 35 7 26 17 89 73 48 56 64

97 26 17 7 35 48 73 89 56 64

97 35 7 26 17 89 73 48 64 56

בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה

97 89 35 26 17 7 73 64 48 56

7 17 26 35 48 56 64 73 89 97

56 64 48 73 89 17 26 7 35 97

97 35 7 17 26 89 73 48 56 64

97 35 7 26 17 48 73 89 56 64

97 89 73 64 56 48 35 26 17 7

97 26 17 7 35 73 89 48 64 56

97 35 7 17 26 89 73 48 64 56

97 35 17 7 26 89 73 56 64 48

97 35 26 17 7 89 73 48 56 64

סעיף ב:

לכל טענה מארבע הטענות הבאות סמנו אם היא נכונה או לא:

בסריקה בסדר תוכי של עץ בינרי, שורש העץ לא יכול להיות ראשון ולא אחרון.

יש לבחור תשובה אחת:

נכון

לא־נכון

שאלה 4

הושלם

ניקוד השאלה:
2.00

אם שני צמתים בעץ לא נמצאים באותה רמה בעץ, ואף אחד מהם אינו השורש, אז בכל סריקה הצומת השמאלי יותר ייסרק לפני הצומת הימני.

יש לבחור תשובה אחת:

נכון

לא־נכון

שאלה 5

הושלם

ניקוד השאלה:
2.00

חיפוש מספר בעץ חיפוש בינרי שמספר איבריו N דורש לכל היותר $\log_2 N$ השוואות.

יש לבחור תשובה אחת:

נכון

לא־נכון

שאלה 6

הושלם

ניקוד השאלה:
2.00

עץ בינרי ללא בנים יחידים מכיל מספר אי-זוגי של צמתים.

יש לבחור תשובה אחת:

נכון

לא־נכון

שאלה 7

הושלם

ניקוד השאלה:
2.00

בחבילה מוגדרות חמש המחלקות A, B, C, D ו-Driver הבאות:

```
public abstract class A
{
    protected int _x;

    public A() {
        _x = 1;
    }

    public abstract int f(int x);

    public void f(A a) {
        _x = a._x;
    }
}

//-----

public class B extends A
{
    public B() {
        super();
    }
    public B(int val) {
        _x = f(val);
    }

    public int f(int x) {
        return _x + x;
    }

    public void f(B b) {
        _x = _x * b._x;
    }
}

//-----
```

```

public class C extends B
{
    public int f(int x) {
        return _x - x;
    }

    public void f(A a) {
        if (a instanceof C)
            _x = _x - a._x;
        else
            super.f(a);
    }

    public void f(B b) {
        _x = _x + b._x;
    }

    public void f(C c) {
        _x = c._x + 1;
    }
}

//-----

```

```

public class D extends B
{
    public D(int val) {
        _x = val - _x;
    }

    public void f(A a) {
        _x = _x + a._x + 1;
    }

    public void f(B b) {
        _x = _x * b._x;
    }

    public void f(D d) {
        _x = d._x - 1;
    }
}

```

במחלקה Driver נמצאת השיטה main שלהלן:

```

public static void main()
{
    A a;
    B b;

    --- כאן יופיעו הפקודות שבשאלות הבאות ---
}

```

עליכם לענות על חמש השאלות הבאות.
שימו לב שהשאלות אינן תלויות אחת בשניה!

נניח שנתונה הפקודה:

```
b = new D(3);
```

מה יהיה ערכה של התכונה `x` של האובייקט `b`?

תשובה:

שאלה 8

הושלם

ניקוד השאלה:
2.00

נניח שנתונות שתי הפקודות הבאות:

```
a = new D(4);  
int k = a.f(a.f(6));
```

מה יהיה הערך של המשתנה `k` לאחר ביצוע הפקודות?

תשובה:

שאלה 9

הושלם

ניקוד השאלה:
3.00

נניח שנתונות שתי הפקודות הבאות:

```
b = new D(6);  
int k = b.f(b.f(6));
```

מה יהיה ערכה של התכונה `x` של האובייקט `b` לאחר ביצוע שתי הפקודות?

תשובה:

שאלה 10

הושלם

ניקוד השאלה:
3.00

נניח שנתונות שלוש הפקודות הבאות:

```
a = new B();  
b = new C();  
a.f(b);
```

מה יהיה ערכה של התכונה `x` של האובייקט `a` לאחר ביצוע שלוש הפקודות?

תשובה:

שאלה 11

הושלם

ניקוד השאלה:
3.00



שאלה 12

הושלם

ניקוד השאלה:

5.00

נניח שנתונות שלוש הפקודות הבאות:

```
D d = new D(9);
```

```
a = new B();
```

```
((A)d).f(a);
```

מה יהיה ערכה של התכונה `_x` של האובייקט `d` לאחר ביצוע שלוש הפקודות?

תשובה:



נתונה המחלקה IntNodeSpecial המייצגת איבר ברשימה, בה מאוחסנים מספרים שלמים. לכל איבר יש שלוש תכונות: המספר השלם num_ ושני מצביעים מטיפוס IntNodeSpecial, והם next_ ו- nextSpecial_.

במחלקה IntNodeSpecial ישנם שני בנאים המתוארים להלן:

1. בנאי המקבל כפרמטר מספר שלם n, ויוצר איבר שבתכונה num_ שלו יהיה הערך n, ובשני המצביעים יהיה null.
 2. בנאי המקבל כפרמטרים מספר שלם n ושני אובייקטים מהמחלקה IntNodeSpecial, ויוצר איבר ברשימה לפי הפרמטרים המתאימים.
- כמו כן, במחלקה ישנן שלוש שיטות אחזור (get) ושלוש שיטות קביעה (set) כמקובל לפי התכונות.
- כתבנו כאן את חתימות הבנאים והשיטות בלבד.

```
public class IntNodeSpecial
{
    private int _num;
    private IntNodeSpecial _next, _nextSpecial;

    public IntNodeSpecial(int n)
    public IntNodeSpecial(int num, IntNodeSpecial n,
                          IntNodeSpecial nSpecial)

    public int getNum()
    public IntNodeSpecial getNext()
    public IntNodeSpecial getNextSpecial()

    public void setNum (int n)
    public void setNext (IntNodeSpecial node)
    public void setNextSpecial (IntNodeSpecial node)
}
```

בנוסף, קיימת במחלקה שיטה setNextSpecial נוספת, שלא מקבלת פרמטר, והיא ממומשת כך:

```
public void setNextSpecial()
{
    IntNodeSpecial temp = this;
    while (temp._next != null && (_num%2!=temp._next._num%2))
        temp = temp._next;
    _nextSpecial = temp._next;
}
```



נתונה המחלקה IntListSpecial המייצגת רשימה שאיבריה הם חוליות מהמחלקה לעיל.

הבאנו כאן את מימוש הבנאי של המחלקה `IntListSpecial`, ואת מימוש השיטה `.setSpecial`.

```
public class IntListSpecial
{
    private IntNodeSpecial _head;

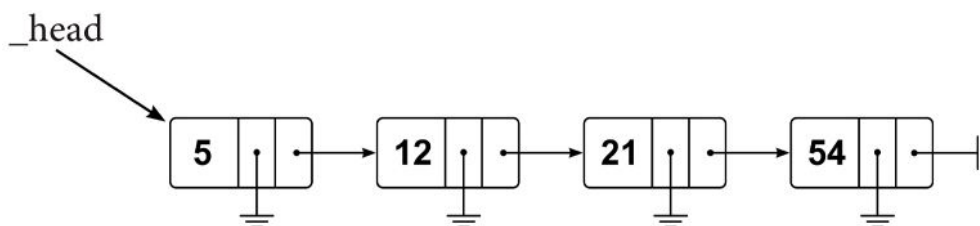
    public IntListSpecial( )
    {
        _head = null;
    }

    public void setSpecial()
    {
        IntNodeSpecial temp = _head;
        while (temp!=null)
        {
            temp.setNextSpecial();
            temp = temp.getNext();
        }
    }
}
```

אתם יכולים להניח שהרשימה מלאה במספרים שלמים. כמו כן ניתן להניח כי לפני הקריאה לשיטה `setSpecial` כל המצביעים מסוג `_nextSpecial` ברשימה עליה הופעלה השיטה הם `null`.

בשאלות להלן, נסמן את איברי הרשימה כמספרים מופרדים בפסיקים, בתוך סוגריים מסולסלים.

כך לדוגמא, נסמן { 5 , 12 , 21 , 54 } את הרשימה הזו:



ענו על השאלות הבאות:



בהנחה שהרשימה list היא כדלהלן (ראש הרשימה הוא המספר השמאלי ביותר):
list = {22, 81, 97, 94, 64, 48, 41}

ענו על השאלות הבאות, בכל מקום הכניסו מספר בלבד, או את הערך null:

לאחר הפעלת השיטה list.setSpecial(),

1. המצביע nextSpecial של האיבר 81, מצביע על האיבר
2. המצביע next של האיבר 81, מצביע על האיבר
3. המצביע nextSpecial של האיבר 94, מצביע על האיבר
4. המצביע next של האיבר 94, מצביע על האיבר

ברצוננו לכתוב במחלקה `IntListSpecial` שיטה בשם `setOneSpecial` שתבצע את מה שביצעה השיטה `setSpecial`, אלא שהיא תבצע זאת במעבר יחיד על הרשימה.

השלימו את השיטה:

```
public void setOneSpecial()
{
    IntNodeSpecial temp1 = null;
    IntNodeSpecial temp2 = null;
    IntNodeSpecial temp = _head;
    while (temp!=null)
    {
        if (  )
        {
            if (  )
            {
                temp2 = temp;
            }
            else
            {
                ;
                ,
            }
        }
        else
        {
            if (  )
            {
                temp1 = temp;
            }
            else
            {
                ;
            }
        }
    }
}
```

v;

}

}

v;

} *// end of while*} *// end of method*