

בחינה בתוכנה 1

סמסטר ב' תש"ף, מועד ב', 8 בספטמבר 2020
לנה דנקין, שי גרשטיין, איתי יצחק

משך הבחינה שלוש שעות.

סך הניקוד על השאלות בבחינה הוא 105, אך הציון המקסימלי אותו ניתן לקבל הוא 100.

יש להניח, אלא אם צויין אחרת, כי:

- הקוד שמופיע במבחן מתאים לגרסא Java8.
- כל החבילות הדרושות יובאו, ואין צורך לכתוב שורות import בגוף הקוד.
- כל מחלקה שהיא public מופיעה בקובץ Java משלה.
- בכל שאלה, כל המחלקות מופיעות באותה חבילה (package).
- בזמן הבחינה, אתם נדרשים לזהות שגיאות קומפילציה שנוצרות כתוצאה מהפרת עקרונות Java-יים ושימוש לא נכון במחלקות/פונקציות. במידה וישנה טעות הקלדה (סוגר חסר, שימוש באות גדולה שלא לצורך וכו') אין לראות בסיבות אלה גורמים לשגיאות קומפילציה.

בבחינה זו מופיע קוד שבחלקו אינו מתקמפל, אינו רץ או שנוגד את הסטנדרטים של Java כפי שנלמדו בקורס, וזאת מתוך מטרה לבחון ידע והבנה של נושאים מסוימים. אין לראות בקטעי קוד אלה דוגמא לכתיבה נכונה ב Java.

הבחינה תיערך בפורמט מקוון. אלה הדגשים למענה על הבחינה המקוונת:

- את התשובות הנכונות יש לסמן בגוף הבחינה במודל.
- בנוסף לסימון הבחינה יש לנמק בתמצות את כל התשובות על דפים נקיים שיוכנו מבעוד מועד.
- יש לנמק את השאלות לפי הסדר, ולסמן בבירור את מספר השאלה אליה מתייחס כל נימוק.

את דפי הנימוקים יש לסרוק בסיום הבחינה ולהעלות למודל. שם הקובץ יהיה answers. נימוק חסר או לא נכון עלול לגרום לאי קבלת נקודות על שאלה במקרים שבהם יוחלט לקבל יותר מתשובה אחת. נימוק לתשובה אחרת מזו שסומנה בטופס הבחינה לא יתקבל.

- בנוסף לדפי נימוקים, אתם יכולים להשתמש בדפי טיוטא שיהיו גם כן דפים לבנים חלקים שיוכנו מבעוד מועד. גם את הטיוטות עליכם לסרוק ולהגיש. שם הקובץ הסרוק יהיה draft.
- במהלך הבחינה לא נדרשת שום הקלדה על המחשב. כל מה שעליכם לעשות בזמן הבחינה הוא לסמן את התשובה הנכונה בכל שאלה ולדפדף בין דפי הבחינה.

בבחינה אסור השימוש בחומר עזר כלשהו, כולל מחשבוני, מחשבים או כל מכשיר אחר פרט לדפים נקיים ולמכשירי כתיבה.

© כל הזכויות שמורות למחברים. מבלי לפגוע באמור לעיל, אין להעתיק, לצלם, להקליט, לשדר, לאחסן במאגר מידע, בכל דרך שהיא, בין מכונית ובין אלקטרונית או בכל דרך אחרת כל חלק שהוא מטופס הבחינה.

בהצלחה!

שאלות 1-3 מתייחסות לקוד הבא. שימו לב, כל השאלות מתייחסות לגירסא הקיימת של הקוד, ללא שינויים מלבד אלה המצויינים בשאלה.

שאלה 1:

מה יודפס בהרצת התוכנית הבאה?

```

/* @imp_inv: if lastIndex > 0, getMax() >= arr[i],
               for i in {0.. lastIndex-1}
 * @imp_inv: if lastIndex == 0, getMax() == Integer.Min_Value */
public class MaxArr {
    public static final int MAX_LEN = 6;
    private int currMax;
    private int[] arr = new int[MAX_LEN];
    private int lastIndex;

    /* @ pre: arr.length < MAX_LEN */
    public MaxArr(int[] arr) {
        currMax = Integer.MIN_VALUE;
        for (int element : arr) {
            addElement(element);
        }
    }

    public float getMax() {
        return currMax;
    }

    public float getLegnth() {
        return lastIndex;
    }

    /* @pre: getLength() < MAX_LEN */
    public void addElement(int element) {
        arr[lastIndex++] = element;
        currMax = Math.max(currMax, element);
    }

    public static void main(String[] args) {
        int[] arr = {1, -3, 5, 2};
        MaxArr mArr = new MaxArr(arr);
        System.out.print((arr == mArr.arr) + " ");
        mArr.addElement(7);
        System.out.print((arr[0] == mArr.arr[0]) + " ");
        System.out.print(mArr.arr[5] == mArr.getMax());
    }
}

```

- .א true true true
- .ב true true false
- .ג true false true
- .ד true false false
- .ה false true true
- .ו false true false
- .ז false false true
- .ח false false false

שאלה זו מצריכה נימוק.

טענה 1 - אם סימנתם שאינה נכונה, הסבירו כיצד מופרת כל איווריאנטה
 טענה 2 - אם סימנתם שהיא נכונה - הסבירו כיצד ניתן לקבל את השגיאה
 טענה 3 - אם סימנתם שאינה נכונה, הסבירו מדוע היא לא מתקיימת

שאלה 2:

לפניכם שלוש טענות הנוגעות לקוד ומתייחסות לשימוש במחלקה MaxArr שאינו בתוך ה main של המחלקה עצמה.

טענה 1: הקוד מקיים של שתי איווריאנטות המימוש

טענה 2: בשימוש תקין במחלקה (על פי החוזה) ניתן לקבל את השגיאה
 addElement בקריאה למתודה ArrayOutOfBoundsException

טענה 3: החוזה מאפשר הכנסת MAX_LEN איברים למבנה הנתונים.

אילו מבין הטענות נכונות? בחרו בתשובה הטובה ביותר:

- א. רק טענה 1 נכונה
- ב. רק טענה 2 נכונה
- ג. רק טענה 3 נכונה
- ד. רק טענות 1+2 נכונות
- ה. רק טענות 1+3 נכונות
- ו. רק טענות 2+3 נכונות
- ז. כל הטענות נכונות
- ח. כל הטענות לא נכונות

שאלה 3: שאלה זו מצריכה נימוק. עבור טענה שסומנה כנכונה, הסבירות באיזה מקרה היא מתקיימת

שאלה 3:

אם נשנה את lastIndex להיות סטטי, מה מהטענות הבאות נכונה? כל הטענות מתייחסות לשימוש במחלקה MaxArr מחוץ ל main של המחלקה עצמה.

טענה 1: עבור אובייקט כלשהו מטיפוס MaxArr שאליו הוכנסו רק איברים חיוביים גדולים מ 0, השירות getMax יכול להחזיר את ערך שאינו הערך המקסימלי מבין הערכים שהוכנסו למבנה הנתונים.

טענה 2: עבור אובייקט כלשהו מטיפוס MaxArr שאליו הוכנסו רק איברים חיוביים גדולים מ 0, השירות getMax יכול להחזיר ערך שלא הוכנס כלל ע"י הבנאי או ע"י השירות addElement.

טענה 3: ניתן יהיה לקבל שגיאת ArrayOutOfBoundsException במקרים שבהם המימוש הקודם (lastIndex לא סטטי) לא היתה מתקבלת שגיאה.

אילו מבין הטענות נכונות? בחרו בתשובה הטובה ביותר:

- א. רק טענה 1 נכונה
- ב. רק טענה 2 נכונה
- ג. רק טענה 3 נכונה
- ד. רק טענות 1+2 נכונות
- ה. רק טענות 1+3 נכונות
- ו. רק טענות 2+3 נכונות
- ז. כל הטענות נכונות
- ח. כל הטענות לא נכונות

שאלה זו מצריכה נימוק. הסבירו בקצרה, לכל טענה, אם היא נכונה או לא.

שאלה 4:

```
public class A {
    private Integer myInteger = new Integer(1);

    public void f(){
        A a1 = new A(); // #
        A a2 = new A();
        a1 = a2;
    }
    public void g() {
        f();
        // ***
    }
}
```

מה נכון להגיד על מצב הזיכרון בעת הרצת הפונקציה g() כשנגיע לשורה המסומנת ב-***?

טענה 1: כאשר הקוד מגיע לשורה `//***` האובייקט שנוצר בשורה `// #` בוודאות שוחרר ע"י ה garbage collector.

טענה 2: כאשר הקוד מגיע לשורה `//***`, בוודאות קיים אובייקט לפחות אובייקט אחד מסוג Integer בתוכנית.

טענה 3: ניתן לממש את finalize של A כך ששני האובייקטים שנוצרו ב f לא ישוחררו ע"י ה garbage collector.

בחר\י בתשובה הטובה ביותר:

- א. רק טענה 1 נכונה.
- ב. רק טענה 2 נכונה.
- ג. רק טענה 3 נכונה.
- ד. רק טענות 1+2 נכונות.
- ה. רק טענות 1+3 נכונות.
- ו. רק טענות 2+3 נכונות.
- ז. כל הטענות נכונות.
- ח. כל הטענות לא נכונות.

שאלה 5: (הניראות של השדה תוקנה במהלך הבחינה)

```
public abstract class AbsAcc<T> {
    private T blance;
    // some code here
}

public class BankAcc extends AbsAcc<Double>{

    public static void func1(double amount, BankAcc from, BankAcc to){
        // some code here
    }

    public void fucn2(double amount, BankAcc other){
        // some code here
    }

    public static void func3(double amount, AbsAcc from, BankAcc to){
        // some code here
    }

    public void fucn4(double amount, AbsAcc other){
        // some code here
    }
}
```

המחלקה BankAcc מייצגת חשבון בנק. כמות הכסף שנמצאת בחשבון מיוצגת ע"י השדה balance שנורש מ AbsAcct. באילו מבין ארבעת המתודות במחלקה ניתן להשתמש כמתודה שמעבירה כסף בכמות amount שנתונה מחשבון אחד לחשבון אחר?

- א. func1
- ב. func2
- ג. func3
- ד. func4
- ה. ישנן 2 מתודות בדיוק שיכולות להעביר כסף בין 2 חשבונות.
- ו. ישנן 3 מתודות בדיוק שיכולות להעביר כסף בין 2 חשבונות.
- ז. ניתן להשתמש בכל המתודות להעברת כסף בין 2 חשבונות.

שאלה זו מצריכה נימוק. הסבירו בקצרה את תשובתכם.

שאלה 6:

```
public class B {
    private int f = 0;

    public static void main(String[] args) {
        B b1 = new B();
        B b2 = new B();
        Object b3 = b1;
        System.out.println(/***/);
    }

    @Override
    public boolean equals(Object obj) {
        f++;
        B b = (B) obj;
        if (b.f == (f - 1))
            return eq(b);
        return this == b;
    }

    public boolean eq(B b) {
        return super.equals(b);
    }
}
```

הקוד במקום המסומן `/***/` חסר. קיבעו באיזו שורת קוד ניתן להחליף את הקטע `/***/` כך שהרצה של הקוד הנ"ל תדפיס `False`?

- א. `b1.equals(b3)`
- ב. `b1.equals(b1)`
- ג. `b1.equals(b2)`
- ד. `b1 == b3`
- ה. בזמן ריצה תיזרק שגיאת המרה `ClassCastException`

שאלה 7: שאלה זו מצריכה נימוק. הסבירו את תשובתכם בקצרה

נתונות מחלקה A כלשהי. הנראות של כלל המתודות והשדות במחלקה A היא public. הסטודנטית שי מעוניינת ליצור מחלקה חדשה B שתשתמש בפונקציונליות של A, ולכן היא מגדירה את B כך שתירש מ A.

בחר/י את התשובה הנכונה ביותר בהתייחסות לכל שימוש עתידי אפשרי שניתן לעשות במחלקה B:

- א. שי יכולה להשתמש בהאצלה בלבד של אובייקט מטיפוס A בתוך המחלקה B במקום להשתמש בירושה.
- ב. שי יכולה להשתמש בהאצלה + הכלה (אך לא האצלה בלבד) של אובייקט מטיפוס A בתוך המחלקה B במקום להשתמש בירושה.
- ג. שי יכולה להגדיר את מחלקה B בתור מחלקה פנימית בתוך המחלקה A במקום להשתמש בירושה.
- ד. רק אם A מחלקה אבסטרקטית לא ניתן להשתמש בהכלה והאצלה במקום ירושה, ובכל מקרה אחר כן ניתן להשתמש בהכלה והאצלה.
- ה. מלבד תשובה זו, יש יותר מתשובה אחת נכונה.
- ו. 1 מלבד תשובה זו, כל התשובות לא נכונות.

תזכורת:

- הכלה (aggregation) – במחלקה א' יש שדה מטיפוס מחלקה ב'.
- האצלה (delegation) – קוראים מתוך מתודות במחלקה א' למתודות של מחלקה ב'.

שאלה זו מצריכה נימוק. לכל פונקציה שלא מתקמפלת, הסבירו איזה היבט של בטיחות טיפוסים היה נפגע לו הקוד היה מתקמפל

שאלה 8:

```
public class A<S> {
    public void f1(Collection<Number> l1, List<? extends Number> l2) {
        l1 = l2;
    }

    public <E> void f2(List<? super E> l, E elm) {
        l.add(elm);
    }

    public void f3(Collection<? extends S> c, List<S> l ) {
        c = l;
    }
}
```

אילו מבין הפונקציות f המופיעות במחלקה A מתקמפלות? בחר/י את התשובה הטובה ביותר .

- א. רק f1 מתקמפלת.
- ב. רק f2 מתקמפלת.
- ג. רק f3 מתקמפלת.
- ד. רק f1+f2 מתקמפלות.
- ה. רק f1+f3 מתקמפלות.
- ו. רק f2+f3 מתקמפלות.
- ז. כל הפונקציות מתקמפלות.
- ח. כל הפונקציות לא מתקמפלות.

שאלה 9:

לפניכם 3 טענות הקשורות לקוד גנרי (תזכורת, מחלקה גנרית היא מחלקה שהגדרה שלה מכילה פרמטר גנרי, למשל, המחלקה A מהשאלה הקודמת).

טענה 1: מחלקה לא גנרית יכולה לממש ממשק גנרי.

טענה 2: מחלקה אבסטרקטית יכולה להיות גנרית.

טענה 3: מתודה גנרית לא בהכרח נמצאת בתוך מחלקה גנרית.

בחר/י בתשובה הטובה ביותר:

- א. כל הטענות לא נכונות.
- ב. רק טענה 1 נכונה
- ג. רק טענה 2 נכונה
- ד. רק טענה 3 נכונה
- ה. רק טענות 1+2 נכונות
- ו. רק טענות 1+3 נכונות
- ז. רק טענות 2+3 נכונות
- ח. כל הטענות נכונות


```

public class A {
    public static void main(String[] args) {
        int x = 0;
        int[] a = { 1, 2, 3};
        for (int i : a) {
            try {
                if (i % 2 == 0) throw new Exception();
                else throw new RuntimeException();
            }
            catch (RuntimeException e) { x = x - 3; }
            catch (Exception e) { x++; }
            finally { x++; }
        }
        System.out.println(x);
    }
}

```

מה יקרה בהרצת הקוד הבא?

- א. תיזרק שגיאה לפני שהתוכנית תדפיס פלט כלשהו.
- ב. ריצת התוכנית תסיים בהצלחה ויודפס 3-.
- ג. ריצת התוכנית תסתיים בהצלחה ויודפס 2-.
- ד. ריצת התוכנית תסתיים בהצלחה ויודפס 1-.
- ה. ריצת התוכנית תסתיים בהצלחה ויודפס 0.
- ו. ריצת התוכנית תסתיים בהצלחה ויודפס 1.
- ז. ריצת התוכנית תסתיים בהצלחה ויודפס 2.
- ח. ריצת התוכנית תסתיים בהצלחה ויודפס 3.

שאלה זו מצריכה נימוק. הסבירו את תשובתכם בקצרה.

שאלה 11:

```
public class A {
    private int i, j;

    public A(int i, int j) {
        this.i = i;
        this.j = j;
    }
    // the rest of the code is not provided

    @Override
    public int hashCode() {
        return i*j;
    }

    public static void main(String[] args) {
        Set<A> s = new HashSet<>();
        s.add(new A(3, 1));
        s.add(new A(2, 1));
        s.add(new A(1, 3));
        s.add(new A(3, 1));
        System.out.println(s.size());
    }
}
```

להלן מספר טענות המתייחסות למימושים השונים של המחלקה A. הניחו כי הקוד הקיים של A לא ישתנה, וניתן רק להוסיף קוד. כמו כן, הניחו כי הקוד של A הוא דטרמיניסטי, כלומר, פונקציות שלה מחזירות ערכים שאינם אקראיים (Random).

- טענה 1: קיים מימוש של A עבורו יודפס 2.
- טענה 2: קיים מימוש של A עבורו יודפס 3.
- טענה 3: קיים מימוש של A עבורו יודפס 4.
- טענה 4: קיים מימוש של A עבורו יודפס 5.

בחרו את התשובה הטובה ביותר:

- א. רק טענה 1 לא נכונה.
- ב. רק טענה 2 לא נכונה.
- ג. רק טענה 3 לא נכונה.
- ד. רק טענה 4 לא נכונה.
- ה. רק טענות 1 ו-3 לא נכונות.
- ו. רק טענות 2 ו-3 לא נכונות.
- ז. רק טענות 1 ו-2 לא נכונות.
- ח. רק טענות 3 ו-4 לא נכונות.

שאלה 12: **שאלה זו מצריכה נימוק. הסבירו את תשובתכם בקצרה**

```
public class A {
    private A() {}
    // the rest of the code is not provided
}
```

לפניכם הקוד החלקי של מחלקה A.

ידוע כי A היא מחלקה אשר אין בה שום מחלקה פנימית/מקוננת/אנונימית ויש לה בנאי יחיד.

בחרו בתשובה הנכונה ביותר.

- א. בריצה של תוכנית כלשהי לא יתכנו מופעים של המחלקה A, אך יתכנו מופעים של מחלקות שיורשות מ-A.
- ב. בריצה של תוכנית יתכנו מופעים של המחלקה A, אך לא יתכנו מופעים של מחלקות שיורשות מ-A.
- ג. בריצה של תוכנית לא יתכנו מופעים של המחלקה A או של מחלקות שיורשות מ-A.
- ד. בריצה של תוכנית יתכנו גם מופעים של המחלקה A וגם של מחלקות שיורשות מ-A.
- ה. ריצה של תוכנית לא תושפע אם בקוד של A נשנה את הנראות של הבנאי הריק לפומבית, ונוסיף לחתימה של המחלקה A את שני ה-modifiers: final abstract.
- ו. מלבד תשובה זו, יש 2 תשובות נכונות.

שאלה 13: **שאלה זו מצריכה נימוק. לכל שורת קוד, הסבירו אם היא מתקמפלת או לא**

```
public class Box<T extends Comparable> {

    public static <T extends Comparable> int comp(T v1, T v2) {
        return v1.compareTo(v2);
    }

    public int f1(List<? extends T> l) {
        return comp(l.get(0), l.get(1));
    }

    public int f2(List<T> l) {
        return comp(l.get(0), l.get(1));
    }

    public static void main(String[] args) {
        Box<String> b = new Box<>();
        b.f1(Arrays.asList(1,2)); // *
        b.f2(Arrays.asList(1,2)); // $
        b.f1(Arrays.asList("1","2")); // #
        b.f2(Arrays.asList("1","2")); // @
    }
}
```

אילו מהקריאות שנמצאות בתוך מתודת ה-main מתקמפלות?

בחר\י בתשובה הטובה ביותר:

- א. רק \$.
- ב. רק #.
- ג. רק @.
- ד. רק # ו-@.
- ה. רק @ ו-\$.
- ו. כולן מלבד *.
- ז. כולן מלבד \$.
- ח. כולן מלבד #.

שאלה זו מצריכה נימוק. הסבירו בקצרה את תשובתכם

שאלה 14:

נתון הקוד הבא:

```
public class A {
    void f() { System.out.println("A"); }
}

public class B extends A implements I {
    private void f() {}; // line 1
    public void g() throws Exception {}; // line 2
}

public interface I {
    void f();
    public default void g() throws Exception {};
}
```

באפשרותנו למחוק אפס, אחת או שתיים מהחתימות (שורות 1 ו-2) במחלקה B. קבעו באילו מקרים הקוד יתקמפל:

- א. הקוד יתקמפל רק במקרה בו נשאיר את שתי השורות.
- ב. הקוד יתקמפל רק במקרה בו נמחק את השורה הראשונה בלבד.
- ג. הקוד יתקמפל רק במקרה בו נמחק את השורה השניה בלבד.
- ד. הקוד יתקמפל אם נמחק את השורה הראשונה, ולא משנה אם נמחק את השניה.
- ה. הקוד יתקמפל אם נמחק את השורה השניה, ולא משנה אם נמחק את הראשונה.
- ו. מלבד תשובה זו כל התשובות לא נכונות.

שאלה 15:

לפניכם 3 טענות הקשורות לקוד גנרי:

טענה 1: בתוך מתודת מופע של מחלקה עם טיפוס גנרי T, ניתן לייצר אובייקט חדש ע"י קריאה ל $\text{new T}()$

טענה 2: מתודה סטטית גנרית לא יכולה לעשות שימוש בפרמטר הגנרי של המחלקה.

טענה 3: מתודה סטטית שאינה גנרית לא יכולה לעשות שימוש במתודה סטטית גנרית מאותה המחלקה.

בחר\י בתשובה הנכונה ביותר:

- א. כל הטענות לא נכונות
- ב. רק טענה 1 נכונה
- ג. רק טענה 2 נכונה
- ד. רק טענה 3 נכונה
- ה. רק טענות 1+2 נכונות
- ו. רק טענות 1+3 נכונות
- ז. רק טענות 2+3 נכונות
- ח. כל הטענות נכונות

שאלה זו מצריכה נימוק. לכל מימוש, ציינו אם הוא תקין או לא, והסבירו בקצרה

שאלה 16:

נתון הקוד הבא:

```
public interface Decidable {
    int i = 0;
    public int decide(char a, char b);
}

public class PrintDecidable {

    public static void printDecision(Decidable item) {
        System.out.print(item.decide('x', 'y'));
    }

    public static void main(String[] args) {
        *****;
    }
}
```

נתונים 3 קטעי קוד שונים להחלפת המקטע המסומן ב-*****:

מימוש 1 -

```
printDecision((a,b) -> new Decidable(){
    public int decide(char a, char b) { return 1; }
});
```

מימוש 2 -

```
printDecision(new Decidable(){
    private int j;
    @Override
    public int decide(char a, char b) {
        return ++j;
    }
});
```

מימוש 3 -

```
printDecision(new Decidable(){
    private int j;
    @Override
    public int decide() {return 1; }
});
```

איזה מימוש מתקמפל והרצתו תייצר את הפלט 1?

בחר/י את התשובה הנכונה ביותר:

- א. רק מימוש 1.
- ב. רק מימוש 2.
- ג. רק מימוש 3.
- ד. רק מימושים 1+2.
- ה. רק מימושים 1+3.
- ו. רק מימושים 2+3.
- ז. כל אחד מהמימושים לא ייצר את הפלט הרצוי.
- ח. כל אחד מהמימושים ייצרו את הפלט המתאים.

שאלה 17:

שאלה זו מצריכה נימוק. לכל טענה נכונה, הסבירו מדוע היא נכונה או לא

שאלה זו מתייחסת למימושן של מחלקות (ולא מנשקים) A,B,C שמימושן אינו נתון. ניתן להניח את קיומן של מחלקות נוספות ללא שום מגבלות.

```
public void func(B b, C c) {
    A a1 = (A) b; // line 1
    A a2 = (C) b; // line 2
    A a3 = (A) c; // line 3
}
```

להלן מספר טענות לגבי הקוד הנתון:

- טענה 1: לכל מימוש של A,B,C, אם שורות 1+2 מתקמפלות, גם שורה 3 מתקמפלת.
- טענה 2: לכל מימוש של A,B,C, אם הקוד מתקמפל, לא תיתכן שגיאת זמן ריצה בשורה 1.
- טענה 3: לכל מימוש של A,B,C, אם נזרקת שגיאת זמן ריצה בשורה 2 בהכרח תיזרק שגיאת זמן ריצה בשורה 3 (אם נשים את שורה 2 בהערה)

בחרו את התשובה הנכונה ביותר:

- א. כל הטענות לא נכונות.
- ב. רק טענה 1 נכונה
- ג. רק טענה 2 נכונה
- ד. רק טענה 3 נכונה
- ה. רק טענות 1+2 נכונות
- ו. רק טענות 1+3 נכונות
- ז. רק טענות 2+3 נכונות
- ח. כל הטענות נכונות

מה יודפס בהרצת התוכנית הבאה?

```
public class A{
    private String str = "1";
    private String foo(){
        return "a";
    }

    private String goo(){
        return "A" + this.foo() + str;
    }

    public static void main(String[] args){
        A a = new B();
        System.out.println(a.goo());
    }
}

public class B extends A{
    private String str = "2";
    private String foo(){
        return "b";
    }
    public String goo(){
        return "B" + foo() + str;
    }
}
```

מה יודפס בהרצת התוכנית A?

- Aa1 א.
- Aa2 ב.
- Ab1 ג.
- Ab2 ד.
- Ba1 ה.
- Ba2 ו.
- Bb1 ז.
- Bb2 ח.

שאלה זו מצריכה נימוק. הסבירו בקצרה איזה איברים עוברים בזרם.

שאלה 19:

מה יודפס בהרצת התוכנית הבאה?

```
public class A{
    public static void main(String[] args){
        Stream<Integer> s = Arrays.asList(1,3,4,2).stream();
        System.out.println("_" + s.filter(x->{
            System.out.print(x);
            return x>2;})
            .map(x-> {
                System.out.print(x);
                return x*2;})
            .sorted()
            .reduce((x,y)->x+y).get());
    }
}
```

בחר/י בתשובה הטובה ביותר:

- א. התוכנית לא תבצע שום הדפסה.
- ב. 134234_14
- ג. 134268_14
- ד. 133442_14
- ה. 136482_14
- ו. _13423414
- ז. _13426814
- ח. _13344214
- ט. _13648214

מה יודפס בהרצת התוכנית Base1 ?

```
public class Base1 {
    public int sum = 0;
    private int i = 3;

    public Base1() {
        sum = i + 3 * getNum();
    }

    public int getNum() {
        return i;
    }

    public static void main(String[] args) {
        Base1 a = new Sub1();
        System.out.println(a.sum);
    }
}

public class Sub1 extends Base1 {
    private int j = 2;

    public Sub1() {
        sum = (sum * 2) + j + getNum();
    }

    @Override
    public int getNum() {
        return j;
    }
}
```

בחרי בתשובה הטובה ביותר:

- 1 א.
- 2 ב.
- 3 ג.
- 4 ד.
- 5 ה.
- 6 ו.
- 7 ז.
- 8 ח.
- 9 ט.
- 10 י.
- 11 יא.
- 12 יב.
- 13 יג.