

מספר סידורי: _____ מספר ת"ז: _____

בחינה בתוכנה 1

מסטר א'+ב' תשע"ז, מועד ב', 13 בספטמבר 2017

ליאור וולף, סיון טולדו, ברית יונגמן, לנה דנקין

משך הבחינה שלוש שעות.

יש להניח:

- הקוד שמופיע במבחן מתאים לגירסא 8 של ג'אווה, אלא אם כן צויין אחרת.
- כל החבילות הדרושות יובאו, ואין צורך לכתוב שורות import בגוף הקוד.
- כל מחלקה שהיא public מופיעה בקובץ java משלה, אלא אם כן צויין אחרת.

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 מסויימים. אין לראות בקטעי קוד אלה דוגמא לכתיבה נכונה ב java.

יש לסמן את התשובה הטובה ביותר בתשובון. לא יינתן ניקוד על סימון תשובה בטופס הבחינה, במחברת הבחינה, או בטופס הנימוקים.

יש לנמק את כל התשובות בטופס הנימוקים המצורף בלבד. נימוק חסר או לא נכון עלול לגרום לאי קבלת נקודות על שאלה גם אם התשובה המסומנת היא הטובה ביותר. המקום המיועד לנימוקים בטופס מספיק לתשובות תמציתיות.

יש לצרף את טופס המבחן למחברת הבחינה. מחברת ללא טופס עזר תפסל. תשובות ונימוקים במחברת הבחינה לא יבדקו.

יש למלא מספר סידורי (מספר מחברת) ומספר ת"ז על כל דף של טופס הבחינה.

אסור השימוש בחומר עזר כלשהו, כולל מחשבוני או כל מכשיר אחר פרט לעט.

לטופס הבחינה מצורף דף לשאלות הסטודנטים. יש לכתוב שאלות שמתעוררות במהלך הבחינה בדף זה ולהעביר לסגל הקורס. שאלות ענייניות תענינה על ידי סגל הקורס בפני כל הנבחנות/ים.

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

בהצלחה!

לפניכם המחלקות Test ו Box. במחלקה Test מופיעה המתודה func שמימשה אינו נתון.

```
class Box{
    private int x;
    public Box(int x){
        this.x = x;
    }

    public boolean equals(Object obj) {
        Box other = (Box) obj;
        return x == other.x;
    }
}

public class Test {
    public static void main(String[] args){
        Box b1 = new Box(2);
        Box b2 = new Box(2);
        Box b3 = b2;
        func(b2);
        System.out.print(b1.equals(b2) + ",");
        System.out.print(b3.equals(b2));
    }

    public static void func(Box b){
        //implementation is not provided
    }
}
```

לפניכם הטענות הבאות:

טענה 1: אם Box תהפוך למחלקה פנימית של Test, ניתן יהיה לממש את func כך שבהרצת התוכנית Test עד סופה יודפס false,false.

טענה 2: אם נוסיף setter ל Box, ניתן יהיה לממש את func כך שבהרצת התוכנית Test עד סופה יודפס false,false.

טענה 3: אם נשנה את הנראות של השדה x במחלקה Box ל public, ניתן יהיה לממש את func כך שבהרצת התוכנית Test עד סופה יודפס false,false.

בחר/י את התשובה הטובה ביותר.

- א. אף טענה אינה נכונה
- 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.

2. שאלה 2:

```
class A{
    public A(){
    }

    public A(int i, int j){
    }
}

class B extends A{
    private int i;

    public B(){
        //this.i = 8;           //line 1
        //super();             //line 2
        //this(5);             //line 3
        //super(1,2);          //line 4
        System.out.println(i);
    }

    public B(int i){
        super(i, i);
    }
}
```

נרצה לבחון את התנהגות הקוד לאחר הוספת כל אחת מבית השורות המסומנות בהערה לבנאי של B. יש לבחון כל שורה בנפרד בהנחה שהיא תהיה שורת הקוד הראשונה בבנאי.

בחר/י את התשובה הטובה ביותר.

- א. אם נוסיף את שורה 1, הקוד לא יתקמפל.
- ב. אם נוסיף את שורה 2, הקוד לא יתקמפל.
- ג. אם נוסיף את שורה 3, הקוד לא יתקמפל.
- ד. אם נוסיף את שורה 4, הקוד לא יתקמפל.
- ה. עבור כל שורה שנוסיף כשורה ראשונה לבנאי – הקוד יתקמפל.
- ו. עבור כל שורה שנוסיף כשורה ראשונה לבנאי – הקוד לא יתקמפל.

```
public class MyClass {  
  
    public static void func(){  
        List<String> lst = Arrays.asList("python","java",null,"c#");  
        for (String i : lst){  
            try{  
                System.out.print(i.length());  
            }  
            finally{  
                System.out.print("#");  
            }  
        }  
    }  
  
    public static void main(String[] args){  
        func();  
    }  
}
```

מה יקרה בהרצת הקוד הבא?

- א. התוכנית תדפיס 6#4# ותעוף על שגיאה בזמן ריצה.
- ב. התוכנית תדפיס 6#4# ותסתיים ללא שגיאה.
- ג. התוכנית תדפיס 6#4## ותעוף על שגיאה בזמן ריצה.
- ד. התוכנית תדפיס 6#4## ותסתיים ללא שגיאה.
- ה. התוכנית תדפיס 6#4##2# ותסתיים ללא שגיאה.
- ו. התוכנית תדפיס 64# ותעוף על שגיאה בזמן ריצה.
- ז. התוכנית תדפיס 64 ותעוף על שגיאה בזמן ריצה.
- ח. התוכנית תדפיס 64#3 ותסתיים ללא שגיאה.

4. שאלה 4:

שאלות 4-6 יתייחסו לקוד הבא, שבו חסר מימוש במחלקה B. בכל שאלה, נחליף את הכוכביות (/* */) במימוש כלשהו. הניחו כי שתי המחלקות נמצאות באותה החבילה.

```
public class A{
    public Object func(String str){
        System.out.println(str);
        return "d";
    }
}
public class B extends A{
    /* */

    public static void main(String[] args){
        A a = new B();
        a.func("a");
    }
}
```

נחליף את ה /* */ במימוש הבא:

```
public String func(String str){
    return (String)(super.func(str + "A"));
}
```

מה יקרה בהרצת המחלקה B?

- א. התוכנית לא תרוץ כיון שמחלקה B אינה מתקמפלת.
- ב. התוכנית תרוץ אך תיזרק שגיאת זמן ריצה.
- ג. התוכנית תדפיס "a"
- ד. התוכנית תדפיס "aA"
- ה. התוכנית תדפיס "aAd"
- ו. התוכנית תדפיס "d".
- ז. התוכנית תיכנס ללולאה אינסופית.

5. שאלה 5:

נחליף את ה /* */ במימוש הבא:

```
public Object func(String str) throws Exception{
    return super.func(str + "A");
}
```

מה יקרה בהרצת המחלקה B?

- א. התוכנית לא תרוץ כיון שמחלקה B אינה מתקמפלת.
- ב. התוכנית תרוץ אך תיזרק שגיאת זמן ריצה.
- ג. התוכנית תדפיס "a"
- ד. התוכנית תדפיס "aA"
- ה. התוכנית תדפיס "aAd"
- ו. התוכנית תדפיס "d".
- ז. התוכנית תיכנס ללולאה אינסופית.

6. שאלה 6:

נחליף את ה /*****/ במימוש הבא:

```
public Object func(Object str){  
    return func(str + "A");  
}
```

מה יקרה בהרצת המחלקה B?

- א. התוכנית לא תרוץ כיון שמחלקה B אינה מתקמפלת.
- ב. התוכנית תרוץ אך תיזרק שגיאת זמן ריצה.
- ג. התוכנית תדפיס "a"
- ד. התוכנית תדפיס "aA"
- ה. התוכנית תדפיס "aAd"
- ו. התוכנית תדפיס "d".
- ז. התוכנית תיכנס ללולאה אינסופית.

7. שאלה 7:

```
public class NaturalNumbers implements Supplier<Integer> {  
    private int i = 0;  
  
    public Integer get() {  
        return ++i;  
    }  
  
    public static void func(int i){  
        System.out.print(i);  
    }  
  
    public static void main(String[] args) {  
        Stream<Integer> s = Stream.generate(new NaturalNumbers());  
        s.limit(2).peek(i->func(i*i)).forEach(i->func(i)); //1  
        // s.limit(2).peek(i->func(i)); //2  
    }  
}
```

בשאלה זו נרצה לדעת מה ידפיס הקוד כפי שהוא, ומה יקרה אם נחליף את שורה //1 בשורה //2 שנמצאת בהערה.

- א. יודפס 12, אם נחליף את שורה //1 בשורה //2 יודפס 14.
- ב. יודפס 14, אם נחליף את שורה //1 בשורה //2 יודפס 14.
- ג. יודפס 1142, אם נחליף את שורה //1 בשורה //2 יודפס 14.
- ד. לא יודפס שום דבר בשני המקרים.
- ה. יודפס 12, אם נחליף את שורה //1 בשורה //2 לא יודפס כלום.
- ו. יודפס 14, אם נחליף את שורה //1 בשורה //2 לא יודפס כלום.
- ז. יודפס 1142, אם נחליף את שורה //1 בשורה //2 לא יודפס כלום.

נתונה המחלקה Gen ובה לא נתון המימוש של המתודה func.

```
public class Gen<T, S>{
    public Gen(){
    }

    /***** missing code of the method func *****/

    public static void main(String[] args){
        Gen<Object, String> gen = new Gen<>();
        List<String> strList = new ArrayList<String>();
        gen.func(strList);
    }
}
```

לפניכם שלוש חתימות אפשריות של המתודה func. עבור אילו מבין החתימות הבאות של המתודה func הקוד של המחלקה Gen יתקמפל?

חתימה 1: `public void func(List<Object> lst)`

חתימה 2: `public void func(List<S> lst)`

חתימה 3: `public void func(List<? extends T> lst)`

- א. חתימה 1 בלבד.
- ב. חתימה 2 בלבד.
- ג. חתימה 3 בלבד.
- ד. חתימות 1 או 2 בלבד.
- ה. חתימות 2 או 3 בלבד.
- ו. חתימות 1 או 3 בלבד.
- ז. לכל אחת מבין החתימות, הקוד יתקמפל לאחר הוספת המתודה.
- ח. לכל אחת מבין החתימות, הקוד לא יתקמפל לאחר הוספת המתודה.

9. שאלה 9:

```
public class Outer{
    private int i = 0;

    public class Inner{
        private int i = 5; //1

        public int getInnerI(){
            return this.i;    //2
        }

        public int getOuterI(){
            Outer out = Outer.this;
            return out.i;    //3
        }
    }

    public static void main(String[] args){
        Inner inner = new Inner(); //4
    }
}
```

תזכורת: נשתמש ב Outer.this מתוך מחלקה פנימית בשביל לגשת לאובייקט של המחלקה העוטפת, במידה וקיים כזה.

בחר/י את התשובה הטובה ביותר:

- א. ישנה שגיאת קומפילציה בשורה המסומנת ב //1 בלבד.
- ב. ישנה שגיאת קומפילציה בשורה המסומנת ב //2 בלבד.
- ג. ישנה שגיאת קומפילציה בשורה המסומנת ב //3 בלבד.
- ד. ישנה שגיאת קומפילציה בשורה המסומנת ב //4 בלבד.
- ה. הקוד מתקמפל.
- ו. ישנה יותר משגיאת קומפילציה אחת בקוד.

10. שאלה 10:

להלן מספר טענות הקשורות לשדות סטטיים. בחר/י את הטענה שאינה נכונה.

- א. אם שדה מוגדר כסטטי, קיים מופע יחיד שלו בכל התוכנית.
- ב. ניתן להגדיר שדה סטטי כ private.
- ג. ניתן לגשת לשדה סטטי מתוך מתודת מופע של המחלקה.
- ד. ניתן לבצע השמה לשדה סטטי פעם אחת בלבד.
- ה. ניתן להשתמש בשדה סטטי של מחלקה X כלשהי גם אם לא נוצר במהלך ריצת התוכנית שום אובייקט מטיפוס דינאמי X.


```
public class A{  
    public static void main(String[] args){  
        B b = new B();  
        for (String s : b){  
            System.out.println(s);  
        }  
    }  
}
```

המחלקה A משתמשת במחלקה B שהקוד שלה לא נתון בבחינה. להלן מספר אופציות להצהרה על המחלקה B.

אופציה 1: `class B implements Map<String, String>`

אופציה 2: `class B implements Iterable<String>`

אופציה 3: `class B implements Iterator<String>`

אילו מבין ההצהרות הבאות מאפשרות לקוד של A להתקמפל (הניחו כי המימוש של מחלקה B בכל אופציה הוא תקין ומחלקה B מתקמפלת).

- א. אופציה 1 בלבד.
- ב. אופציה 2 בלבד.
- ג. אופציה 3 בלבד.
- ד. אופציות 1+2 בלבד.
- ה. אופציות 1+3 בלבד.
- ו. אופציות 2+3 בלבד.
- ז. כל אחת מבין האופציות תאפשר לקוד של A להתקמפל.
- ח. כל אחת מבין האופציות לא תאפשר לקוד של A להתקמפל.

שאלות 12 ו 13 מתייחסות לקטע קוד הבא. בשאלה זו התעלמו מהכוכביות (/***/).

```
public class A {
    public int i;

    public A(int i) {
        this.i = i;
    }

    /***** Option 1 *****/
    @Override
    public int hashCode() {
        return i;
    }

    @Override
    public boolean equals(Object obj) {
        return true;
    }
    /***** End Option 1 *****/

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

מה יודפס בהרצת התוכנית?

- א. 2true
- ב. 1true
- ג. 2false
- ד. 1false
- ה. לא ניתן לדעת (תלוי במימוש של HashSet)

13. שאלה 13:

בשאלה זו, נחליף את הקוד שנמצא בשאלה הקודמת בין הכוכביות בקוד הבא:

```
/****** Option 2 *****/
@Override
public int hashCode() {
    return 1;
}

@Override
public boolean equals(Object obj) {
    return false;
}
/****** End Option 2 *****/
```

מה יודפס כעת בהרצת התוכנית?

- א. 2true
- ב. 1true
- ג. 2false
- ד. 1false
- ה. לא ניתן לדעת (תלוי במימוש של HashSet)

14. שאלה 14:

```
public abstract class A{
    public void f1(String str){
        System.out.println(str);
    }

    public static void main(String[] args){
        A a = new A(){
            public void f2(String str){
                //f1(str);    /**
            }
        };
        //a.f2("abc");    /**
    }
}
```

מהי הטענה הנכונה מבין הטענות הבאות?

- א. הקוד מתקמפל כפי שהוא. אם נוציא את השורה האחרונה (מסומנת ב **) מהערה, הקוד לא יתקמפל.
- ב. הטיפוס הדינאמי של a הוא A.
- ג. הקוד מתקמפל כפי שהוא. אם נוציא את השורה המסומנת ב * מהערה, הקוד לא יתקמפל.
- ד. הקוד אינו מתקמפל כפי שהוא.
- ה. כל הטענות, מלבד טענה זו, אינן נכונות.
- ו. מלבד טענה זו, ישנה יותר מטענה אחת נכונה.

```
public class C extends A {
    public C(int i) {
        super(i);
    }

    public void inc() {
        this.i++; // *
        foo();
    }

    public static void foo() {
        System.out.print("C");
    }
}

public class A {
    public int i;

    public A(int i) {
        this.i = i;
    }

    public void inc() {
        foo();
    }

    public static void foo() {
        System.out.print("A");
    }

    public static void main(String[] args) {
        C c = new C(1);
        c.inc();
        A ac = new C(1);
        ac.inc();
        System.out.print(c.i);
        System.out.print(ac.i);
    }
}
```

מה יודפס בהרצת התוכנית A?

- א. הקוד לא מתקמפל ולכן התוכנית לא תרוץ - שגיאת קומפילציה בשורה המסומנת ב-*
- ב. הקוד מתקמפל ומדפיס CC22
- ג. הקוד מתקמפל ומדפיס CA22
- ד. הקוד מתקמפל ומדפיס CA21
- ה. הקוד מתקמפל ומדפיס AA22
- ו. הקוד מתקמפל ומדפיס CA11
- ז. הקוד מתקמפל ומדפיס CC21

שאלה זו מתייחסת לגרסה 7 של ג'אווה

נתונות מחלקות A, B ו-C כלשהן, ומנשקים I ו-F.

מחלקה A הינה מחלקה אבסטרקטית ומממשת את מנשק I.

מחלקה C יורשת ממחלקה A, ומחלקה B יורשת ממחלקה A ומממשת את מנשק F.

ניתן להניח כי כל המחלקות והמנשקים הנ"ל אינם יורשים או מממשים אף מנשק או מחלקה מלבד אלה המצויינים בשאלה.

טענה 1: תמיד ניתן להמיר את מנשק I למחלקה אבסטרקטית - השינוי הנוסף היחידי הוא במחלקה A אשר תירש מהמחלקה האבסטרקטית I.

טענה 2: תמיד ניתן להמיר את מנשק F למחלקה אבסטרקטית - השינוי הנוסף היחידי הוא במחלקה B אשר תירש מהמחלקה האבסטרקטית F.

טענה 3: תמיד ניתן להמיר את מחלקה A למנשק - השינוי הנוסף היחידי הוא במחלקות B ו-C אשר יממשו את מנשק A במקום לרשת מהמחלקה האבסטרקטית A.

בחרי' את התשובה הטובה ביותר:

- א. אף טענה אינה נכונה
- ב. טענה 1 בלבד נכונה
- ג. טענות 1 ו-3 נכונות בלבד
- ד. טענות 1 ו-2 נכונות בלבד
- ה. טענות 2 ו-3 נכונות בלבד
- ו. כל הטענות נכונות

```
public interface H {
    public void f(int i) throws IOException;
}

public class A implements H { // *
    @Override
    public void f(int i) {
        System.out.print("A");
    }

    public static void main(String[] args) {
        H h = new A();
        h.f(1); // **
    }
}
```

בחר/י את התשובה הנכונה ביותר מבין התשובות הבאות:

- א. הקוד מתקמפל ומדפיס A
- ב. הקוד לא מתקמפל - שגיאת קומפילציה בשורה המסומנת ב-*, אך אם נשנה את הטיפוס הסטטי של h ל-A הקוד יתקמפל וידפיס A
- ג. הקוד לא מתקמפל – שגיאת קומפילציה בשורה המסומנת ב-*, גם אם נשנה את הטיפוס הסטטי של h ל-A
- ד. הקוד לא מתקמפל - שגיאת קומפילציה בשורה המסומנת ב-**, אך אם נשנה את הטיפוס הסטטי של h ל-A הקוד יתקמפל וידפיס A
- ה. הקוד לא מתקמפל - שגיאת קומפילציה בשורה המסומנת ב-**, גם אם נשנה את הטיפוס הסטטי של h ל-A הקוד לא יתקמפל
- ו. הקוד מתקמפל אך נזרקת שגיאת זמן ריצה מסוג IOException

```
public class A {
    public float foo(float a, float b) throws IOException {
        return a + b;
    }
}

public class B extends A {
    /* float foo(float a, float b) {           //1
        return super.foo(a, b);
    } */

    /* public int foo(int a, int b) throws Exception { //2
        return (int)super.foo(a, b);
    } */

    /* public float foo(float a, float b) throws Exception { //3
        return super.foo(a, b);
    } */

    /* public float foo(float a, float b) { //4
        return super.foo(a, b);
    } */
}
```

אילו מבין ארבע הפונקציות שמימושיהן מופיעים בהערה ניתן להוסיף למחלקה B כך שהקוד יתקמפל?

- א. פונקציה 1 בלבד
- ב. פונקציה 2 בלבד
- ג. פונקציה 3 בלבד
- ד. פונקציה 4 בלבד
- ה. פונקציה 2 ו-4 (כל אחת בנפרד ושתייהן ביחד)
- ו. פונקציה 2 ו-3 (כל אחת בנפרד ושתייהן יחד)

```
public class Gui {
    public static void main(String[] args) {
        Display display = new Display();
        Shell shell = new Shell(display);
        shell.setLayout(new GridLayout());
        Button button = new Button(shell, SWT.NONE);
        button.setText("Click and check the console");
        button.addSelectionListener(new I());
        button.addSelectionListener(new I()); // *
        shell.open();
        while (!shell.isDisposed()) {
            if (!display.readAndDispatch()) {
                display.sleep();
            }
        }
    }
}

public class I implements SelectionListener {
    public static int i = 0;

    @Override
    public void widgetDefaultSelected(SelectionEvent arg0) {
    }

    @Override
    public void widgetSelected(SelectionEvent arg0) {
        i++; // **
        System.out.print(i);
    }
}
}
```

בחר/י את התשובה הטובה ביותר המתייחסת לקוד זה:

- א. הקוד אינו מתקמפל - שגיאת קומפילציה בשורה המסומנת ב-*
- ב. הקוד אינו מתקמפל - שגיאת קומפילציה בשורה המסומנת ב-**
- ג. הקוד מתקמפל ורץ - במידה והמשתמש ילחץ על הכפתור פעמיים יודפס: 1122
- ד. הקוד מתקמפל ורץ - במידה והמשתמש ילחץ על הכפתור פעמיים יודפס: 1234
- ה. הקוד מתקמפל ורץ - במידה והמשתמש ילחץ על הכפתור פעמיים יודפס: 1111
- ו. הקוד מתקמפל ורץ - במידה והמשתמש ילחץ על הכפתור פעמיים יודפס: 1212


```
public class A {
    public String a = new String("A");

    public A() {
        this("AA"); // ***
        System.out.print(a);
    }

    public A(String aa) {
        a = aa;
    }
}

public class C extends A {
    public String a = new String("C"); // *

    public C() {
        super();
        System.out.print(a); // **
    }

    public static void main(String[] args) {
        C c = new C();
        A ca = new C();
        A aa = new A();
    }
}
```

בחר/י את התשובה הטובה ביותר המתייחסת לקוד זה.

- א. הקוד לא מתקמפל - שגיאת קומפילציה בשורה המסומנת ב-*
- ב. הקוד לא מתקמפל - שגיאת קומפילציה בשורה המסומנת ב-**
- ג. הקוד לא מתקמפל - שגיאת קומפילציה בשורה המסומנת ב-***
- ד. הקוד מתקמפל ומדפיס AACACAA
- ה. הקוד מתקמפל ומדפיס AACAAAAA
- ו. הקוד מתקמפל ומדפיס ACACA
- ז. הקוד מתקמפל ומדפיס AAAAAACAA

```
public class Singleton {
    public static Singleton instance = null;

    private Singleton() {
    }

    public static Singleton getInstance() {
        if (instance == null) {
            instance = new Singleton();
        }
        return instance;
    }
}

public class Test {
    public static void main(String[] args) {
        Singleton s1 = Singleton.getInstance();
        Singleton s2 = Singleton.getInstance();
        System.out.println(s1 == s2);
        System.out.println(s1.equals(s2));
        System.out.println(s1.instance == s2.instance);
    }
}
```

בחר/י את התשובה הטובה ביותר:

- א. הקוד מדפיס true true true
- ב. הקוד מדפיס true true false
- ג. הקוד מדפיס true false true
- ד. הקוד מדפיס false true true
- ה. הקוד מדפיס true false false
- ו. הקוד מדפיס false true false