מספר בחינה:
0

בחינה בתוכנה 1

סמסטר א' תש״ף, מועד א', 2 בפברואר 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 לגירסא 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.

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

יש לנמק בתמצות את כל התשובות על גבי טופס הבחינה במקומות המסומנים. נימוק חסר או לא נכון עלול לגרום לאי קבלת נקודות על שאלה במקרים שבהם יוחלט לקבל יותר מתשובה אחת. נימוק לתשובה אחרת מזו שסומנה על גבי טופס התשובות לא יתקבל – התשובה נקבעת אך ורק על פי התשובה המסומנת בטופס התשובות.

יש לצרף את טופס המבחן למחברת הבחינה. מחברת ללא טופס עזר תפסל. סימוני תשובות 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 מחשבונים 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.

בהצלחה!

 מספר סידורי: _____ מספר ת"ז:_____

מספר בחינה:
0

בשאלות 1-8 נממש את המחלקה LastAccessList אשר מהווה מבנה נתונים גנרי שהוא List של אובייקטים מטיפוס כלשהו. עבור כל איבר ב LastAccessSet נרצה לשמור חותמת זמן שבה נעשתה הגישה האחרונה לאותו איבר, בשליפה או בשמירה מתוך מבנה הנתונים. את חותמת הזמן ייצג הטיפוס Date שקיים ב Java ואין צורך לממשו.

את המימוש נעשה באמצעות ירושה והכלה. נירש מ ArrayList ונחזיק שדה מטיפוס HashMap עבור שמירת חותמות זמן.

הניחו את קיומו של השירות SleepServices.*sleep*() אשר משהה את ריצת התוכנית ב 10 שניות.
עליכם לקרוא את כל השאלות על מנת להבין את הדרישות מהמחלקה. במענה על כל סעיף, הניחו כי הסעיפים הקודמים ממומשים נכון על פי החוזים ודרישות התרגיל.

fixed during the exam: In the Tester class, replace b with a
In the lines marked with *.
In addition,
assume a==aaa
and b=bbb

להלן דוגמת שימוש במחלקה:

```
public class Tester{
    public static void main(String[] args) {
        LastAccessList<String> lst = new LastAccessList<>();
        lst.add("a");
        SleepServices.sleep();
        lst.add("b");
        SleepServices.sleep();
        System.out.println("(1) a " + lst.getTimestampForItem("a"));
        System.out.println("(2) b " + lst.getTimestampForItem("b"));
        lst.get(0);
        SleepServices.sleep();
       *System.out.println("(3) b " + lst.getTimestampForItem("b"));
        System.out.println("(4) b " + lst.getTimestampForItem("b"));
       *lst.add("b");
       *System.out.println("(5) b " + lst.getTimestampForItem("b"));
        System.out.println("(6) b " + lst.getTimestampForItem("b"));
        SleepServices.sleep();
        System.out.println("(7) " + lst.size());
        SleepServices.sleep();
       *System.out.println("(8) b " + lst.getTimestampForItem("b"));
        System.out.println("(9) b " + lst.getTimestampForItem("b"));
    }
}
```

פלט התוכנית המבוקש עבור הרצת התוכנית בתאריך 17/01/2020 14:18:15 הוא:

```
(1) aaa : 17/01/2020 14:18:15
(2) bbb : 17/01/2020 14:18:25
(3) aaa : 17/01/2020 14:18:35
(4) bbb : 17/01/2020 14:18:25
(5) aaa : 17/01/2020 14:18:45
(6) bbb : 17/01/2020 14:18:25
(7) 3
(8) aaa : 17/01/2020 14:18:45
(9) bbb : 17/01/2020 14:18:25
```


מספר סידורי: _____ מספר ת"ז:_____

זהו שלד המחלקה אותה נממש:

```
@ imp_inv: size() >= lastAccessMap.size()
public class LastAccessList   /*** Q1 ***/ {
    private /*** Q1 ***/ lastAccessMap;

    private Date getCurrentDate() {
        return new Date();
    }

    private void setTimestampForItem(T item) {
        lastAccessMap.put(item, getCurrentDate());
    }

    public LastAccessList() {
        this.lastAccessMap = new HashMap();
    }

    /** @post !contains(item) implies $ret == null */
    public Date getTimestampForItem(T item) {
        return lastAccessMap.get(item);
    }

    @Override
    /** @post: $ret == true implies getTimestampForItem(o) == current date*/
    public boolean contains(Object o) { /*** Q2 ***/ }

    @Override
    /** @post: getTimestampForItem(o) == current date, where o is the object in the
index place */
    public T get(int index) {
        // implementation is not provided, assume it's correct
    }

    @Override
    /** @post: size() == prev(size) + 1
        @post: getTimestampForItem(e) == current date */
    public boolean add(T e) {   /*** Q3 ***/   }


    @Override
    public int size() { /*** Q4 ***/ }


}
```

 מספר סידורי: _____ מספר ת"ז:_____

1. <u>שאלה 1:</u>

מבין האופציות הבאות, השלימו את הגדרת המחלקה LastAccessList והגדרת השדה שלה.

```
public class LastAccessList<T> extends ArrayList<T>{    //op 1
   private Map<T, Date> lastAccessMap;

}

public class LastAccessList<T, Date> extends ArrayList<T>{  //op 2
   private Map<T, Date> lastAccessMap;

}

public class LastAccessList<T, S> extends ArrayList<T>{ //op 3
   private Map<T, S> lastAccessMap;

}

public class LastAccessList<T> extends ArrayList<T>{ //op 4
   private Map<T, S> lastAccessMap;

}

public class LastAccessList extends ArrayList<T>{ //op 5
   private Map<T, Date> lastAccessMap;

}

public class LastAccessList extends ArrayList<T, Date>{ //op 6
   private Map<T, Date> lastAccessMap;

}

public class LastAccessList extends ArrayList<T, S>{ //op 7
   private Map<T, S> lastAccessMap;

}

public class LastAccessList extends ArrayList<T>{ //op 8
   private Map<T, S> lastAccessMap;

}
```

בחרי בתשובה הטובה ביותר: איזו מבין האופציות מתאימה למימוש המחלקה LastAccessList?

א. אופציה 1
ב. אופציה 2
ג. אופציה 3
ד. אופציה 4
ה. אופציה 5
ו. אופציה 6
ז. אופציה 7
ח. אופציה 8

 מספר סידורי: _____ מספר ת"ז:_____

נימוק:

.

2. שאלה 2:

לפניכם שני מימושים אפשריים לשירות contains.

```
public boolean contains(Object o) {   //op1
    if (!lastAccessMap.containsKey(o)) { return false; }
    setTimestampForItem((T)o);
    return true;
}

public boolean contains(Object o) {   //op2
    setTimestampForItem((T)o);
    return super.contains(o);
}
```

בחר\י בתשובה הטובה ביותר:

נימוק:

א. שני המימושים תקינים ומקיימים את חוזה המחלקה.

ב. רק המימוש הראשון תקין ומקיים את חוזה המחלקה.

ג. רק המימוש השני תקין ומקיים את חוזה המחלקה.

ד. שני המימושים לא תקינים.

3. <u>שאלה 3:</u>

מבין שורות הקוד הבאות, הרכיבו את מימוש השירות add:

```
/* 1 */    if (! lastAccessMap.containsKey(e)) {
/* 2 */    if (! this.contains(e)) {
/* 3 */    setTimestampForItem(e);
/* 4 */    return super.add(e);
/* 5 */    return this.add(e);
/* 6 */    }
```

אם לדוגמא, בחרתם לממש את השירות add באופן הבא :

```
/* 1 */    if (! lastAccessMap.containsKey(e)) {
/* 2 */    if (! this.contains(e)) {
```

עליכם לבחור ברצף השורות 1,2 (כלומר, המספר הכי שמאלי שייך לשורה הראשונה במימוש).
בחר\י בתשובה הטובה ביותר:

נימוק:

א. הרצף הנכון הוא 1,3,6,4
ב. הרצף הנכון הוא 1,3,6,5
ג. הרצף הנכון הוא 3,4
ד. הרצף הנכון הוא 3,5
ה. הרצף הנכון הוא 1,3,4,6
ו. הרצף הנכון הוא 2,3,4,5
ז. הרצף הנכון הוא 2,3,4,6
ח. הרצף הנכון הוא 2,3,4,5

4. <u>שאלה 4:</u>

במחלקה Tester שמובאת בתחילת הבחינה, הדפסה (7) מבצעת קריאה ל ()size שאמורה להדפיס את הערך 3. בשאלה זו, הניחו כי add ממומשת נכון על פי החוזה שלה.

בחר\י בתשובה הטובה יותר:

נימוק:

א. אם לא יהיה שום מימוש של ()size במחלקה LastAccessList הקוד של Tester לא יתקמפל.

ב. אם לא יהיה שום מימוש של ()size במחלקה LastAccesList הקוד של Tester יתקמפל, אך יזרוק שגיאת זמן ריצה בזמן הקריאה ל ()size.

ג. אם לא יהיה שום מימוש של ()size במחלקה LastAccessList הקוד של Tester יתקמפל, ירוץ, אך לא ידפיס 3. אם נממש את ()size ע"י החזרת ()lastAccessMap.size הקריאה ל size כן תדפיס 3.

ד. אם לא יהיה שום מימוש של ()size במחלקה LastAccessList הקוד של Tester יתקמפל, ירוץ וידפיס 3.

ה. אם לא יהיה שום מימוש של ()size במחלקה LastAccessList הקוד של Tester יתקמפל, ירוץ, אך לא ידפיס 3. אם נממש את ()size ע"י החזרת ()super.size הקריאה ל size כן תדפיס 3.

5. <u>שאלה 5:</u>

התבוננו בתוכנית הבאה:

```
List<String> anotherList = new LastAccessList<>();
anotherList.add("a"); //$
SleepServices.sleep();
anotherList.add("b");
SleepServices.sleep();
LastAccessList<String> castList = (LastAccessList<String>) anotherList; //*
System.out.println("(1) a " + castList.getTimestampForItem("a")); //#
System.out.println("(2) b " + castList.getTimestampForItem("b"));
```

הניחות כי LastAccessList ממומשת באופן תקין לפי הדרישות ולפי החוזים, והתוכנית Tester אשר מובאת בתחילת הבחינה רצה כנדרש. בחר\י בתשובה הטובה ביותר:

נימוק:

א. הקוד מתקמפל ורץ, ועובד בדיוק כמו שהיה עובד אם anotherList היתה מוגדרת מלכתחילה בתור `LastAccessList<String>`

ב. הקוד מתקמפל ורץ, אך הדפסות (1) ו (2) ידפיסו null -ים במקום חתימות זמן.

ג. הקוד לא מתקמפל בגלל השורה המסומנת ב *

ד. הקוד מתקמפל אך יזרוק שגיאת זמן ריצה בשורה המסומנת ב *

ה. הקוד מתקמפל אך יזרוק שגיאת זמן ריצה בשורה המסומנת ב #

ו. הקוד מקמפל, אז יזרוק שגיאת זמן ריצה בשורה המסומנת ב $

6. <u>שאלה 6:</u>

הסטודנטית ברית מציעה לממש את LastAccessList הפוך מהמימוש המוצע – המחלקה תירש מ HashMap ותכיל שדה מטיפוס ArrayList. כמובן שלאור השינוי בירושה ובהכלה, המימוש החדש יהיה מותאם לעיצוב החדש, ויהיה שונה מהמימוש המקורי. בפרט, כמובן שחתימות השירותים שהוא חושף יכולות להשתנות.
לפניכם שלוש טענות הקשורות להצעת המימוש החדשה:

טענה 1: LastAccessList לא אמורה לקיים יחס is-a עם המחלקה HashMap, ולכן לא נכון לרשת ממנה.
טענה 2: הצעת המימוש של ברית לא מאפשרת שימוש חוזר במימוש של ArrayList, ולכן יהיה שכפול קוד של הלוגיקה הממומשת ב ArrayList.
טענה 3: לא ניתן לממש את ההצעה של ברית כך שהקוד של ה Tester יתקמפל.

נימוק:

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

7. <u>שאלה 7:</u>

נוסיף למחלקה LastAccessList שירות נוסף ומחלקה פנימית.

```
public class LastAccessList   /*** Q1 ***/

    // previous code here

    public class LastAccessIterator implements Iterator<T>{
        Iterator<T> it;

        public LastAccessIterator() {
             it = lastAccessMap.keySet().stream()
                    .sorted((x,y) ->
                       lastAccessMap.get(x).compareTo(lastAccessMap.get(y)))
                   .collect(Collectors.toList())
                   .iterator();
        }

        @Override
        public boolean hasNext() { return it.hasNext(); }

        @Override
        public T next() { return it.next(); }
    }

    public Iterator<T> getAccessOrderIterator(){
        return new LastAccessIterator();
    }
}
```

הניחו כי המחלקה Date מממשת את המנשק Comparable וכי מיון רשימת איברים מטיפוס date יחזיר את התאריך המוקדם ביותר בתוך האיבר הראשון.
מה יודפס בריצת התוכנית הבאה?

```
public static void main(String[] args) {
    LastAccessList<String> lst = new LastAccessList<>();
    lst.add("aaa");
    SleepServices.sleep(10);
    lst.add("bbb");
    SleepServices.sleep(10);
    lst.add("bbb");
    Iterator<String> it = lst.getAccessOrderIterator();
    for (; it.hasNext(); ) {
        System.out.print(it.next() + " ");
    }
}
```

בחר\י בתשובה הטובה ביותר:

נימוק:

א. aaa bbb bbb
ב. aaa bbb
ג. bbb bbb aaa
ד. bbb aaa
ה. יודפסו 3 תאריכים, התאריך הראשון שיודפס יהיה המוקדם מביניהם.
ו. יודפסו 3 תאריכים, התאריך הראשון שיודפס יהיה המאוחר מביניהם.
ז. יודפסו 2 תאריכים, התאריך הראשון שיודפס יהיה המוקדם מביניהם.
ח. יודפסו 2 תאריכים, התאריך הראשון שיודפס יהיה המאוחר מביניהם.

8. שאלה 8:

הסטודנטית ברית החליטה שהיא מעוניינת לממש גירסא חדשה של LastAccessList שלא תספק תמיכה בשירות get. לצורך כך, היא מעוניינת לממש מחלקה חדשה שתירש מ LastAccessList שבה ידרס המימוש של get עם מימוש שזורק חריג. ברית מתלבטת בין שלושה מימושים אפשריים:

```
@Override public T get(int index) {                    // v-1
    throw new Exception("Unsupported Operation");
}

@Override public T get(int index) throws Exception{   // v-2
    throw new Exception("Unsupported Operation");
}

@Override public T get(int index) {                    // v-3
    throw new RuntimeException("Unsupported Operation");
}
```

באילו מימושים ניתן לדרוס את המימוש הקיים של get כך שהקוד של המחלקה הדורסת יתקמפל? הניחו כי שאר האלמנטים הקשורים לירושה (כמו הגדרת בנאי וכו') מתקמפלים:

נימוק:

א. רק v-1.
ב. רק v-2
ג. רק v-3
ד. רק v-1 ו v-2
ה. רק v-1 ו v-3
ו. רק v-2 ו v-3
ז. בחירת כל אחת מבין הגירסאות v-1, v-2 ו v-3 תאפשר לקוד להתמפל.
ח. הקוד לא יתקמפל עבור כל אחת מבין הגירסאות v-1, v-2 ו v-3.

 מספר סידורי: _____ מספר ת"ז:_____

9. שאלה 9:

הסטודנטית ברית מעוניינת לממש מחלקה A כלשהי כך שתהיה immutable. A יורשת מ-Object, מכילה שדות מופע בלבד, ולא מוגדרות בתוכה מחלקות פנימיות. בנוסף, המחלקה A מוגדרת להיות final.

לפניכם מספר טענות על אופן המימוש של המחלקה A.

טענה 1: אם כל שדות המופע של A הם final, אז A היא immutable.

טענה 2: אם A היא immutable, כל שדות המופע שלה הם בהכרח final.

טענה 3: אם כל מתודות המופע ושדות המופע שמוגדרים ב-A הם פרטיים, אז A היא בהכרח immutable.

בחרו את התשובה הנכונה ביותר:

נימוק:

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

10. שאלה 10:

לפניכם שלוש טענות הקשורות לקוד המשתמש בפרמטר גנרי:

טענה 1: אם T הוא פרמטר גנרי של מחלקה כלשהי, למחלקה זו יכול להיות שדה סטטי מטיפוס T

טענה 2: מחלקה גנרית יכולה לרשת ממחלקה לא גנרית, ומחלקה לא גנרית יכולה לרשת ממחלקה גנרית.

טענה 3: ההשמה בשורה השניה בקוד המצורף תתקמפל תמיד, ללא תלות במה שיכתב במקום הכוכביות.

```
Set<******> genSet = ********;
Set<?> jokerSet = genSet;
```

בחר/י את התשובה הנכונה ביותר:

נימוק:

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

11. 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 boolean equals(Object obj) {  return true; }

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

להלן מספר טענות המתייחסות למימושים השונים של המחלקה A. הניחו כי:
1. הקוד הקיים של A לא ישתנה, וניתן רק להוסיף קוד.
2. הקוד של A הוא דטרמיניסטי. כלומר, עבור אותו הקלט מחזירות את אותה התשובה בכל הרצה.
2. גודל טבלת ה hash גדול מ 4.

first 2. corrected to:
Deterministic - each execution of the code gets the same output

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

בחרו את התשובה הטובה ביותר:

נימוק:

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. קיימות שתי טענות לא נכונות.
ו. קיימות שלוש טענות לא נכונות.
ז. כל הטענות נכונות.

12. שאלה 12:

```
public class Car {
    private int year;
    private Engine engine  = new Engine(this);

    public Car(int year) {  this.year = year;  }

    public class Engine{
        public Car car;

        public Engine(Car car) {  this.car = car;  }

        public int f() {  return year;  }
    }

    public static void main(String[] args) {
        Car c1 = new Car(1960);
        Car c2 = new Car(1970);
        c2.engine.car = c1.engine.car; // *
        System.out.println(c2.engine.f()); // **
    }
}
```

מה יקרה בהרצת התוכנית הבא? בחר/י בתשובה הטובה ביותר:

נימוק:

א. יש שגיאת קומפילציה בשורה *.
ב. יש שגיאת קומפילציה בשורה **.
ג. תיזרק שגיאה בשורה ** ולא יודפס כלום.
ד. ריצת התוכנית תסתיים בהצלחה ויודפס 1960
ה. ריצת התוכנית תסתיים בהצלחה ויודפס 1970

13. שאלה 13:

```
public class A {
    public int i = 1;
    public void foo() {   System.out.println(i);  }
}
public class B extends A {
    private int i = 3;   //*

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

מה יקרה בהרצת התוכנית הבאה? בחר/י בתשובה הטובה ביותר:

נימוק:

א. התוכנית מדפיסה 11
ב. התוכנית מדפיסה 13
ג. התוכנית מדפיסה 31
ד. התוכנית מדפיסה 33
ה. קיימת שגיאת קומפילציה בשורה המסומנת ב *
ו. התוכנית מדפיסה 1 ועפה על חריג
ז. התוכנית מדפיסה 3 ועפה על חריג.
ח. קיימת שגיאת קומפילציה באחת מהבין השורות המסומנות ב ** וב ***

14. שאלה 14:

```
public class A {
    protected String s = "A";

    public void f() {  System.out.print(s); }

    public A() { f(); }
}

public class B extends A {
    private String s = "B";

    public void f() { System.out.print(s); }

    public B() { f();  }

    public static void main(String[] args) {
        A a = new B(); // ***
        System.out.print(a.s);
    }
}
```

מה יודפס בהרצת התוכנית B? בחר/י בתשובה הטובה ביותר.

נימוק:

א. קיימת שגיאת קומפילציה במחלקה B.
ב. התוכנית עפה על חריג בשורה ***.
ג. התוכנית מדפיסה BBA
ד. התוכנית מדפיסה nullBA
ה. התוכנית מדפיסה ABA
ו. התוכנית מדפיסה BBB
ז. התוכנית מדפיסה nullBB
ח. התוכנית מדפיסה ABB

15. שאלה 15:

```
class A{
    /**  missing contract */
    public int func(int i) { /* some implementation here */}
}

public class B extends A{
    /**
     * @pre i > 3
     * @post $ret < 10
     */
    @Override
    public int func(int i) { /* some implementation here  */}
}
```

החוזה של השירות func של המחלקה A אינו נתון. מבין האופציות המוצעות, איזה חוזה הוא חוקי ותקין על פי עקרונות הירושה? בחר/י בתשובה הטובה ביותר:

נימוק:

א. @pre i>0, @post $ret < 20
ב. @pre i> 5, @post $ret < 20
ג. @pre i>0, @post $ret < 2
ד. @pre i>5, @post $ret<2
ה. מלבד תשובה זו, יש יותר מתשובה נכונה אחת.
ו. מלבד תשובה זו, כל התשובות לא נכונות.

 מספר סידורי: _____ מספר ת"ז:_____

16. שאלה 16: Due to several mistakes, this question is canceled

```
public class MyGui {
    private static int j = 1;

    public static void main(String[] args) {
        List<Integer> iList = new ArrayList<>();
        Scanner s = new Scanner(System.in);
        while (iList.size() == 0) {
            System.out.println("Enter a positive integer: ");
            int i = s.nextInt();
            if (i > 0) { iList.add(i); }
        }
        s.close();
        Display display = Display.getDefault();
        Shell shell = new Shell(display);
        shell.setLayout(new RowLayout(SWT.VERTICAL));
        Button ok = new Button(shell, SWT.PUSH);
        ok.setText(j);
        ok.addSelectionListener(************);//missing code goes here
        shell.pack();
        shell.open();
        while (shell.isDisposed()) {
            if (!display.readAndDispatch())
                display.sleep();
        }
        display.dispose();
    }
}
```

נרצה כי בכל לחיצה על הכפתור הערך של המשתנה j יגדל ב- iList.get(0) (מספר חיובי שיוזן על ידי המשתמש ונשמר בתוך iList), וערכו החדש יופיע על הכפתור.

טענה 1: ניתן לממש התנהגות זו על ידי כתיבת מאזין (Listener) שמוגדר כמחלקה אנונימית (בקוד החסר המסומן בכוכביות) אשר מממשת את המנשק SelectionListener .
טענה 2: ניתן לממש התנהגות זו על ידי כתיבת מאזין שמוגדר כמחלקה מקוננת סטטית SelectionListener, וליצור מופע שלו בקוד החסר המסומן בכוכביות.
טענה 3: ניתן לממש התנהגות זאת על ידי כתיבת מאזין שמממש את המנשק SelectionListener ומוגדר כמחלקה חדשה בקובץ נפרד.
בחר/י בתשובה הטובה ביותר:

נימוק:

א. רק טענה 1 נכונה.
ב. רק טענה 2 נכונה.
ג. רק טענות 1 ו-2 נכונות, וימשיכות להיות נכונות אם j יוגדר להיות public.
ד. כל הטענות לא נכונות, וימשיכות להיות לא נכונות גם אם iList יוגדר להיות public.
ה. רק טענות 1 ו-2 נכונות, ואם j יוגדר כpublic אז גם טענה 3 תהיה נכונה.
ו. כל הטענות לא נכונות. אם j יוגדר להיות public אחת הטענות כן תהפוך להיות נכונה.

 מספר סידורי: _____ מספר ת"ז:_____

17. שאלה 17:

```
public class B{
    public static void func1(List<?> lst) {}

    public static <T> void func2(List<T> lst) {}

    public static void func3(List lst) {}

    public static void func4(List<Object> lst) {}
}
```

נרצה להוריד את השירות func1 מהמחלקה B. לפניכם שלוש טענות:
טענה 1: ניתן להחליף את כל הקריאות ל func1 בקריאות ל func2, והקוד ימשיך להתקמפל.
טענה 2: ניתן להחליף את כל הקריאות ל func1 בקריאות ל func3, והקוד ימשיך להתקמפל.
טענה 3: ניתן להחליף את כל הקריאות ל func1 בקריאות ל func4, והקוד ימשיך להתקמפל.
בחר/י בתשובה הטובה ביותר:

נימוק:

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

18. שאלה 18:
בחר/י בתשובה הטובה ביותר:

נימוק:

א. על מנת שנוכל לפתח תוכניות Java (כלומר, לקמפל אותן), עלינו להתקין JRE על המחשב.
ב. תוכנית Java מקומפלת ניתן להריץ על כל מחשב שעליו מותקנת מערכת ההפעלה עליה ביצענו את הקומפילציה.
ג. בהנתן קבצי class, ניתן להריץ אותם ללא התקנת java על המחשב.
ד. מלבד תשובה זו, כל התשובות לא נכונות.
ה. מלבד תשובה זו יש יותר מתשובה נכונה אחת.

19. שאלה 19:

```
public class Line{
    private Point pointA, pointB;

    @Override public Object clone() { /* some implementation here */}
   /* some methods are implemented here */
}

public class Point {
    private int x, y;
    public Point(int x, int y) {this.x = x; this.y = y}

    public int getX() {return x;}
    public int getY() {return y;}

    /* some methods are implemented here */
}
```

נרצה לממש את השירות clone עבור המחלקה Line כך שיחזיר עותק של המחלקה Line. המחלקות Line ו Point עשויות להכיל מימושי שירותים נוספים מלבד מה שנתון, אך השדות היחידים שלהן נתונים.
בחר/י בתשובה הטובה ביותר:

נימוק:

א. לא ניתן לממש את clone מבלי ש Line תממש את המנשק Cloneable.
ב. ניתן לממש את clone כך שתבצע שכפול עמוק (deep clone), רק אם Point תהיה Cloneable.
ג. ניתן לממש את clone כך שתבצע שכפול עמוק (deep clone), גם אם Point היא לא Cloneable.
ד. מלבד תשובה זו, כל התשובות לא נכונות.

20. שאלה 20:
בחר\י בתשובה הטובה ביותר:

נימוק:

א. שדות סטטיים נשמרים על ה heap.
ב. שדות סטטיים לא מקבלים ערכים דיפולטיים.
ג. שדה סטטי אינו יכול להיות final.
ד. שדה סטטי אינו יכול להיות מטיפוס פרימיטיבי.
ה. ניתן לגשת לשדה סטטי רק מתוך שירות סטטי.
ו. מלבד תשובה זו כל התשובות לא נכונות.
ז. מלבד תשובה יש לפחות שתי תשובות נכונות.

21. שאלה 21:

```
public class Bar {

    static class FirstException extends Exception {
        public FirstException(String message) { super("1" + message);}
    }

    static class SecondException extends FirstException {
        public SecondException(String message) { super("2" + message);}
    }

    static class ThirdException extends Exception {
        public ThirdException(String message) { super("3" + message);}
    }

    public static void main(String[] args) throws Exception{
        Exception e1 = new FirstException("");
        Exception e2 = new SecondException("");
        Exception e3 = new ThirdException("");
        List<Exception> l = Arrays.asList(e1,e2,e3);
        foo(l); //*
    }

    public static void foo(List<? extends Exception> lst) throws Exception{
        for (Exception ex : lst) {
            try {
                throw ex; // **
            }
            catch (FirstException e) {
                System.out.print(ex.getMessage());
            }
        }
    }
}
```

מה יקרה בהרצת התוכנית Bar? בחר/י בתשובה הטובה ביותר:

נימוק:

א. יש שגיאת קומפילציה בשורה המסומנת ב *
ב. יש שגיאת קומפילציה בשורה המסומנת ב **
ג. התוכנית תעוף על שגיאת זמן ריצה ולא תדפיס כלום.
ד. התוכנית תדפיס 123 ולאחר מכן תעוף על שגיאת זמן ריצה.
ה. התוכנית תדפיס 12 ולאחר מכן תעוף על שגיאת זמן ריצה.
ו. התוכנית תדפיס 112 ולאחר מכן תעוף על שגיאת זמן ריצה.
ז. התוכנית תדפיס 1123 ולאחר מכן תעוף על שגיאת זמן ריצה.
ח. התוכנית תסיים בהצלחה ללא שום הדפסה.

 מספר סידורי: _____ מספר ת"ז:_____