בחינה בתוכנה 1 (2157 - 0368)
ובתוכנה 1א (2004 - 0368)

ליאור וולף ואורנית דרור

סמסטר א' תשס"ז
מועד ג' , 21 בדצמבר 2007

משך הבחינה שלוש שעות.

יש לענות על כל השאלות. בשאלות שבהן יש צורך לנמק, תשובה ללא נימוק לא תזכה באף נקודה.

יש לענות על כל השאלות ב**גוף הבחינה במקום המיועד לכך**. המקום המיועד מספיק לתשובות מלאות. יש לצרף את טופס המבחן למחברת הבחינה. מחברת ללא טופס עזר תפסל. תשובות במחברת הבחינה לא תיבדקנה.

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.
בהצלחה!

לשימוש הבודקים:

1	**א**	**ב**	**ג**	**ד**	**ה**	**ו**	**ז**	
2	**א**	**ב**	**ג**					
3	**א**	**ב**	**ג**	**ד**	**ה**	**ו**		
4	**א**	**ב**						
5								
							סה"כ	

במבחן נדון במחלקות שמייצגות מספרים רציונליים, כגון $1/3$ או $99/100$.
תזכורת מתמטית קצרה: כדי לצמצם שברים צריך לחלק את המונה והמכנה בגורם המשותף הגדול ביותר (GCD באנגלית). למשל, הגורם המשותף הגדול ביותר של 42 ו-56 הוא 14, ולכן

$$\frac{42}{56} = \frac{3 \cdot 14}{4 \cdot 14} = \frac{3}{4}.$$

כדי לחבר או לחסר שברים, צריך למצוא את המכנה המשותף (LCM באנגלית). למשל, המכנה המשותף של 21 ו-6 הוא 42, ולכן

$$\frac{2}{21} + \frac{1}{6} = \frac{4}{42} + \frac{7}{42} = \frac{11}{42},$$

שאלה 1, 25 נקודות

א. בין המתכנתים ביל ואומה התפתח ויכוח האם **טרם** לכתיבת המחלקה המייצגת מספרים רציונליים יש לכתוב **מנשק** המתאר את הפונקציונליות של המחלקה. ביל טען כי המחלקה פשוטה דיה וכי אין הצדקה לתאר את אותו הדבר גם ע"י מנשק וגם ע"י מחלקה. ציינו 3 טיעונים **נגדיים** המצדדים בשימוש במנשק. נמקו או הדגימו בקצרה את טיעוניכם:

1. _____

_____

_____

_____

_____

2. _____

_____

_____

_____

_____

3. _____

_____

_____

_____

_____

ב. לאחר שביל השתכנע בנחיצות המנשק, החל דיון בינו ובין אומה על נחיצות השרותים הבאים. עבור כל אחד מהם ציינו האם יש לו מקום במנשק, במחלקה, בשניהם או באף אחד מהם. נמקו בקצרה:

1. add - לחיבור שני מספרים רציונליים

_____

_____

2. subtract – לחיסור שני מספרים רציונליים

_____

_____

3. multiply - להכפלת שני מספרים רציונליים

_____

_____

4. divide – לחלוקת שני מספרים רציונליים

_____

_____

5. equals - להשוואת שני מספרים רציונליים

_____

_____

6. gcd – לחישוב הגורם המשותף הגדול ביותר של שני שלמים

_____

_____

7. lcm – לחישוב מכנה משותף של שני שלמים

_____

_____

8. normalize – להבאת שבר פשוט למצב מצומצם

_____

_____

9. toString – לייצוג המספר כמחרוזת של שבר פשוט (למשל לצורכי הדפסה)

_____

_____

10. toDecimalString - לייצוג המספר כמחרוזת של שבר עשרוני (למשל לצורכי הדפסה)

_____

_____

ג. מה לגבי השרותים: getNumerator ו- getDenominator להחזרת המונה והמכנה של השבר בהתאמה? ביל ואומה הסכימו כי הדבר תלוי בהקשרי השימוש של הטיפוס החדש. ציינו באילו הקשרים יש הצדקה להכללת השרותים במנשק ובאילו אין. נמקו או הדגימו את תשובתכם:

_____

_____

_____

_____

_____

_____

_____

שאלה 2, 25 נקודות

ביל ואומה מתלבטים לגבי הקבעון (mutability) של הטיפוס החדש. טיפוס מקובע (immutable) הוא טיפוס שלא ניתן לשנות את ערכי העצמים שלו לאחר שנוצרו. ראינו בתרגול את הטיפוסים String ו- StringBuffer המממשים גרסא מקובעת ושאינה מקובעת למחרוזות.

1. ציינו את היתרונות שיש לטיפוס רציונלי מקובע ואת היתרונות שיש לטיפוס רציונלי שאינו מקובע:

   יתרונות הטיפוס הרציונלי המקובע (immutable):

_____

_____

_____

_____

   יתרונות הטיפוס הרציונלי שאינו מקובע (mutable):

_____

_____

_____

_____

2. באיזה מקרה יהיה ההבדל בין מימוש השרות equals במחלקות מקובעות לעומת מימושו במחלקות שאינן מקובעות? הסבירו במילים, אין צורך לכתוב קוד (רמז: המחלקה String).

_____

_____

_____

_____

_____

_____

3. הגדירו את המנשק MutableRational לתאור טיפוס רציונלי שאינו מקובע. על המנשק להכיל את החתימות המלאות של כל השרותים שציינתם בשאלה 1 סעיף ב' שהם חלק מהמנשק.

```
public interface MutableRational {

}
```

- הגדירו את המנשק ImmutableRational לתאור טיפוס רציונלי מקובע. על המנשק להכיל את החתימות המלאות של כל השרותים שציינתם בשאלה 1 סעיף ב' שהם חלק מהמנשק.

```
public interface ImmutableRational {

}
```

שאלה 3, 18 נקודות

א. המתכנת סטיב מימש את המחלקה Rational לייצוג טיפוס רציונלי ובה נכלל קוד ההשוואה הבא:

```
public class SimpleRational implements ImmutableRational {

	private int num;
	private int denom;

	public SimpleRational(int num, int denom) {
		this.num = num;
		this.denom = denom;
	}

	public boolean equals(SimpleRational   other) {
		return num==other.num && denom==other.denom;
	}

	// the rest of the class…

}
```

המחלקה אינה מכילה מימושים נוספים של השרות equals.
כדי לבדוק את נכונות המימוש כתב סטיב את מחלקת הבדיקה הבאה:

```
public class TestRationalEquality {

	public static void main(String[] args) {
		SimpleRational  r1 = new SimpleRational (1,2);
		SimpleRational r2 = new SimpleRational (1,2);
		ImmutableRational ir1 = new SimpleRational (1,2);
		ImmutableRational ir2 = new SimpleRational (1,2);

		// HERE
	}
}
```

עבור כל אחת משורות הקוד הבאות ציינו האם ניתן להגדירה בפונקצית ה-main של המחלקה TestRationalEquality במקום ההערה // HERE. סמנו את האפשרות המתאימה: האם תהיה שגיאת קומפילציה (אם כן, ציינו מה השגיאה), או שתהיה תעופה בזמן ריצה (אם כן, ציינו מאיזה סיבה), או שהקוד ירוץ בצורה תקינה (אם כן, ציינו מה יהיה פלט ההדפסה) או שהקוד ירוץ אך יחשוף באג במימוש השרות equals (אם כן, ציינו מה יהיה פלט ההדפסה, מה **אמור** היה להיות פלט ההדפסה ומה סיבת הבאג).

א.
```
System.out.println(r1.equals(r2));
```

שגיאת קומפילציה (מהי?) / תעופה בזמן ריצה (מה הסיבה) / קוד תקין (מה מודפס) / קוד תקין החושף באג (מה מודפס? מה אמור להיות מודפס? איפה הבאג?)

_____
__

ב.
System.out.println(ir1.equals(ir2));

שגיאת קומפילציה (מהי?) / תעופה בזמן ריצה (מה הסיבה) / קוד תקין (מה מודפס) / קוד תקין החושף באג (מה מודפס? מה אמור להיות מודפס? איפה הבאג?)

_____
__

_____
__

ג.
System.out.println(r1.equals(ir1));

שגיאת קומפילציה (מהי?) / תעופה בזמן ריצה (מה הסיבה) / קוד תקין (מה מודפס) / קוד תקין החושף באג (מה מודפס? מה אמור להיות מודפס? איפה הבאג?)

_____
__

_____
__

ד.
System.out.println(ir1.equals(r1));

שגיאת קומפילציה (מהי?) / תעופה בזמן ריצה (מה הסיבה) / קוד תקין (מה מודפס) / קוד תקין החושף באג (מה מודפס? מה אמור להיות מודפס? איפה הבאג?)

_____
__

_____
__

ה.
r1 = ir1;
System.out.println(r1.equals(ir1)

שגיאת קומפילציה (מהי?) / תעופה בזמן ריצה (מה הסיבה) / קוד תקין (מה מודפס) / קוד תקין החושף באג (מה מודפס? מה אמור להיות מודפס? איפה הבאג?)

_____
__

_____
__

ו.
ir1 = r1;
System.out.println(r1.equals(ir1));

שגיאת קומפילציה (מהי?) / תעופה בזמן ריצה (מה הסיבה) / קוד תקין (מה מודפס) / קוד תקין החושף באג (מה מודפס? מה אמור להיות מודפס? איפה הבאג?)

_____

_____

שאלה 4, 8 נקודות

הסבירו בקצרה את תפקיד המחלקות הבאות

א.

FormAttachment

_____
_____
_____
_____

ב.

Display

_____
_____
_____
_____

שאלה 5, 24 נקודות

כתבו איטרטור מסנן <UniqueIterator<T, אשר מחזיר מכל רצף של איברים זהים רק איבר בודד.

למשל עבור הרצף 1 1 1 2 2 2 3 3 5 5 5 1 1 1 4
הוא יחזיר ערכים כאילו הרצף הוא 1 2 3 5 1 4

האיטרטור יוגדר כמחלקה פנימית בתוך מחלקות אשר ממשות את המנשק <Iterable<T.
על האיטרטור המסנן להעזר בפונקציות של האיטרטור שמחזירה הפונקציה ()iterator של הטיפוס <Iterable<T. .

```
public class SomeClasss<T> implements Iterable<T>{

    // rest of the class…

    public class UniqueIterator<T> implements Iterator<T>{
```

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

```
    }
}
```