

[דף סיכום בחינה](#)

מזהה בחינה: 000000086482 מזהה סטודנט: 0208083196

מזהה קורס: 7284-46919 שם קורס: תכנות מונחה עצמים

מספר שאלה	ניקוד מירבי	ציון
1	25.00	18.00
2	25.00	21.00
3	25.00	25.00
4	25.00	16.00

ציון בחינה סופי : 80.00


הבחינה הבדוקה בעמודים הבאים

בית הספר הארצי להנדסאים (ע"ר)

מחברת בחינה


הוראות לנבחן

(13)



ID 0208083196

Exam 000000086482



- לפני תחילת הבחינה אנא קרא בעיון את ההוראות ומלא את הפרטים בכתב יד ברור.
שים לב: מחברות הבחינה נסרקות למאגר נתונים.
יש להקפיד: לא לקפל, לא לתלוש ולא לכתוב בצבעים או בעפרון.
- כתוב את פרטייך האישיים על גבי מחברת הבחינה מיד עם תחילת הבחינה.
- כתוב את תשובותיך בכתב יד ברור **משני צידי הדף.**
- ציין בראש כל דף את מספר השאלה עליה אתה עונה.
- כתוב את כל עבודתך (כולל הטייטה) במחברת זו. אל תכתוב דבר על דף אחר.
- מתח קו אלכסוני ורשום "מבוטל" על אותם חלקים שאינך רוצה שייבדקו.
- אם מחברת זו לא תספיק לך, בקש מהמשרת מחברת נוספת. כתוב על כל מחברת את פרטייך האישיים.
- הוצאת מחברת בחינה (או חלק ממנה) מחוץ לחדר הבחינה הינה עבירה משמעתית חמורה.
- על הנבחן לציית להוראות המשרתים. כל פנייה למשרת תיעשה על ידי הרמת יד בלבד.
- נבחן שיפריע למהלך הבחינה, יועמד לדין משמעת של ביה"ס. ניתן להפסיק את בחינתו ולתבעו בוועדת משמעת של ביה"ס.
- נבחן לא יהיה רשאי לעזוב את חדר הבחינה, אלא על פי נהלי בית הספר ובאישור המשרת.
- מותר להשתמש בחומר עזר רק אם הדבר הותר במפורש בכתב בגוף שאלון הבחינה. הימצאות חומר אסור בידי הנבחן בשעת הבחינה על שולחנו, או בסמוך אליו, מהווה עבירה משמעת חמורה.

הערה: ההוראות מתייחסות לזכר ולנקבה כאחד.

תאריך הבדיקה: _____

חתימת המרצה: _____


חתימת הסטודנט

ציון:

בהצלחה!

כאן 19:32 10'0

לתשומת לבך:

יש לכתוב את תשובותיך משני צידי הדף
אין לכתוב מעבר לקו האדום
יש לכתוב את הבחינה בעט (כחול/שחור) בלבד.

שם משפחה: _____ ח"ן _____ שם פרטי: _____ מוצ'ה _____

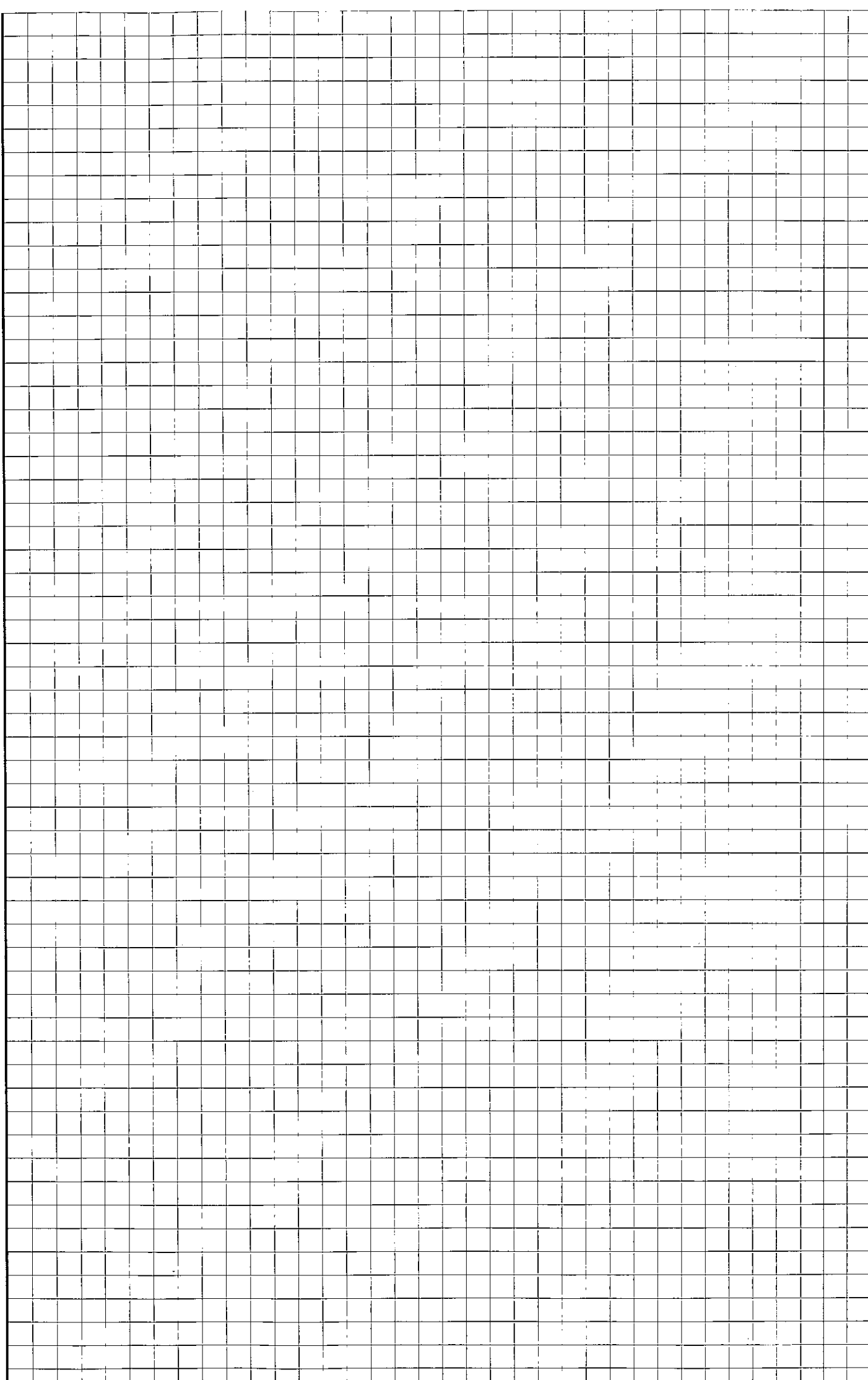
מספר תעודת זהות: 6 1 9 0 6 / 2 3

תאריך הבחינה 19/06/23 מגמה: תוכנה מחזור: _____

שם הקורס: תכנות מערכת עזרים שם המרצה: צ'רין בוג'ין

סמסטר: חורף / אביב / קיץ מועד: אמצע / ב / מיוחד

מחברת מספר 1 מתוך סה"כ מחברות 1



שם: הנריק חיון

208083196

בית ספר להנדסאים טכניון

מגמת תוכנה

מבחן מגן – מועד א'

תכנות מונחה עצמים

ב JAVA

הנחיות לבחן

שלוש שעות

משך הבחינה:

מבנה השאלון ומפתח הערכה:

במבחן זה 4 שאלות. יש לענות על כולן.

סה"כ: 100 נקודות – כל השאלות שוות בערך.

חומר עזר מותר לשימוש:

1. מחשבון (אין להשתמש במחשב כף יד או כמחשבון עם תקשורת חיצונית).
2. קלסר אחד בלבד עם חומר ההרצאות. אין להוציא דפים מהקלסר. אין לצרף ספרים או חוברות עם פתרונות לקלסר

הוראות כלליות:

1. בטרם תתחיל לענות על השאלות, קרא בעיון את כל השאלון וודא שההנחיות בדף השער מובנות לך היטב.
2. כתוב את התשובות במחברת הבחינה בעט בלבד בכתב יד ברור ומסודר.
3. התחל כל תשובה בעמוד חדש וציין בראשו את מספר השאלה והסעיף (אין צורך להעתיק את השאלה או הסעיף).
4. אם אתה זקוק לטיוטה, הקצה לה מקום במחברת הבחינה ורשום באופן ברור "טיוטה" ומתח קו על הכתוב כדי שלא ייבדק.
5. בתשובה לשאלה חישובית, עליך להציג את שלבי הפתרון באופן מפורט ולהסבירם בקצרה. קבלת מרב הנקודות מותנית במילוי דרישה זו.
6. בכל שאלה קיימים הנתונים לפתרונה. אם לדעתך חסר נתון, הוסף אותו על פי שיקול דעתך ופתור בעזרתו את השאלה. ציין בתשובתך את הנתון שהוספת ונמק את בחירתך.

בהצלחה !

נתונות המחלקות הבאות :

```
class B extends A {
    public int _a;
    private int _b;

    public B(int a, int b) {
        super(a, b);
        _a = ++a;
        _b = ++b;
    }

    public int getB() {
        return _b;
    }

    public int f() {
        return super.f();
    }

    public int f(int c) {
        return super.f(_b + c);
    }

    public String toString() {
        return "a=" + _a + "b= " +
    }
}

class A {
    public int _a;
    private int _b;

    public A(int a, int b) {
        _a = a;
        _b = b;
    }

    public int getB() {
        return _b;
    }

    public int f() {
        return f(0);
    }

    public int f(int c) {
        return _b + c;
    }

    public String toString() {
        return "a=" + _a + "b= " + _b;
    }
}

public class Q1 {
    public static void main(String[] args) {
        A aa = new A(0, 1);
        A ab = new B(0, 1);
        B bb = new B(-2, -1);
        // *** כאן תיכנס השורה ***
    }
}
```

לגבי כל אחת משורות הקוד 1 – 5 סמנו מה יקרה אם נכתוב אותה במקום שורות ההערה. ב main של המחלקה - Q1

- א. תהיה שגיאת קומפילציה – במקרה זה יש לציין את הסיבה לשגיאה.
- ב. תהיה שגיאה בזמן ריצה - במקרה זה יש לציין את הסיבה לשגיאה.
- ג. הקוד ירוץ בצורה תקינה. במקרה זה יש לכתוב את הפלט שיודפס על המסך.

- (1) System.out.println(bb._b);
- (2) System.out.println(((B)aa).getB());
- (3) System.out.println(((A)ab).getB());
- (4) System.out.println(aa);
- (5) System.out.println(ab.f());

<pre> public class First { private static int count = 0; protected int x; protected int y; public First (int num) { this.x = num; this.y = num; count ++; System.out.println ("First 1"); } public First (int num1 , int num2) { this.x = num1; this.y = num2; count ++; System.out.println ("First 2"); } public static int getCount() { return count; } public int getX() { return x; } public int getY() { return y; } public int sum() { return this.x + this.y; } public void add(First other) { this.x += other.x; this.y += other.y; System.out.println("x = "+ this.x + "y = "+ this.y); } } </pre>	<pre> public class Second extends First { private int z; public Second (int num) { super (num); this.z = num; System.out.println ("Second"); } public int sum() { return super.sum() + this.z; } public void add (First other) { this.x += other.getX(); this.y += other.getY(); if (other instanceof Second) this.z += ((Second)other).z; System.out.println("x = "+ this.x + "y = "+ this.y+ " z=" +this.z); } } </pre>
--	--


```
public class Tester
{
    public static void main(String[] args)
    {
        First f1 = new First (40);
        First f2 = new First (40, 50);
        First f3 = new Second (100);
        Second s1 = new Second (100);
        Second s2 = new Second (100);
        // ***
    }
}
```

- א. ציירו את העצמים שנוצרו בפעולה main , וכתבו את הפלט של הפעולה.
- ב. הציבו כל אחת מן הפקודות 1-10 שלהלן בפעולה main במקום המצוין לעיל ב- *** .
- כתבו במחברת את מספר הפקודה וציינו אם הקוד תקין או לא תקין.
- אם הקוד תקין – כתבו את הפלט, ואם הוא אינו תקין, הסבירו מדוע.
- הערה: אין קשר בין הפקודות. כלומר, יש להתייחס לכל פקודה כאילו היא היחידה בפעולה main .

1. System.out.println ("Total = " + First.getCount());
2. System.out.println ("Total = " + Second.getCount());
3. System.out.println ("sum = " + s1.sum());
4. System.out.println ("sum = " + f3.sum());
5. s1=new First (100);
6. f1.add (s2);
7. s1.add (s2);
8. s2.add (f3);
9. ((First)s1).add (f1);
10. s1=new Second (100, 100);

במסעדת "הרמה" אפשר להזמין מקום עבור סועד אחד או יותר. בסוף הארוחה, כל אחד מן הסועדים משלם בנפרד על מה שאכל. כל סועד יכול לבחור אם לשלם במזומן אֶן באמצעות אפליקציה אֶן בכרטיס אשראי. למשלמים בכרטיס אשראי יש אפשרות לפצל את החשבון לכמה תשלומים שווים.

כדי לנהל מערכת תשלומים, מפתחים עבור המסעדה פרויקט ובו הממשק **IPayment (Interface)**, והמחלקות **Restaurant, Reservation, Credit, App, Cash**, כמפורט להלן:

❖ **IPayment** – תשלום

בממשק קיימת הפעולה: `double getPrice()`

❖ **Cash** – תשלום במזומן

תכונות המחלקה:

- `sumCash` – הסכום לתשלום במזומן, מטיפוס ממשי.

- `name` – שם הלקוח, מטיפוס מחרוזת.

❖ **App** – תשלום באמצעות אפליקציה

תכונות המחלקה:

- `sumApp` – הסכום לתשלום באפליקציה, מטיפוס ממשי.

- `phoneNumber` – מספר טלפון, מטיפוס מחרוזת.

❖ **Credit** – תשלום בכרטיס אשראי

תכונות המחלקה:

- `num` – מספר התשלומים, מטיפוס שלם.

- `part` – הסכום בכל תשלום, מטיפוס ממשי.

- `creditNumber` – מספר כרטיס האשראי, מטיפוס מחרוזת.

❖ **Reservation** – הזמנה

תכונות המחלקה:

- `date` – תאריך ההזמנה, מטיפוס מחרוזת.

- `total` – החשבון סך הכול של כל הסועדים יחד באותה הזמנה, מטיפוס ממשי.

- `payments` – מערך השומר את התשלום ששילם כל אחד מן הסועדים באותה הזמנה, מטיפוס **IPayment** (המערך

ללא ערכי null).

❖ **Restaurant** – המסעדה

תכונות המחלקה:

- `array` – מערך בגודל 10,000, מטיפוס **Reservation**.


- `current` – כמות ההזמנות השמורות, מטיפוס שלם.

הניחו שההזמנות נשמרות ברצף במערך ואין ביניהן ערכי null.

הערה: הניחו שבכל מחלקה הוגדרו פעולות `get` ו-`set` ופעולות בונות.

(המשך בעמוד הבא)

א. (1) סרטטו תרשים הייררכייה המתאר את הקשרים של המחלקות והממשק של הפרויקט.

יש לסמן מימוש ממשק באמצעות החץ <---- והכלה באמצעות הסימן .

(2) המסעדה עושה הנחות לסועדים בהתאם לצורת התשלום: סועד שמשלם במזומן והסכום לתשלום הוא

מעל 200 שקלים זכאי ל- 10% הנחה, סועד שמשלם באמצעות האפליקציה זכאי תמיד ל- 5% הנחה,

וסועד שמשלם בכרטיס אשראי משלם מחיר מלא.

פעולת הממשק getPrice מחזירה את סך התשלום שכל סועד משלם לאחר שקלול ההנחה (כאשר אין הנחה,

הפעולה מחזירה מחיר מלא). ממשו את פעולת הממשק getPrice בכל אחת מן המחלקות הנדרשות.

ב. כתבו במחלקה **Reservation** פעולה ששמה cashTotal. הפעולה מחזירה את סכום הכסף המזומן שהתקבל בהזמנה (לאחר שקלול ההנחה).

ג. לפניכם פעולה במחלקה **Reservation** :

```
public void printDetails () {  
    for (int i = 0; i < payments.length; i++) {  
        System.out.println (payments[i].getDetails());  
    }  
}
```

הפעולה מדפיסה עבור כל סועד פרטים מסוימים על פי הקריטריונים שלהלן:

- אם שילם הסועד במזומן, הפעולה מדפיסה את השם שלו.

- אם שילם הסועד באמצעות האפליקציה, הפעולה מדפיסה את מספר הטלפון שלו.

- אם שילם הסועד בכרטיס אשראי, הפעולה מדפיסה את מספר כרטיס האשראי שלו.

הוסיפו את הפעולות הנחוצות לפרויקט כדי שהפעולה תבצע את הנדרש. ציינו עבור כל פעולה שהוספתם לאיזו מחלקה או לאיזה ממשק היא שייכת.

הערה: אין לשנות את הפעולה printDetails.

נתונות שלוש מחלקות – Mammal (יונק), Antelope (אנטילופה) ו- Beaver (בונה):

```
public class Mammal {
    protected int weight;

    public Mammal (int w) {
        weight = w;
    }
    public int getWeight () {
        return weight;
    }
    public boolean isSame (Mammal other) {
        System.out.println ("In Mammal");
        return (this == other);
    }
}

.....

public class Antelope extends Mammal {
    public Antelope (int w) { super (w); }
    public boolean isSame(Antelope other) {
        System.out.println ("In Antelope");
        return ((other != null) && (this.weight == other.weight));
    }
}

.....

public class Beaver extends Mammal {
    public Beaver (int w) { super (w); }
    public boolean isSame (Mammal other) {
        System.out.println ("In Beaver");
        return ((other != null) && (other instanceof Beaver) && (this.weight == ((Beaver)other).weight));
    }
}

.....
```

(שימו לב: המשך השאלה בעמוד הבא.)

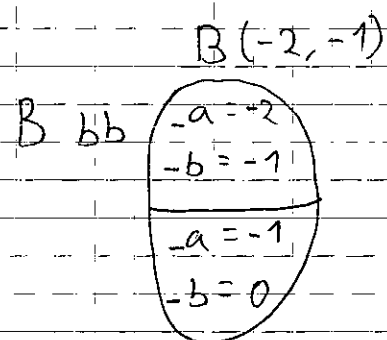
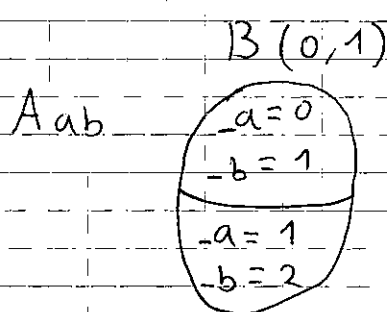
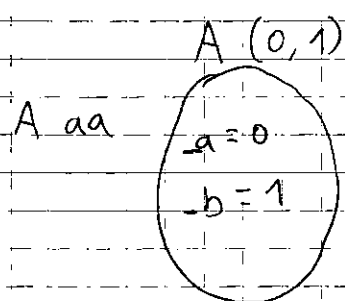

```
public class Program {
    public static void main (String[] args) {
        Antelope a1 = new Antelope (10);
        Object a2 = new Antelope (10);
        Beaver b1 = new Beaver (10);
        Mammal b2 = new Beaver (10);
        *****
    }
}
```

המחלקה Program נמצאת בחבילה (Package) שונה מן המחלקה Mammal .
 הציבו כל אחת מן השורות 1–10 שלהלן בפעולה main , במקום שמסומן לעיל בכוכביות ***** .
 כתבו במחברת את מספר השורה וציינו אם הקוד תקין או לא תקין. אם הקוד תקין – כתבו את הפלט, ואם הוא
 לא תקין – הסבירו מדוע.

1. `System.out.println (a1.weight);`
2. `System.out.println (((Beaver)a2).getWeight());`
3. `System.out.println (a1.isSame (a2));`
4. `System.out.println (a2.isSame (a1));`
5. `System.out.println (b1.isSame (b2));`
6. `System.out.println (b2.isSame (b1));`
7. `System.out.println (a1.isSame ((Beaver)b2));`
8. `System.out.println (a1.isSame ((Antelope)a2));`
9. `System.out.println (b1.isSame ((Antelope)a2));`
10. `System.out.println (b1.isSame ((Beaver)a2));`

בהצלחה!

שאלה 1:



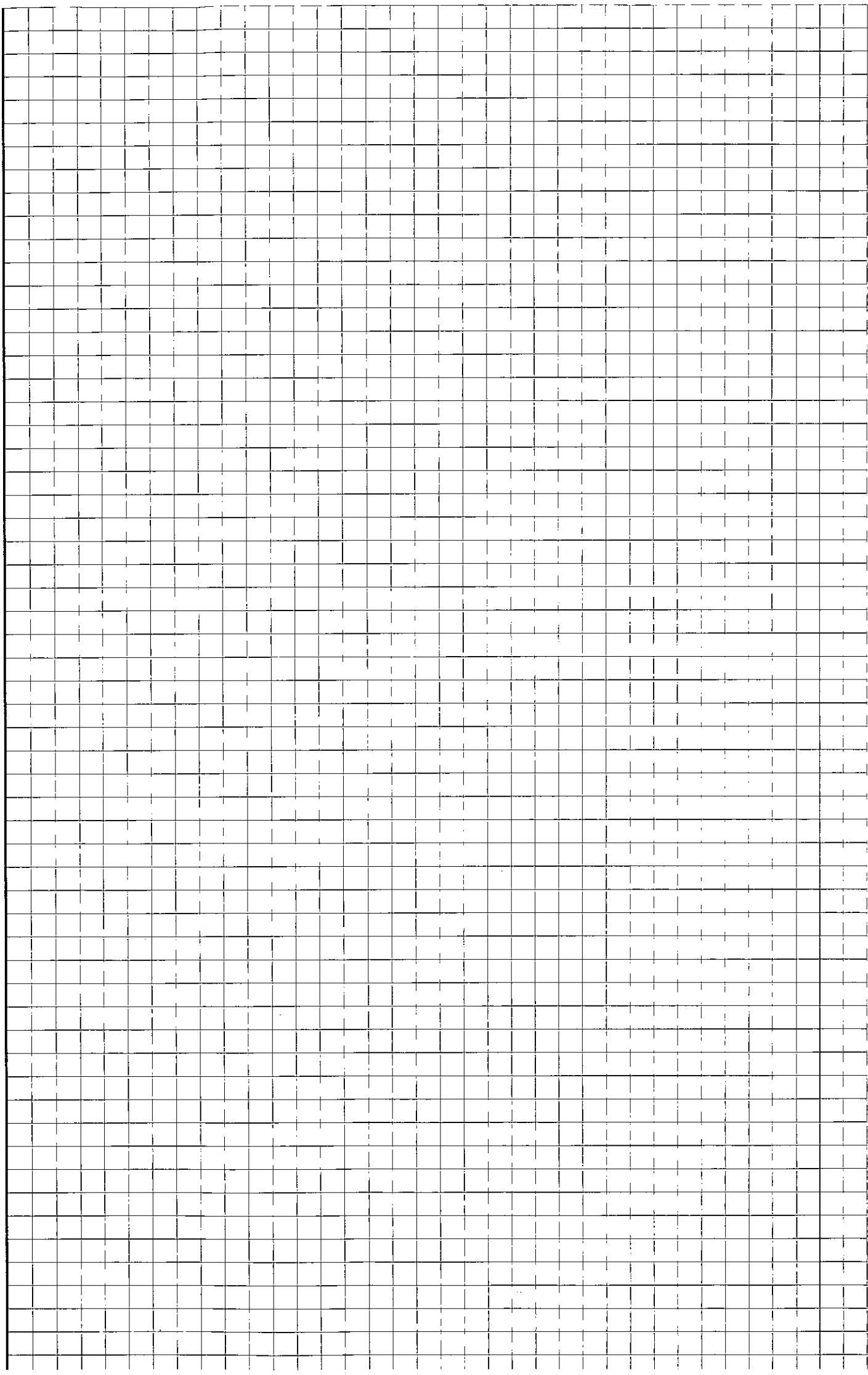
(1) מקור יחל בזורה תקנה. כלל: 0 ✗

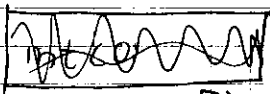
(2) תהיה בטוח שיש חיבור - כיוון שיש להם את aa שהם שווים A
 B כי B יורש A וכלל הפסק ✓

(3) מקור יחל בזורה תקנה. כלל: 2 ✓

(4) מקור יחל בזורה תקנה. כלל: $a=0 \quad b=1$ ✓

(5) מקור יחל בזורה תקנה. כלל: 1 ✗

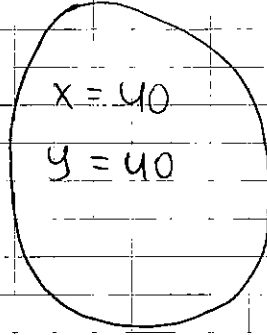




First (40)

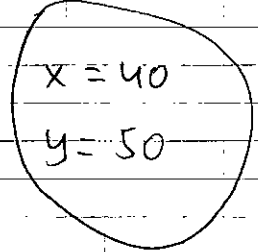
int count = 1 2 3 4 5

First f1



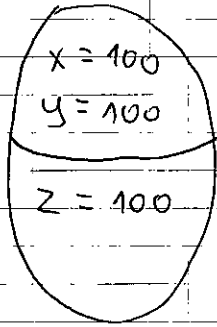
First (40, 50)

First f2



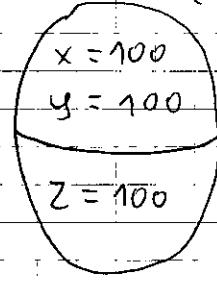
Second (100)

First f3



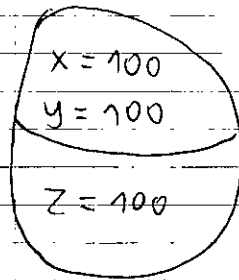
Second s1

Second (100)



Second (100)

Second s2



הקצו זיכרון = 100

:C6

First 1

First 2



First 1

Second

First 1

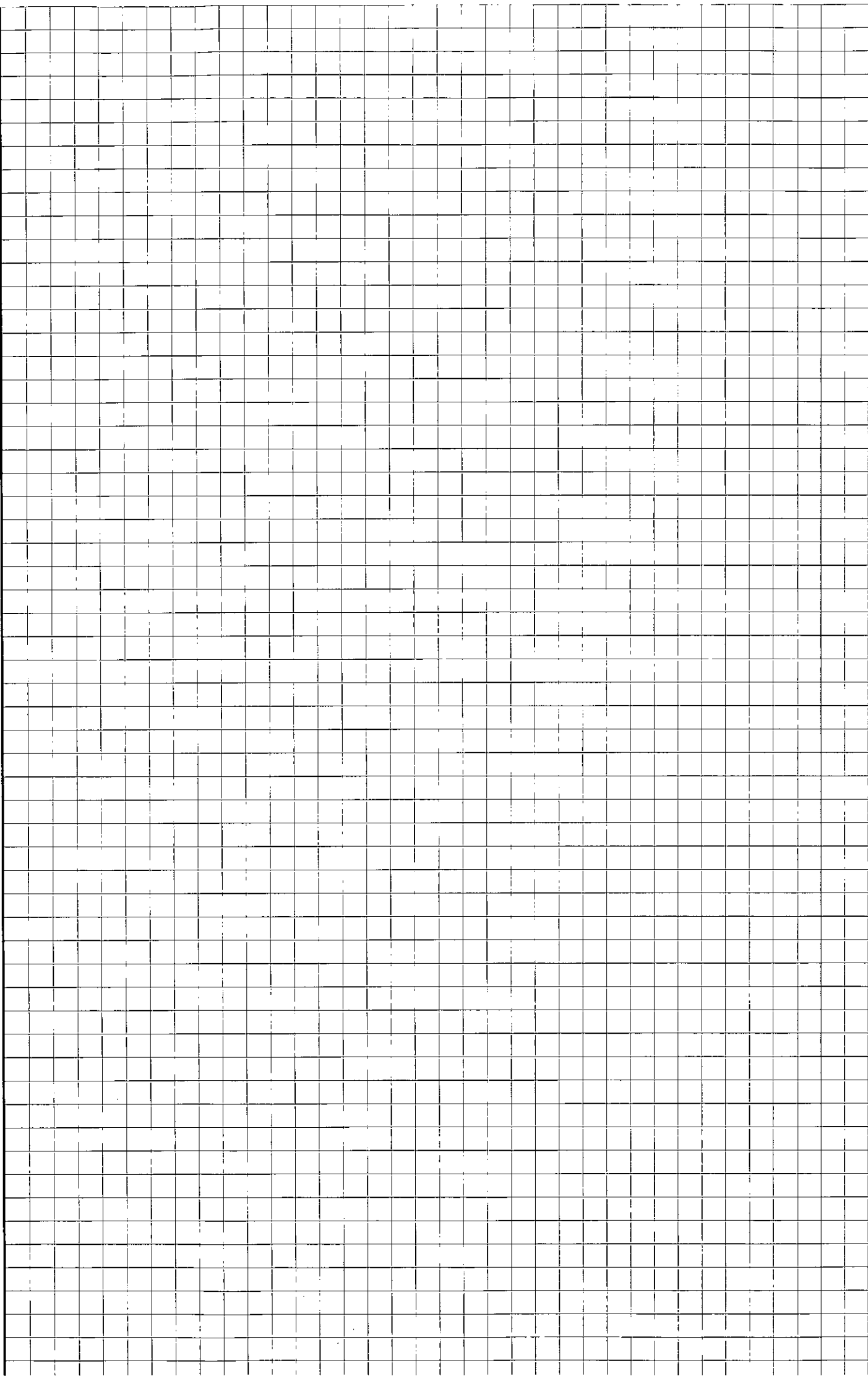
Second



First 1

Second

יש רק חמש שורות פלט



שאלה 2: השק

ב-1) קוד תקין. כן: $Total = 5$ ✓

2) קוד לא תקין. התקציה `getCount` לא נמצאת במחלקה `second` ✗
 התקציה `אומרת` סליה אך אין גיבן למכור אליה קצרה `first`

3) קוד תקין. כן: $sum = 300$ ✓

4) קוד תקין. כן: $sum = 300$ ✓

5) קוד לא תקין. לא ניתן למכור אובייקט `first` או `second` ✓
 משה `second` כיוון `second` וכל `first` ולא הכנס

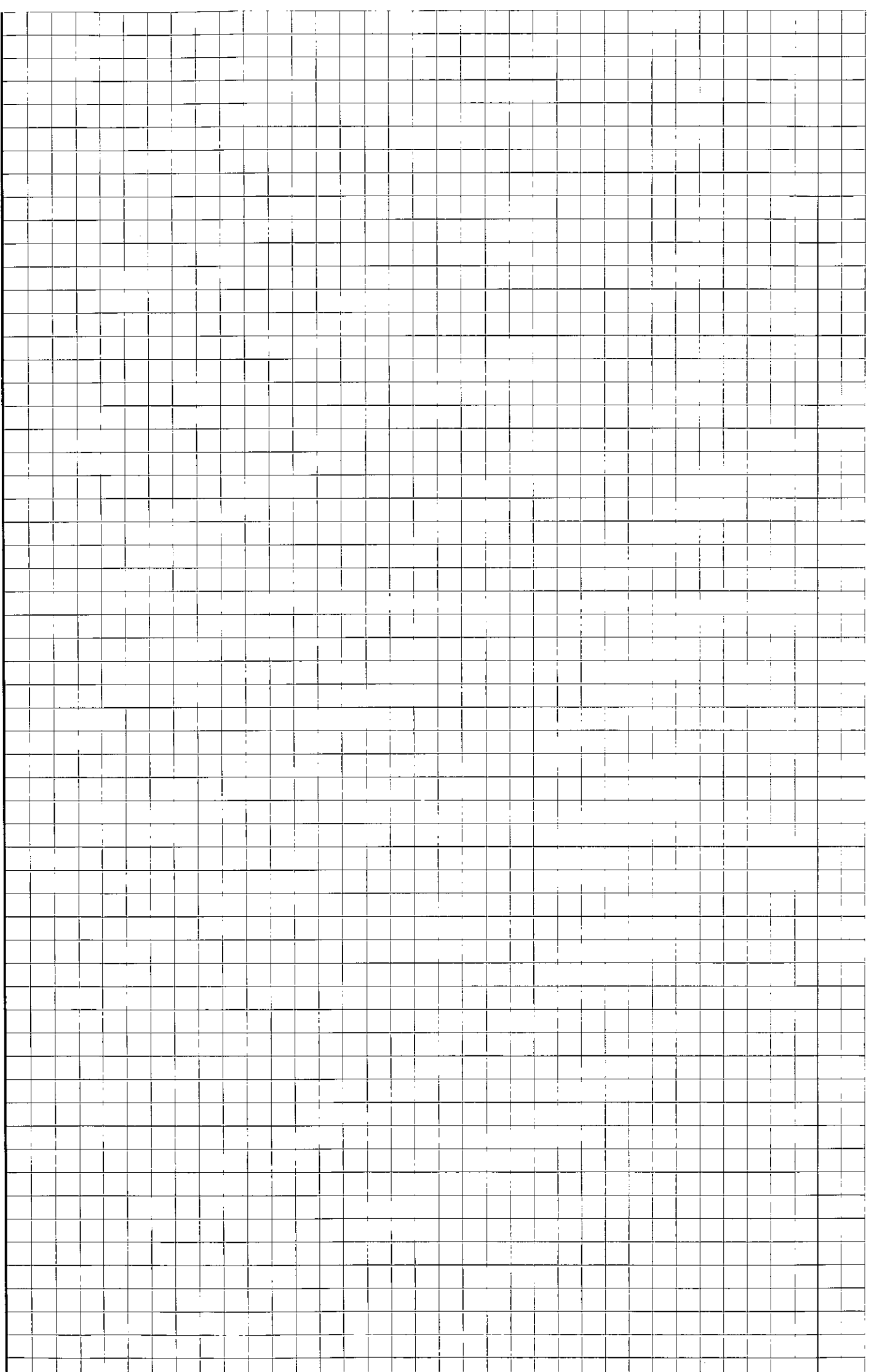
6) ~~קוד לא תקין. התקציה `first.add` אינה קיימת במחלקה `first`.
 התקציה `second` אינה קיימת במחלקה `second`.
`first.add((first) s2)` ✓
 $x = 140$ $y = 140$ כן: קוד תקין. כן: ✓~~

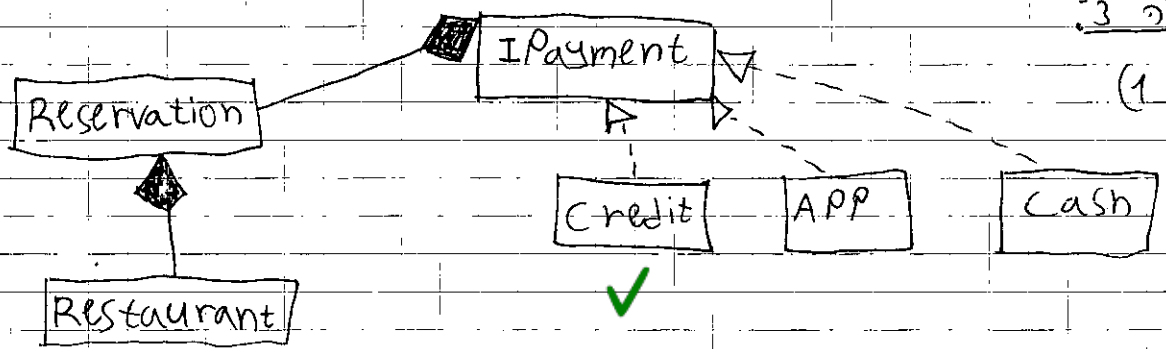
7) קוד תקין. כן: $x = 200$ $y = 200$ $z = 200$ ✓

8) קוד תקין. כן: $x = 200$ $y = 200$ $z = 200$ ✓

9) קוד תקין. כן: $x = 140$ $y = 140$ $z = 100$ ✓

10) קוד לא תקין, במחלקה `second` אין `first` ✓
 מחלקה





:cash mpn

```

public double getPrice() {
    if (this.sumCash > 200)
        return (this.sumCash * 90) / 100.00;
    return this.sumCash;
}
  
```

:APP mpn

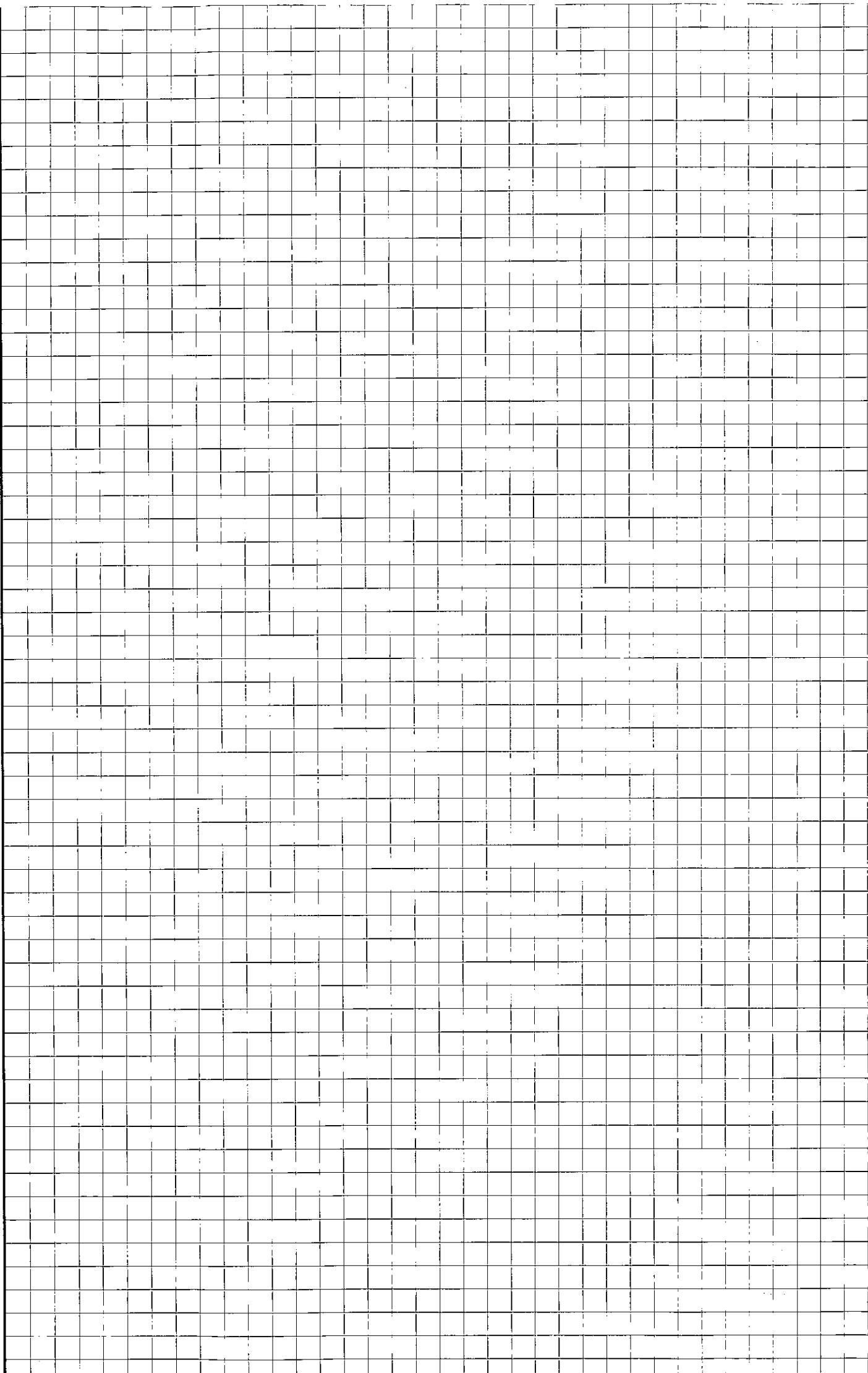
```

public double getPrice() {
    return (this.sumAPP * 95) / 100.00;
}
  
```

:credit mpn

```

public double getPrice() {
    return this.num * this.part;
}
  
```



punn -3 dice

```
public double cashTotal() {
```

(2)

```
    double cashSum = 0;
```

```
    for (int i = 0; i < this.payments.length; i++) {
```

```
        if (this.payments[i] instanceof cash) {
```

✓

```
            cashSum += ((cash) this.payments[i]).getPrice();
```

```
        return cashSum;
```

```
    }
```

```
    :cash mpnwa
```

(2)

```
public String getDetails() {
```

✓

```
    return this.name;
```

```
}
```

```
    :App mpnwa
```

```
public String getDetails() {
```

✓

```
    return this.phoneNumber;
```

```
}
```

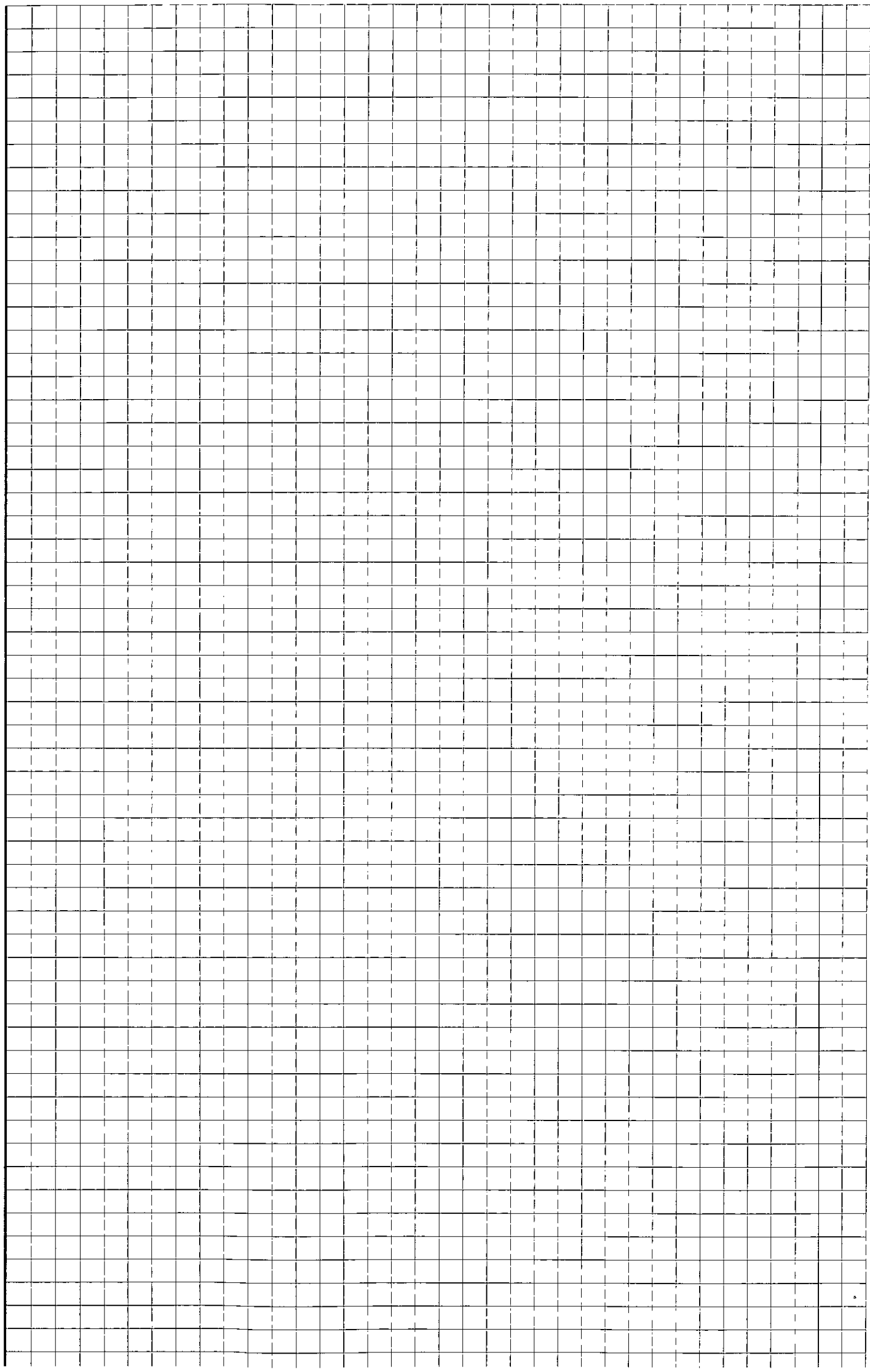
```
    :credit mpnwa
```

```
public String getDetails() {
```

```
    return this.creditNumber;
```

✓

```
}
```



אלופ 4

1 קוד תקין. כל: 10

✗

2 קוד לא תקין. זה לא יתקן כי הכתובת setheight לא נמצאת

3 מתודה Beaver שליוו את האובייקט ✓

3 קוד תקין. כל: In Antelope

✗

true

4 קוד תקין. כל: In Antelope

✗

true

5 קוד תקין. כל: In Beaver

✓

true

6 קוד תקין. כל: In Beaver

✓

true

7 לא תקין, הכתובת isSame במתודה Antelope שליוו את האובייקט

8 Beaver שליוו את האובייקט ✗

8 קוד תקין. כל: In Antelope

✓

true

9 קוד תקין. כל: In Beaver

✓

false

10 שטח בין רצה בין לא יהיה ניתן להחזיר את 2 שטח

✓

Beaver שליוו את Antelope

