

דוח משוב לסטודנט

מזהה סטודנט

0336540331

מזהה מבחן

000000091848

תאריך בחינה

יום ראשון, 16 במרץ 2025

מזהה קורס

7284-50495

שם קורס

תכנות מונחה עצמים

מרצה

כהן איינהורן ענבל

ניקוד שאלות פתוחות	ציון מבחן מקורי	ציון מבחן סופי
70.00	70.00	70.00

סיכום

מספר שאלה	ניקוד	ניקוד מירבי
1.1	10.00	10.00
1.2	3.00	5.00
1.3	8.00	15.00
2.1	3.00	6.00
2.2	6.00	8.00
2.3.1	5.00	7.00
2.3.2	6.00	7.00
2.3.3	6.00	8.00
2.4	4.00	4.00
3.1	9.00	14.00
3.2	4.00	4.00
3.3	0	4.00
3.4	4.00	4.00
3.5	2.00	4.00



מה"ט המכון הממשלתי להכשרה בטכנולוגיה ובמדע

בית הספר הארצי להנדסאים (ע"ר)

מחברת בצינה

הוראות לנבחן

(24)



ID 0336540331
Exam 000000091848


- (24)



ID 0336540331

Exam 000000091848



1. לפני תחילת הבחינה אנא קרא בעיון את ההוראות ומלא את הפרטים בכתב יד ברור.
 2. **שיום לב:** מחברות הבחינה נסרקות למאגר נתונים.
 3. **יש להקפיד: לא לקפל, לא לתלוש ולא לכתוב בצבעים או בעפרון.**
 4. כתוב את פרטייך האישיים על גבי מחברת הבחינה מיד עם תחילת הבחינה.
 5. כתוב את תשובותיך בכתב יד ברור **משני צידי הדף.**
 6. ציין בראש כל דף את מספר השאלה עליה אתה עונה.
 7. כתוב את כל עבודתך (כולל הטיטה) במחברת זו. אל תכתוב דבר על דף אחר.
 8. מתח קו אלכסוני ורשום "מבוטל" על אותם חלקים שאינך רוצה שייבדקו.
 9. אם מחברת זו לא תספיק לך, בקש מהמשרת מחברת נוספת. כתוב על כל מחברת את פרטייך האישיים.
 10. הוצאת מחברת בחינה (או חלק ממנה) מחוץ לחדר הבחינה הינה עבירה משמעתית חמורה.
 11. על הנבחן לציית להוראות המשרתים. כל פנייה למשרת תיעשה על ידי הרמת יד בלבד.
 12. נבחן שיפריע למהלך הבחינה, יועמד לדין משמעת של ביה"ס. ניתן להפסיק את בחינתו ולתבעו בוועדת משמעת של ביה"ס.
 13. נבחן לא יהיה רשאי לעזוב את חדר הבחינה, אלא על פי נהלי בית הספר ובאישור המשרת.
 14. 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:


UJDU

אני מאשר כי קראתי את ההוראות האמורות לעיל:

חתימת הסטודנט

צ"ל:

בהצלחה!

לתשומת לבך:

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

שם משפחה: שמ' אורן שם פרטי: עאנון

3	3	6	5	4	0	3	3	1
---	---	---	---	---	---	---	---	---

מספר תעודת זהות:

תאריך הבחינה 16.03.25 מגמה: _____ מחזור: 48-5

שם הקורס: מכונות מולטי-פונקציונליות שם המרצה: ד"ר חגית כהן א"י תות

מועד: אמצע / א / ב / מיוחד

סמסטר: חורף / אביב / קיץ

מחברת מספר מתוך סה"כ מחברות

for
what.

3×2.5

double
something.

בית הספר הארצי להנדסאים

מבחן סמסטר א' 2025 מועד ב – תכנות מונחה עצמים

שם משפחה	שם פרטי	מס' ת. זהות	כיתה
שא"און'	שא"און'	336540331	48-5

קורס: מחשבים 48/5+6

משך הבחינה: 150 דקות

מרצה: ענבל כהן-איינהורן.

תאריך: 16/03/2025

הוראות כלליות

1. במבחן זה 3 שאלות יש לענות על כל השאלות.
2. יש להשתמש אך ורק בחומר העזר שהינו קלסר אחד עם חומר הרצאות.
3. אין להעביר חומר עזר מנבחן אחד למשנהו.
4. אין להעתיק תשובות מחומר עזר שהותר בשימוש.
5. יש לכתוב את התשובות במחברת הבחינה בלבד! בעט (כחול/שחור) ובכתב יד ברור.
6. בדיקת המבחן תתבצע בצירוף טופס הבחינה.
7. חל איסור מוחלט להוציא שאלון מחדר הבחינה!

בהצלחה!

שאלה 1: (30 נקודות)

נתונות שתי מחלקות First ו-Second :

```
public class First
{
    private double someValue;

    public First()
    {
    }

    public First(double someValue)
    {
        this.someValue = someValue;
    }

    public double what()
    {
        return this.someValue * 3.5;
    }
}

public class Second extends First
{
    private String blabla;
    private int value;

    public boolean isSmaller()
    {
        return this.value < this.what();
    }
}
```


א. לפניכם ארבעה משפטים.
קיבעו לכל אחד מהם אם הוא נכון או אינו נכון, ונמקו קביעתכם.
תשובה ללא נימוק – לא תזכה בנקודות.

- 1 – המחלקה First יכולה לגשת ישירות לתכונה value של המחלקה Second.
- 2 – המחלקה Second יכולה לגשת ישירות לכל התכונות ולכל הפעולות של המחלקה First.
- 3 ✓ המחלקה Second יורשת את כל התכונות וכל הפעולות של המחלקה First.
- 4 – המחלקה First יורשת את הפעולה isSmaller() מהמחלקה Second.

ב. הוסיפו וממשו במחלקה Second פעולה בונה שמקבלת כפרמטרים מחרוזת ומספר שלם ומאתחלת את התכונות כאשר התכונה someValue תאותחל להיות מחצית מערכו של value. חתימת הפעולה:

```
public Second(String blabla, int value)
```

ענו על סעיף ג בהנחה שהפעולה של סעיף ב קיימת ותקינה:

ג. לפניכם שתי הגדרות מפעולה ראשית main שנמצאת במחלקה TestAll:

```
First something = new Second("Hello", 5);  
First nextThing = new First(42.2);
```

בעבור כל אחת מ 5 ההוראות שלפניכם, קיבעו אם היא תקינה או אינה תקינה. אם אינה תקינה, נמקו קביעתכם.
תשובה ללא נימוק לא תזכה בנקודות.

- 1. boolean flag1 = something.isSmaller();
- 2. boolean flag2 = nextThing.isSmaller();
- 3. boolean flag3 = something.what() > 5;
- 4. boolean flag4 = ((Second)something).isSmaller();
- 5. boolean flag5 = ((Second)nextThing).isSmaller();

שאלה 2: (40 נקודות)

נתונה מערכת המתארת משחק קלפים הכוללת 4 המחלקות הבאות:
המחלקה **Card** (המתארת קלף) ולה התכונות המוגדרות כ `protected`:

- מספר קלף – `id` מטיפוס מספר שלם וייחודי
- כוח – `power` מטיפוס מספר שלם
- צבע – `color` מטיפוס מחרוזת

והפעולות:

- `calcForce()` – מחזירה את הכוח של ה `Card`
- `toString()` – מחזירה מחרוזת המתארת את פרטי ה `Card`

הניחו שקיימת פעולה בונה שמקבלת את תכונות המחלקה, פעולות `get` ו `set` לכל התכונות.

אין צורך לממש את המחלקה **Card**!

המחלקה **Goblin** (המתארת קלף מסוג גובלין) היורשת מ `Card` ולה התכונה:

- מספר כלים – `tools` מספר שלם

המחלקה **Dragon** (המתארת קלף מסוג דרקון) היורשת מ `Card` ולה התכונה:

- אורך כנף – `wingLen` מטיפוס מספר שלם

המחלקה **Game** ולה התכונות:

- `cards` - מערך של קלפים (עד 10 קלפים)
- `lastPosition` – שמירת המיקום הפנוי האחרון להוספת קלף

* הניחו שקיימת פעולה בונה ופעולות `set` ו `get` לכל התכונות של כל המחלקות הנ"ל.

אין צורך לממש!

סעיף א:

יש לכתוב את הכותרות של כל המחלקות חוץ מ Card
יש לממש את הפעולה toString בכל המחלקות חוץ מ Card

סעיף ב:

יש לממש את הפעולה calcForce() במחלקות **Goblin** ו **Dragon** לפי ההנחיות הבאות:

במחלקה Goblin על הפעולה להחזיר את הכוח פלוס מספר הכלים

במחלקה Dragon על הפעולה להחזיר את הכוח כפול אורך הכנף

סעיף ג:

יש לממש במחלקה **Game** את הפעולות הבאות:

- (1) removeCard(int id) – מקבלת id של קלף ומוציאה את הקלף ממערך הקלפים (שימו לב לא להשאיר חורים של null במערך, אין צורך לשמור על סדר הקלפים במערך)
- (2) calcTotalForce(String c) – מחשבת ומחזירה את סכום כל הכוחות של הקלפים בצבע c.
- (3) avgWingAndTotalTools () – הפעולה מדפיסה:
 - את סכום הכלים של כל קלפי Goblin
 - את ממוצע אורכי הכנף של קלפי Dragon

סעיף ד

שרטטו דיאגרמת (מידול) UML של 4 המחלקות – כולל הקשרים ביניהן.
(יש לציין שם מחלקה בלבד)

שאלה 3: (30 נקודות)

א. נתונות ארבע מחלקות: Ski, SkiP, SkiA, SkiC. עקבו באמצעות מעקב אובייקטים אחר הפעולה main במחלקה SkiC, ורשמו את הפלט. יש לכלול בעבור כל עצם את ערכי התכונות שלו ובנוסף יש לציין את ערכי כל תכונות המחלקה.

```
public class Ski {
    protected String a;
    private static int tS = 0;

    public Ski(String a) {
        this.a = a;
        tS++;
    }

    public static int getT() {
        return tS;
    }

    public void perform() {
        System.out.println(this.a + " is skiing down the
                                slope.");
    }
}
```

```
public class SkiA extends Ski {
    private static int tA = 0;

    public SkiA(String a) {
        super(a);
        tA++;
    }

    public void perform() {
        System.out.println(a + " falls!");
    }

    public static int getTA() {
        return tA;
    }
}
```



```

public class SkiP extends Ski {
    private static int tP = 0;

    public SkiP(String a) {
        super(a);
        tP++;
    }

    public void perform() {
        System.out.println(a + " performs an amazing
                                trick!");
    }

    public static int getTP() {
        return tP;
    }
}

```

```

public class SkiC {
    public static void main(String[] args) {
        Ski[] c = new Ski[5];
        c[0] = new SkiA("Daria");
        c[1] = new SkiP("Itay");
        c[2] = new SkiP("Idan");
        c[3] = new SkiA("Ben");
        c[4] = new SkiP("Hila");

        for (int i = 0; i < c.length; i++) {
            c[i].perform();
        }

        System.out.println("\nT: " + Ski.getT());
        System.out.println("T pro " + SkiP.getTP() +
                            ", T amat: " + SkiA.getTA());
    }
}

```

- ב. האם יש שימוש במנגנון הפולימורפיזם? אם כן ציינו היכן בא לידי ביטוי ומה היתרון שלו בתוכנית זו, אם לא הסבירו היכן הייתם מוסיפים שימוש שלו.
- ג. מה יקרה אם נגדיר את הפעולה `perform()` כמתודה `static` במחלקה `Ski`? האם הפלט ישתנה? הסבירו תשובתכם.
- ד. האם ניתן להפעיל את הפעולה `getTP()` על `c[3]`? נמקו תשובתכם ואם כן הראו כיצד.
- ה. האם ניתן להפעיל את הפעולה `getTP()` על `c[2]`? נמקו תשובתכם ואם כן הראו כיצד.

הערה: תשובות ללא הסבר/מעקב לא יזכו בנקודות.

בהצלחה (:

1c) ① ~~First~~ לא נכון כי מחלקה Second מחלקה - ע"פ בע"פ

2) ② ~~Second~~ לא נכון, מחלקה Second יכולה לשמש כ- First ע"י ירושה

3) ~~First~~ כי יש שם פרטי double someVal

4) ~~Second~~ נכון - מחלקה יורשת את כל התכונות Second extends First

5) ~~First~~ לא נכון כי Second היא מחלקה פרטית Object N First יורשת

10
(1.1)

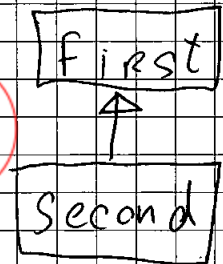
יפה

2) public Second(String blabla, int value) {
~~super();~~ someValue צריך להיות מחצית מערכו של value


this.blabla = blabla;
this.value = value;

2
5

3
(1.2)



בנאי כפ"ה אב"ד ע"ש, כי אצל מחלקה - ע"פ
e' בנאי ר"ק. public First() {
שם - זרק תחנה, ע"ש בנאי

 -2 ✓
(1.3)

הסבר שגוי

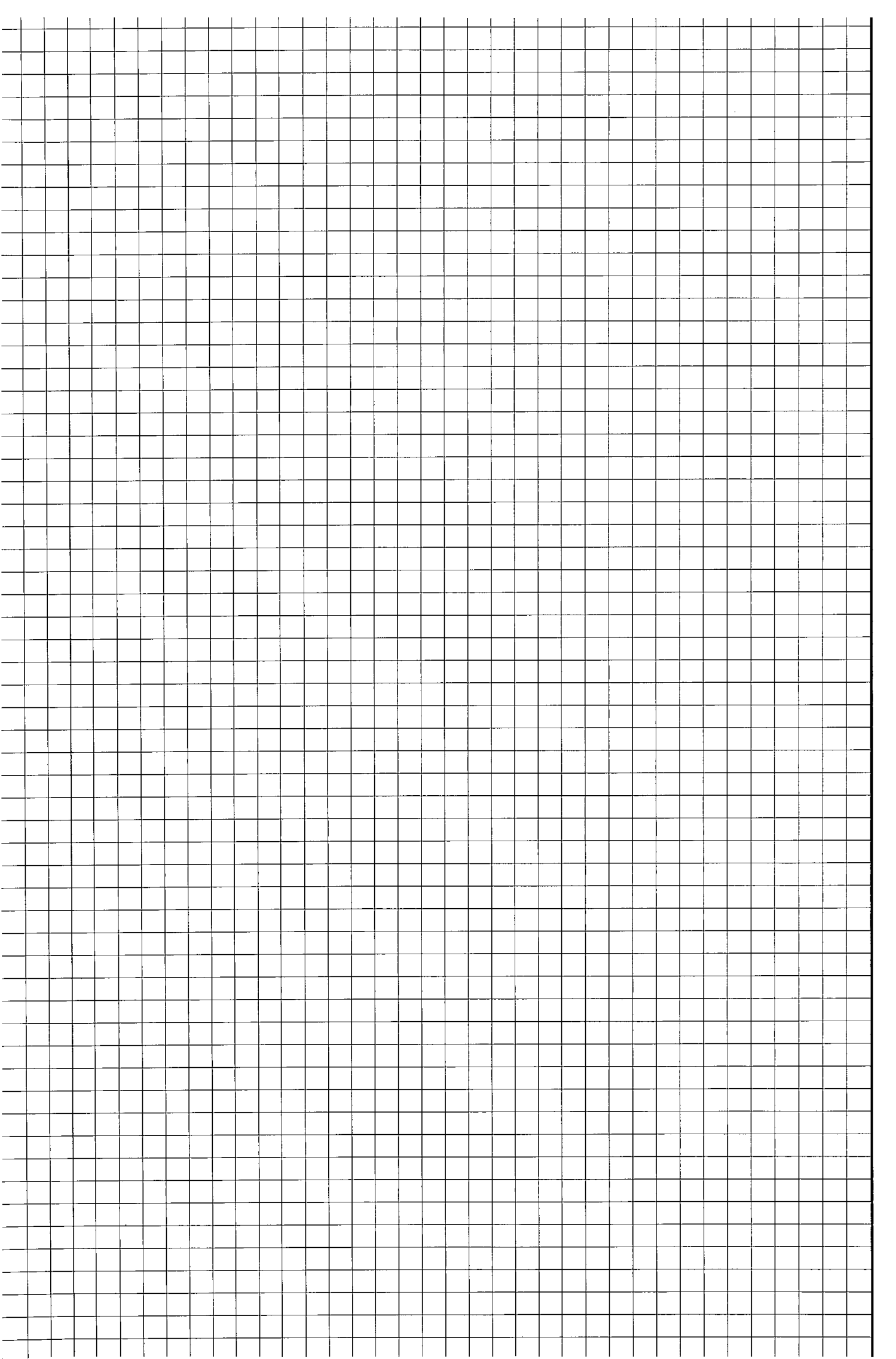


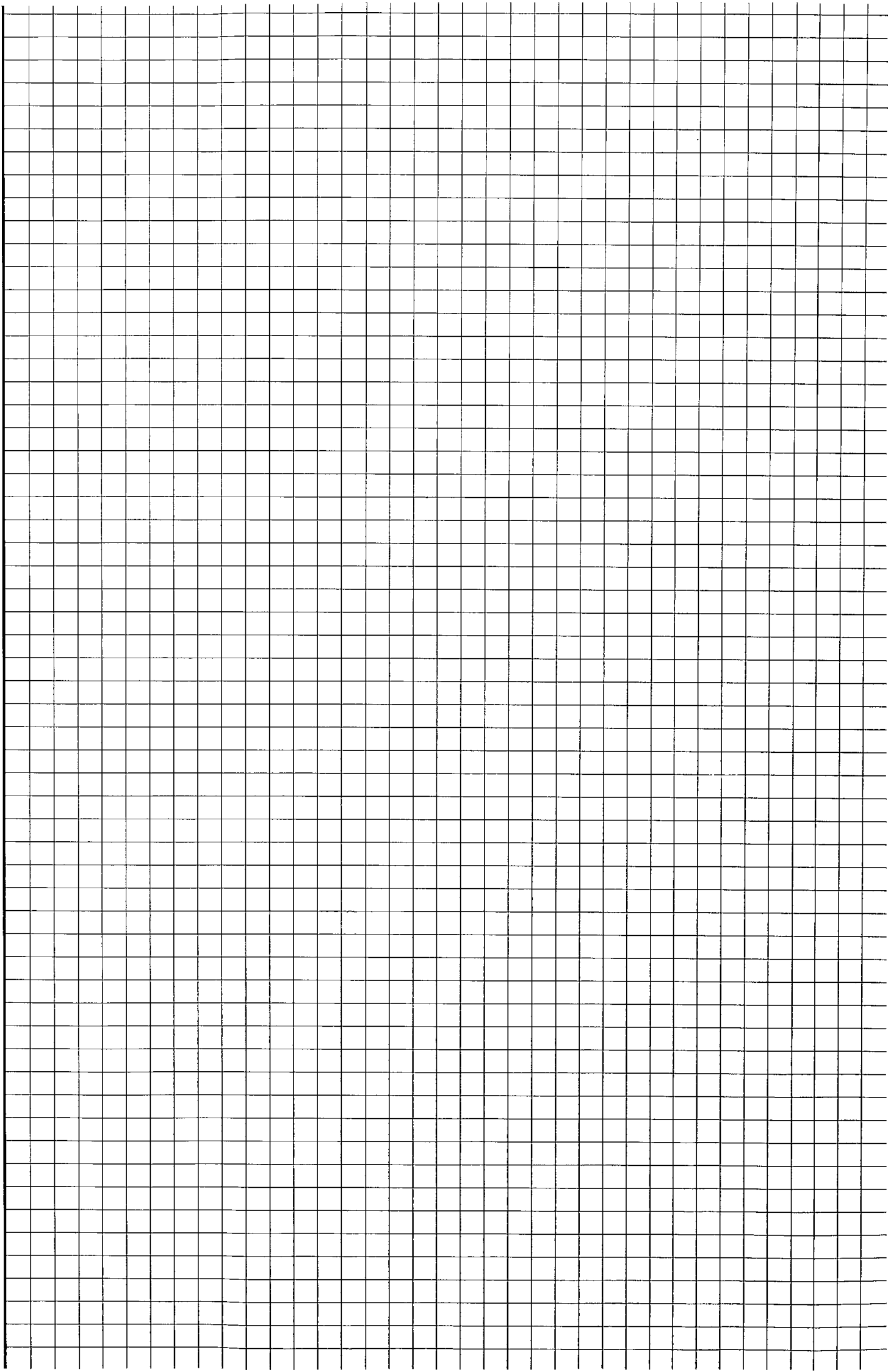
לא נכון



✓

למה אי אפשר?





✓ 1) class Goblin extends Card {
 public Goblin(int id, int power, string color,
 int tools) {
 super(id, power, color);
 this.tools = tools;
 }
 @Override
 public String toString() {
 ✓ return toString() + "Count tools: " + tools;
 }
 }
 public void

✓ 2) class Dragon extends Card {
 public Dragon(int id, int power, string color,
 int wingLen) {
 super(id, power, color);
 this.wingLen = wingLen;
 }
 @Override
 public String toString() {
 ✓ return toString() + "wingLen" + wingLen;
 }
 }
 public void

3) class Game {
 private static final int MAX_CARDS = 10;
 private Card[] cards;
 private int lastPosition;
 public Game() {
 cards = new Card[MAX_CARDS];
 lastPosition = 0;
 }
 }

public String toString() {

~~return~~ "last position is " + lastPosition;

לא נכון, חסרה לולאה לחדפסת פרטי הקלפים

-1
(2.1)

3
(2.1)

Goblin

2

2

1) public int calcForce() {
 return power * ^{.this}coals;
}

-1
(2.2)

יש להשתמש ב super.calcForce

2) Dragon

public int calcForce() {
 return power * ^{this}wingLen;
}

-1
(2.2)

יש להשתמש ב super.calcForce

6
(2.2)

1) public void removeCard(int id) {

2

2

for (int i=0; i < game.length-1; i++) {

if (card.getId() == id) {

~~for~~ for (int j = i; j < cards.length-1; j++) {

✓

cards[j] = cards[j+1]

5
(2.3.1)

cards[MAX_SIZE-1] = cards[~~last~~lastPosition];

}

~~Force of card~~

2)

public int calcTotalForce(String c) {

int totFe = 0;

~~for each card in cards~~

~~card totFe += card~~

~~for (int i = 0; i < Game.length; i++)~~

totFe += card[i].calcForce;

}

7 01'0

2) public int calcTotalForce(

⤴ 2

(String c){

int totFe = 0;

FOR (int i=0; ~~while~~ i < cards.length-1; i++) {

IF (cards[i].getColor().equals(c)) {

totFe += ⁽¹⁾ ~~this~~ cards[i].calcForce();

}

}

return totFe;

}

6
(2.3.2)

3) public void avgWingAndTotalTools() {

~~FOR (int i=0; i < cards.length-1; i++) {~~

int avgWing = 0, countWings = 0;

int totalTools = 0; ~~while~~

FOR (int i=0; i < cards.length-1; i++) {

IF (cards[i] instanceof Dragon) {

countWings++; avgWing += cards[i].^{get}WingLength();

IF (cards[i] instanceof Goblin) {

totalTools += cards[i].getTools();

}

}

-2
(2.3.3)

חסרה חזרה מסה

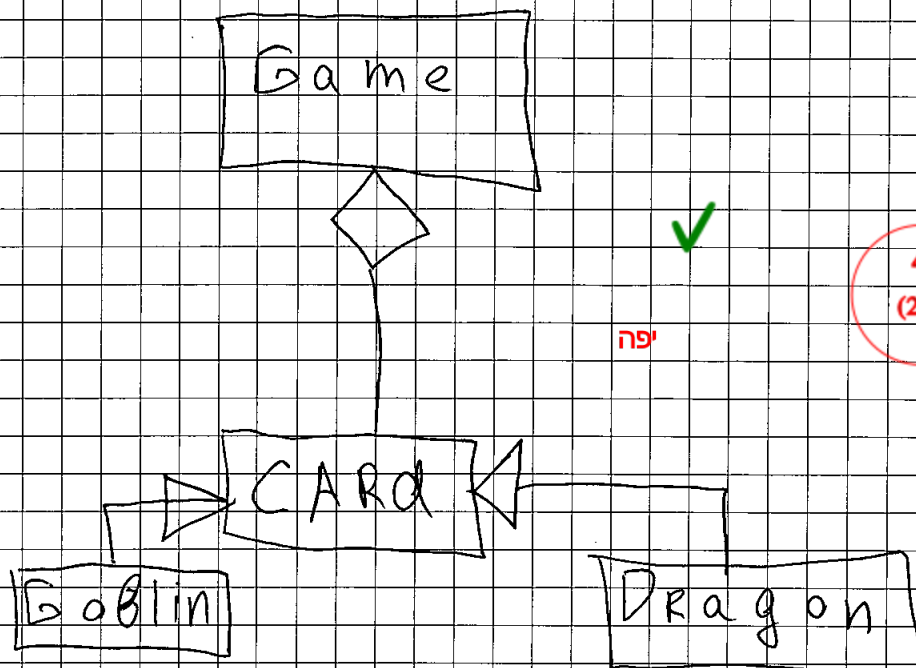
avgWing = avgWing / countWings;

System.out.println("avgWing = " + avgWing +

" totalTools = " + totalTools);

}

6
(2.3.3)



TS	TA	TP
1	1	
2		1
3		1
4	1	
5		1

-1
(3.1)

Daria

I tay

לא מצוין static

Idan

Ben

Hila

0) "Daria falls ~~down the slope~~"

$tS++$; $tA++$; $tS=1$; $tA=1$;

1) "Itay performs an amazing trick",

$tS++$; $tP++$; $tS=2$; $tP=1$;

2) "Idan ... trick",

$tS++$; $tP++$; $tS=3$; $tP=2$

3) "Ben Falls"

$tS++$; $tA++$; $tS=4$; ~~trick~~ $tA=2$;

4) "Hila ... trick"

$tS++$; $tP++$; $tS=5$; $tP=3$;

-4

(3.1)

חסר פלט

Sk: je מ"מ מ"מ מ"מ מ"מ מ"מ מ"מ

א"כ א"כ א"כ א"כ א"כ א"כ

4

(3.2)

Sk: א"כ א"כ א"כ א"כ א"כ א"כ

מ"מ מ"מ מ"מ מ"מ מ"מ מ"מ

מ"מ מ"מ מ"מ מ"מ מ"מ מ"מ

מ"מ מ"מ מ"מ מ"מ מ"מ מ"מ

מ"מ מ"מ מ"מ מ"מ מ"מ מ"מ

perform() je override

"this a + ..."

"is a skiing down the slope"

like' ...

Sk: A ...

Sk: P ...

Sk: P ...

... =

[2] = new Sk: P ("Idan")

2

(3.5)

1

7

1

8

4

13

5

