

פרק 9

פונקציות בספריה Math

שימוש בפונקציות – לשם מה?

השימוש בפונקציות מסייע לבצע פעולות מורכבות בקלות וביעילות. בשפת Java קיימות ספריות המכילות אוספים של פונקציות מוכנות. הפונקציות בספריה הוגדרו ונכתבו מראש על ידי מתכנני השפה והמתכנת יכול להשתמש בהן ולשלב אותן בתוכניות שהוא כותב. הפונקציה היא למעשה תת-תכנית הנסותרת מעיני המשתמש ומיועדת לביצוע חישוב מסוים והחזרת הערך שלו. המשתמש צריך לדעת כיצד לזמן את הפונקציה ולא יכול לראות את ההגדרה הפנימית שלה.

הפונקציה מקבלת ערכים על-ידי פרמטרים, מבצעת חישובים, ומחזירה תוצאה. קיימות ספריות בנושאים שונים כגון: ספריות קלט ופלט, ספריית פונקציות מתמטיות, פונקציות ספריה לטיפול בחלונות או בממשקי אינטרנט. בפרק זה נתמקד בספריה של פונקציות **מתמטיות** שכיחות המוגדרות בספריה Math. למשל: חישוב שורש ריבועי, העלאה בחזקה, חישוב ערך מוחלט (ראה טבלה בעמוד 98).



הגדרה

מושגים חדשים

פרמטרים: משתנים המופיעים ב**כותרת הפונקציה** ומקבלים תוכן בעת זימון הפונקציה.
זימון פונקציה: קריאה לביצוע הפונקציה מתוך מחלקה. הזימון כולל את שם הפונקציה ואת הערכים של הפרמטרים עבורם הפונקציה צריכה להתבצע. הזימון יכול להופיע בתוך משפט השמה, פלט או ביטוי.
ערכי פרמטרים: הערכים האמיתיים שמופיעים ב**זימון הפונקציה** ועבורם היא מתבצעת.

שים ♥: בפרק זה נלמד להשתמש בפונקציות ספריה מוכנות. בפרק 13 נלמד לבנות ולהגדיר פעולות בהן ניתן יהיה להשתמש בדומה לשימוש בפונקציות הקיימות בספריות הבנויות של השפה.



.....

.....

.....

.....

זימון פונקציה מתוך הספרייה Math



מבנה חלקי של זימון פונקציה

(רשימת ערכים) שם הפונקציה.Math

הסבר על מרכיבי ההוראה

זימון פונקציה בכל שפת תכנות מתבצע באמצעות ציון שם הפונקציה ובתוך סוגריים רשימת הערכים עליהם היא פועלת:

- **שם הפונקציה**: שם הפונקציה מעיד בדרך כלל על סוג הפעולה אותה היא מבצעת. למשל $abs(-8) \leftarrow 8$ הפונקציה מחשבת את ערכו המוחלט (absolute) של המספר -8.
- **הערכים המועברים לפונקציה**: ערכים שהם הקלט לפונקציה ועליהם היא פועלת. הערך יכול להיות מועבר לפונקציה מתוך קבוע, משתנה או ביטוי חשבוני.

הסבר על מהלך השימוש בפונקצית ספרייה

זימון פונקציה הוא הקריאה לביצוע הפונקציה מתוך קטע תכנית כלשהו. הזימון יכול להיות מהפעולה הראשית (main) או מפעולה אחרת. הפונקציה מקבלת ערכים, מבצעת חישובים ומחזירה ערך לפעולה שזימנה אותה. מכיוון שהפונקציה מחזירה ערך לא מספיק לזמן אותה על ידי כתיבת שם הפונקציה ורשימת ערכים בלבד, אלא יש להורות מה לבצע עם הערך המתקבל, למשל, להדפיס את הערך או להשימו למשתנה.

קיימות ארבע דרכים עיקריות לזמן פונקציה:

דרך א: בהוראת השמה

זימון פונקציה יכול להתבצע תוך כדי השמת הערך המוחזר על ידי הפונקציה למשתנה אחר. טיפוס המשתנה חייב להיות זהה לטיפוס הפונקציה. דוגמה: `absolute = Math.abs(num);`
הסבר: תחילה יחושב ערכו המוחלט של המשתנה num על-ידי פונקצית הערך המוחלט הנמצאת בספרייה Math, והערך שיוחזר יושם במשתנה absolute.

דרך ב: בהוראת פלט

זימון פונקציה יכול להתבצע תוך כדי הדפסת פלט. דוגמה: `System.out.println(Math.abs(num));`
הסבר: תחילה יחושב ערכו המוחלט של המשתנה num, ואז הערך המוחזר יוצג כפלט. הערך המוחזר יהיה זהה לטיפוס המשתנה num.

דרך ג: בתוך ביטוי בוליאני

<pre>if (Math.sqrt(num) ==10.0) System.out.println ("special");</pre>	הסבר: תחילה יחושב השורש של המשתנה num ולאחר מכן תתבצע בדיקה האם הערך המוחזר שווה ל-10.0
---	---

דרך ד: בתוך ביטוי חשבוני

<pre>avg=(Math.abs(num1)+Math.abs(num2))/2;</pre>	הסבר: תחילה יחושב ערכו המוחלט של המשתנה num1, אחר כך יחושב ערכו המוחלט של המשתנה num2 ואז יחושב סכום של הערכים המוחלטים, יחולק ב-2 ויושם במשתנה avg.
---	--

כללים לשימוש בהוראה

- יש לדאוג כי טיפוס הערך המוחזר מהפונקציה יתאים לטיפוס המשתנה אליו הוא מושם.
- קיימות פונקציות שלהן פרמטר אחד, קיימות פונקציות עם מספר פרמטרים וקיימות פונקציות ללא פרמטרים.
- כאשר לפונקציה יש מספר פרמטרים, הפרמטרים מופרדים בפסיק (,).

פונקציות שכיחות בספריה *Math* ב-*Java*

כדי להשתמש בפונקציות המוגדרות בספריה *Math* יש לרשום לפני כל שימוש בפונקציה את שם הספריה. תבנית הזימון היא: שם_הפונקציה **Math.** (לדוגמה: `y=Math.abs(x);`)

זימון הפונקציה	פעולת הפונקציה	דוגמאות	טיפוס הפרמטרים	טיפוס הערך המוחזר
Math.abs(num) פונקצית ערך מוחלט	מחזירה את הערך המוחלט	<code>Math.abs(-12) → 12</code> <code>Math.abs(12) → 12</code>	int; long; float; double	בהתאם לטיפוס הערך המועבר
Math.pow(num1,num2) פונקצית חזקה	מחזירה את $num1^{num2}$	<code>Math.pow(2, 3) → 8.0</code> <code>Math.pow(-4, 2) → 16.0</code> <code>Math.pow(2.5, 2) → 6.25</code> <code>Math.pow(2, 2.5) → 5.66</code>	int; long; float; double	ממשי - double
Math.sqrt(num) פונקצית שורש	מחזירה את השורש החיובי	<code>Math.sqrt(16) → 4.0</code>	int; long; float; double	ממשי - double
Math.round(num) פונקצית עיגול	מחזירה את עיגול המספר num. שבר מ-0.5 ומעלה מעביר לשלם הבא.	<code>Math.round(15.2) → 15</code> <code>Math.round(0.78) → 1</code> <code>Math.round(-8.9) → -9</code> <code>Math.round(-8.2) → -8</code>	float; double	שלם <code>round(float) → int</code> <code>round(double) → long</code>
Math.floor(num) פונקצית השלם הקטן	מחזירה את הערך השלם (מטיפוס double !!!) הגדול ביותר שקטן מ-num.	<code>Math.floor(15.2) → 15.0</code> <code>Math.floor(0.78) → 0.0</code> <code>Math.floor(-8.9) → -9.0</code> <code>Math.floor(-8.2) → -9.0</code>	int; long; float; double	ממשי - double
Math.min(a,b)	מחזירה את המספר המינימלי בין שני הפרמטרים	<code>Math.min(1,3) → 1</code> <code>Math.min(3.5,2.8) → 2.8</code>	int; long; float; double	בהתאם לטיפוס הערך המועבר
Math.max(a,b)	מחזירה את המספר המקסימלי בין שני הפרמטרים	<code>Math.max(1,3) → 3</code> <code>Math.max(3.5,2.8) → 3.5</code>	int; long; float; double	בהתאם לטיפוס הערך המועבר
Math.random()	מחזירה מספר אקראי ממשי גדול או שווה לאפס וקטן מ-1	<code>Math.random() →</code> אחד מן המספרים הממשיים בתחום 0..1 לא כולל 1		ממשי - double

שים ♥ : בספריה Math מוגדרות פעולות רבות נוספות. ניתן לעיין בתיעוד המחלקה באתר של חברת sun : <http://java.sun.com/j2se/1.3/docs/api/java/lang/Math.html>

שימוש בקבועים

בספריה Math קיימים קבועים מתמטיים : PI, E.
 השימוש בקבוע בתכנית נעשה על-ידי אחת מהאפשרויות הבאות :
 אפשרות א : זימון מהספריה Math בדומה לזימון פונקציה. המבנה : שם הקבוע.Math למשל : Math.PI
 אפשרות ב : הגדרה עצמאית של קבוע שמופיעה לרוב לפני הכותרת של הפעולה הראשית.
`static final int N=8;`

שימוש בקבועים יורחב בחלק ב של הספר החל מהפרק הראשון : מערכים.
 שים ♥ : מקובל להציג קבוע **באותיות גדולות** כדי להבדילו ממשתנה.

הפונקציה *random* – הגרלת מספר (יצירת מספרים אקראיים)

לעיתים בפתרון בעיה יש צורך בהגרלת מספרים. שפת התכנות Java מאפשרת לקבל מספרים אקראיים בעזרת הפונקציה random.
 זימון הפונקציה Math.random() מחזיר מספר אקראי ממשי גדול או שווה ל-0 וקטן מ-1.
כדי לקבל מספרים שלמים יש להפעיל פעולות חשבוניות על הערך האקראי שמחזירה הפונקציה Math.random().

הגרלת מספר שלם בטווח 0-5 :

<pre>int n; n = (int)(Math.random()*6);</pre> <div style="text-align: center; margin-top: 10px;"> </div>	<p>Math.random() מחזירה מספר ממשי מטיפוס double בתחום 0-1 כולל 0 ולא כולל 1. כלומר : $0 \leq \text{Math.random}() < 1$ כאשר נכפיל ב-6 נקבל מספר ממשי בתחום : $0 \leq \text{math.random}() < 6$ כאשר נמיר למספר שלם על-ידי (int) נקבל שלם בתחום : $0 \leq (\text{int})(\text{math.random}()*6) \leq 5$</p>
--	---

הגרלת מספר שלם בטווח 1-6 :

<pre>int n; n = (int)(Math.random()*6)+1;</pre> <div style="text-align: center; margin-top: 10px;"> </div>
--

הגרלת מספר שלם בטווח 12-45 :

<pre>int n; n = 12+(int)(Math.random()*34)</pre> <div style="text-align: center; margin-top: 10px;"> </div>

הגרלת מספר שלם בטווח כלשהו x-y :

<pre>int n; n = x + (int)(Math.random()*(y-x+1));</pre>	<p>כאשר רוצים להגריל מספר שלם בטווח שבין שני משתנים שלמים x, y יש לחשב את גודל טווח המספרים על פי הנוסחה : $y-x+1$. את הערך המוגרל כופלים בגודל טווח המספרים שחושב, מקצצים אותו כך שיהיה מספר שלם, ומוסיפים לו את הערך הנמוך של התחום x, כדי לקבל מספרים בטווח הנדרש.</p>
---	--

בעיות פתורות על-פי השלבים לפיתוח אלגוריתם



דוגמה
פתורה

דוגמה פתורה 1: הגרלה... *

כתוב קטע תכנית המדמה הטלת מטבע ומדפיס 1 אם התקבל עץ או 2 אם התקבל פלי.

```
n = (int)(Math.random()*2)+1;
System.out.println (n);
```

דוגמה פתורה 2: הגרלת ממשיים **



דוגמה
פתורה

פתח ויישם אלגוריתם המגריל שלושה מספרים ממשיים בטווח 20-40, ומציג כפלט:

- את המספר הגדול ביותר.
- את השורש הריבועי המעוגל של כל אחד מהמספרים שהוגרלו.
- הודעה מתאימה אם יש לפחות שני מספרים שהשורש המעוגל שלהם שווה.

פתרון הבעיה על-פי השלבים לפיתוח אלגוריתם

ניתוח קלט – פלט:

מספרים אקראיים: 20.25, 26.8, 29.777

פלט: המספר הגדול: 29.777

השורשים המעוגלים: $20.25 \rightarrow 5$ $26.8 \rightarrow 5$ $29.777 \rightarrow 5$

יש לפחות 2 מספרים שהשורש המעוגל שלהם שווה

טבלת משתנים

שם המשתנה	טיפוס המשתנה	הסבר/תפקיד	תחום ערכים
n1, n2, n3	ממשי	מספרים אקראיים	20-40
max1	ממשי	המספר הגדול מבין n1, n2	20-40
max2	ממשי	המספר הגדול מבין n1, n2, n3	20-40
sqrt1, sqrt2, sqrt3	שלם	שורשים מעוגלים	4-6

אלגוריתם	תת-משימות
<ol style="list-style-type: none"> הגרל מספר בטווח 20-40 $n1 \leftarrow$ הגרל מספר בטווח 20-40 $n2 \leftarrow$ הגרל מספר בטווח 20-40 $n3 \leftarrow$ 	הגרלת 3 מספרים ממשיים
<ol style="list-style-type: none"> מצא מקסימלי $\max1 \leftarrow (n1, n2)$ מצא מקסימלי $\max2 \leftarrow (\max1, n3)$ הצג כפלט את המקסימלי $\max2$ 	מציאת מקסימלי והצגתו כפלט
<ol style="list-style-type: none"> חשב שורש מעוגל של n1 והשם ב-sqrt1 חשב שורש מעוגל של n2 והשם ב-sqrt2 חשב שורש מעוגל של n3 והשם ב-sqrt3 	חישוב השורשים המעוגלים של שלושת המספרים
<ol style="list-style-type: none"> אם $\sqrt{1} = \sqrt{2}$ או $\sqrt{2} = \sqrt{3}$ או $\sqrt{1} = \sqrt{3}$ אז הצג כפלט "יש לפחות שני מספרים שווה" אחרת הצג כפלט "לכל המספרים שורש מעוגל שונה" 	הדפסת הודעה מתאימה אם יש לפחות שני מספרים שהשורש המעוגל שלהם שווה

פתרון ב- Java

```
import java.util.*;
class Solv2RandomNumbers
{
static Scanner reader = new Scanner(System.in);
public static void main(String[] args)
{
    double n, n1, n2, n3, max1, max2;
    long sqrt1, sqrt2, sqrt3;
    n = Math.floor(Math.random()*21);
    n1 = n+20+Math.random();
    n = Math.floor(Math.random()*21);
    n2 = n+20+Math.random();
    n = Math.floor(Math.random()*21);
    n3 = n+20+Math.random();
    System.out.println ("The numbers are: " + n1 + " " + n2 + " " + n3);
    max1= Math.max(n1,n2);
    max2 = Math.max(max1,n3);
    System.out.println ("The biggest number is" + max2);
    sqrt1 = Math.round(Math.sqrt(n1));
    sqrt2 = Math.round(Math.sqrt(n2));
    sqrt3 = Math.round(Math.sqrt(n3));
    if (sqrt1==sqrt2 || sqrt2==sqrt3 || sqrt3==sqrt1)
        System.out.println ("At least 2 numbers have the same round sqrt");
    else
        System.out.println ("All numbers have different round sqrt");
}
}
```

טבלת מעקב - הרצה עבור קלטים מייצגים

n	n1	n2	n3	max1	max2	sqrt1	sqrt2	sqrt3	פלט
17									
	37.25								
3									
		23.84							
9									
			29.77						
									The numbers are: 37.25 23.84 29.77
				37.25	37.25				The biggest number is 37.25
						6	4	5	
									All numbers have different round sqrt

בעזרת כתיבה פונקציונאלית במקום סעיפי האלגוריתם 4-5 ניתן היה לרשום הוראה אחת:

`max3 = Math.max(Math.max(n1,n2),n3);`

תחילה מחושב הערך המקסימלי בין `n1`, `n2` והערך המוחזר היה נשלח לבדיקת המקסימלי השני מול `n3`. וכך היה מתקבל הערך המקסימלי בין 3 המשתנים.

שים



תרגילים

תרגיל 1: תרגם לקוד בשפת תכנות ★

כתוב תכנית

השלם את ההוראות המתאימות לכל אחד מהקטעים הבאים :

קטע 1: שורש ריבועי

נתון המשתנה num ובו מספר שלם.
כתוב הוראות המדפיסות את ריבוע המספר ואת השורש הריבועי שלו.

קטע 2: אלכסון מלבן

נתונים אורכי צלעות המלבן במשתנים a, b.
כתוב הוראה להשמת אורך האלכסון של המלבן ב-c
(את האלכסון יש לחשב באמצעות משפט פיתגורס: $c = \sqrt{a^2 + b^2}$).

תרגיל 2: ריבוע קסם ★

נתון התשבץ הבא :

3	2	1
6	5	4
9	8	7

רשום בטבלה את הפלט המתקבל מכל אחת מההוראות ב- בסעיפים הבאים (1-9) בהתאמה.

	Java
1.	System.out.println (Math.sqrt(16));
2.	System.out.println (Math.abs(9-18));
3.	System.out.println ((int) (2.94));
4.	System.out.println (Math.sqrt((15%5)*5+9));
5.	System.out.println (Math.sqrt(Math.pow(3,2)+Math.pow(4,2)));
6.	System.out.println (Math.sqrt((int) (7.9)*7));
7.	System.out.println (Math.pow(388/100,2)-1);
8.	System.out.println (Math.abs(-15)-(int)(14.5));
9.	System.out.println (Math.max(36/10%2 ,6));

השלם :

סכום המספרים בכל השורות : _____ סכום המספרים בכל העמודות : _____
סכום המספרים באלכסון הראשי : _____ סכום המספרים באלכסון המשני : _____



★ ★ תרגיל 3: חקר פונקציות

- א. מהי המטרה של כל אחת מהקודים א ו-ב ?
ב. השתמש בטבלת מעקב ובדוק את הפלט של שני הקטעים (לאחר מכן הרץ במחשב לבדיקה).

```
import java.util.*;
class Functions1
{
    static Scanner reader = new Scanner(System.in);
    public static void main(String[] args)
    {
        double a, b, root, fracA, fracB, minimum, maximum, s, power;
        int intA, intB;
        System.out.println ("Enter two real numbers");
        a = reader.nextDouble();
        b = reader.nextDouble();
        intA = (int)(a);
        intB = (int)(b);
        root = Math.sqrt(a+b);
        fracA = a-intA;
        fracB = b-intB;
        minimum = Math.min(a,b);
        maximum = Math.max(a,b);
        power = Math.pow(intA,intB);
        s = Math.sin(a);
        System.out.println ("The float numbers are:" + a + " "+b);
        System.out.println ("The integer numbers are:"+ intA + " " + intB);
        System.out.println ("The root of a + b is:" + root);
        System.out.println ("fracA=" + fracA);
        System.out.println ("fracB=" + fracB);
        System.out.println ("min(a,b)=" + minimum);
        System.out.println ("max(a,b)=" + maximum);
        System.out.println ("power(intA,intB)=" + power);
        System.out.println ("pi="+Math.PI);
        System.out.println ("sina=" + s);
    }
}
```

ראה הסבר על המרת טיפוסים נתונים בעמוד 42

כאן יודפס הקבוע PI.

```
import java.util.*;
class Functions2
{
    static Scanner reader = new Scanner(System.in);
    public static void main(String[] args)
    {
        double num1;    double num2;
        num1 = Math.random()*100;
        num2 = Math.sqrt(Math.random()*num1);
        System.out.println ("num1=" + num1);
        System.out.println ("num2=" + num2);
    }
}
```


בתרגילים הבאים כתוב קטעי תכנית בהנחה שהפונקציות הללו לא קיימות בשפה (שים לב: בנייה של פונקציות חדשות תלמד בפרק 13, כאן נתייחס רק להוראות הדרושות).

★ תרגיל 4: ערך מוחלט של מספר

כתוב סדרת הוראות לחישוב והדפסת הערך המוחלט של מספר שנקלט.

הנחה: אין בספרייה Math פונקציה abs.

הפונקציה abs מקבלת מספר שלם ומחזירה כפלט את ערכו המוחלט.
דוגמאות:
 $\text{abs}(7) \rightarrow 7$
 $\text{abs}(-7) \rightarrow 7$

★ תרגיל 5: ערך מינימלי

כתוב סדרת הוראות לחישוב והדפסת הערך הקטן בין שני נתוני קלט.

הנחה: אין בספרייה Math פונקציה min.

הפונקציה min מקבלת שני מספרים ומחזירה את המספר המינימלי בין שני המספרים. דוגמאות:
 $\text{min}(1,3) \rightarrow 1$ $\text{min}(3.5,2.8) \rightarrow 2.8$

★ תרגיל 6: הערך השלם של מספר ממשי

כתוב סדרת הוראות לחישוב והדפסת הערך השלם של מספר ממשי.

אם הייתה פונקציה intNum:
הפונקציה intNum מקבלת מספר שלם או ממשי ומחזירה את החלק השלם.
דוגמאות:
 $\text{intNum}(578.99) \rightarrow 578$

★ תרגיל 7: הערך השבור של מספר ממשי

כתוב סדרת הוראות שתקבל מספר ממשי ותחזיר את הערך השבור (החלק אחרי נקודה).

אם הייתה פונקציה fracNum:
הפונקציה fracNum מקבלת מספר ממשי ומחזירה את החלק השבור.
דוגמאות:
 $\text{fracNum}(5.99) \rightarrow 0.99$

★ ★ תרגיל 8: עיגול של מספר

כתוב סדרת הוראות לחישוב והדפסת התוצאה המעוגלת של מספר ממשי שהתקבל כקלט.

הנחה: אין בספרייה Math פונקציה round.

הפונקציה round מקבלת מספר ממשי ומחזירה את ערכו המעוגל.
דוגמאות:
 $\text{round}(14.2) \rightarrow 14$
 $\text{round}(14.8) \rightarrow 15$

★ תרגיל 9: שטח והיקף של מעגל

פתח ויישם אלגוריתם המקבל כקלט רדיוס של מעגל, מחשב ומדפיס את:

א. קוטר המעגל.

ב. היקף המעגל.

ג. שטח המעגל.

★ תרגיל 10: היתר במשולש ישר זווית

פתח ויישם אלגוריתם המקבל כקלט שני מספרים, המייצגים את אורכי הניצבים במשולש ישר זווית. האלגוריתם יחשב וידפיס את אורכו של היתר.

★ תרגיל 11: ממוצע מספרים אקראיים

פתח ויישם אלגוריתם המגריל שלושה שברים עשרוניים בין 0 ל-1 (לא כולל 1) למשתנים $n1, n2, n3$, ומדפיס את ממוצע המספרים שהתקבלו בהגרלה.

★ תרגיל 12: שטח והיקף של משולש

פתח ויישם אלגוריתם המקבל כקלט 3 אורכים של צלעות משולש (הנח כי הקלט חוקי, כלומר הצלעות מהוות צלעות של משולש). על האלגוריתם להציג כפלט את שטח המשולש ואת היקפו.

ניתן לחשב שטח של משולש על-פי נוסחת הרון:
$$שטח = \sqrt{P * (P - A) * (P - B) * (P - C)}$$
 חצי היקף $p = a, b, c =$ אורכי הצלעות,

★★ תרגיל 13: החלק השלם והחלק הממשי

פתח ויישם אלגוריתם המקבל כקלט סכום כסף כמספר ממשי.

על האלגוריתם להציג כפלט הודעה המפרטת את מספר השקלים ואת מספר האגורות בנפרד, באופן הבא: הסכום הוא X שקלים ו-Y אגורות.

דוגמה לקלט: 25.15 הפלט: הסכום הוא 25 שקלים ו-15 אגורות.

★★ תרגיל 14: מספר משולש

פתח ויישם אלגוריתם המקבל כקלט מספר תלת ספרתי ומציג כפלט הודעה האם המספר הוא "מספר משולש".

מספר משולש הוא מספר שמקיים את התכונה הבאה: אם מעלים כל אחת מספרותיו בחזקת 3 וסוכמים את התוצאות מתקבל המספר עצמו. דוגמה ל"מספר משולש": $153 = 1^3 + 5^3 + 3^3$

★★★ תרגיל 15: דיוק עשרוני

פתח ויישם אלגוריתם הקולט מספר עשרוני עם דיוק כלשהו אחרי הנקודה (לכל היותר 3 ספרות אחרי הנקודה) ומציג כפלט את המספר המעוגל עם דיוק של ספרה אחת אחרי הנקודה.

למשל, אם המספר שנקלט הוא 35.791 הפלט הוא: 35.8.



תרגיל 16: עיגול ציונים ★ ★ ★

בקורס שנתי במכללה מקבלים הסטודנטים ציון ביניים בסוף סמסטר א וציון סופי בסוף השנה. הציון הוא מספר ממשי בטווח 4-10. המרצה למתמטיקה מעגלת את הציון בסוף סמסטר א אך שומרת את החלק השבור לציון הסופי. פתח ויישם אלגוריתם המקבל כקלט ציון ביניים של תלמיד (מספר עשרוני) ומציג כפלט את הציון המעוגל ואת החלק השבור שנשמר לסוף השנה (חיובי או שלילי בהתאם למה שהתקבל).
 דוגמאות: קלט: 96.4 פלט: הציון המעוגל 96, החלק שנשמר לציון בסוף השנה +0.4
קלט: 67.8 פלט: הציון המעוגל 68, החלק שנשמר לסוף שנה: -0.2

מהו הפלט?

תרגיל 17: מהו הפלט? מה מטרת התכנית? ★

נתון קטע הקוד הבא:

```
public static void main(String[] args)
{
    int h, result;
    h = reader.nextInt();
    result = h/Math.abs(h);
    System.out.println ("Result=" + result);
}
```

- א. מה מטרת הקטע (בנה לפחות 2 טבלאות מעקב עבור קלטים מייצגים (!!!).
- ב. שפר את התכנית: רשום שמות משמעותיים למשתנים, הוסף הודעה מתאימה למשתמש לפני הוראת הקלט, והרץ את הקוד על המחשב.
- ג. מה הם הפלטים האפשריים?

תרגילים: מספרים אקראיים ★

תרגיל 18: ההוראה המתאימה

בכל שורה בטבלה מופיעה רשימה של מספרים אקראיים שהתקבלו כפלט של סדרת הוראות. רשום סדרת הוראות מתאימה שתאפשר את הפלט.

ההוראות	תחום המספרים	המספרים האקראיים המתקבלים כפלט
1.		0, 1, 2, 3,, 100
2.		4, 5, 6, 7, 8, 9, 10
3.		100, 101, 102, 103
4.		-2, -1, 0, 1, 2, 3
5.		-100, -99, -98,, 0

תרגיל 19: זרוק קובייה... *

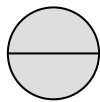
1. פתח ויישם אלגוריתם המגריל מספר שיכול להתקבל כתוצאה מזריקת קובייה ומציג כפלט:
 - א. את המספר שהתקבל.
 - ב. את ריבוע המספר שהתקבל.
 - ג. את הערך המוחלט של המרחק שלו מממוצע המספרים שמתקבלים בהטלת קובייה (1..6).
2. הרץ את האלגוריתם לפחות 3 פעמים ורשום את התוצאות שקיבלת בטבלה שלהלן:

הרצה ראשונה	הרצה שנייה	הרצה שלישית

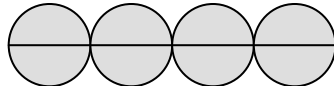


רגע של גרפיקה

תרגיל 20: קו ועיגול **



- א. כתוב מחלקה המגרילה מספר בטווח 30-50 המתאר את קוטרו של עיגול, ומציגה עיגול שבתוכו מוצג קוטרו כמתואר בשרטוט.
- ב. הגדל את הקו פי 4, והוסף לאורכו עוד 3 עיגולים.



תרגיל 21: הכדור הקופץ ***

במשחק מחשב ניתן "להקפיץ" כדור מתחתית המסך עד לנקודת גובה שבוחר השחקן. לאחר שהכדור "חוזר" לתחתית המסך הוא קופץ שוב. גובה הקפיצה השנייה שווה לשורש גובה הקפיצה הראשונה. גובה שאינו מספר שלם – מעוגל.

בנה מחלקה המקבלת כקלט את הגובה שבוחר השחקן. יש להציג כפלט את הכדור במיקום הקפיצה הראשונה, ולאחר מכן להזיז אותו למיקום הקפיצה השנייה.