

מטלת מנחה (ממ"ן) 11

הקורס: 20441 - מבוא למדעי המחשב ושפת Java

נושאי המטלה: יסודות השפה

חומר הלימוד למטלה: יחידות 1-2

משקל המטלה: 1 נקודה

מספר השאלות: 2

מועד אחרון להגשה: 5.4.2025

סמסטר: 2025

(ת)

שימו לב:

- יש להקפיד על שמות המחלקות בדיוק כמו שנכתבו.
- יש לתעד את התכניות בתיעוד פנימי באנגלית בלבד (בתחילת התכנית התיעוד מסביר מה מבצעת התכנית באופן כללי ובמהלך התכניות התיעוד מסביר את הקוד).
- אין להוסיף שיטות מעבר לאלה הנדרשות במטלה במפורש.
- אין להשתמש בחומר מתקדם ובפרט לא בלולאות.
- יש להשתמש בקבועים היכן שאפשר.
- יש להקפיד על הזחה (אינדנטציה - עימוד) נכונה, ועל שמות משתנים בעלי משמעות (באנגלית) ולפי המוסכמות בקורס.
- יש להקפיד על פורמט הפלט בדיוק כפי שמצוין בשאלה: איות נכון, אותיות גדולות וקטנות, רווחים, וכו'.
- באתר הקורס תוכלו למצוא קובץ הנחיה לפתרון המטלות התכנותיות. כדאי מאד לעיין בו ולפעול לפיו. הקובץ נמצא בלשונית "מדריכי עזר" שנמצאת ביחידה 1, והוא נקרא הנחיות לכתיבת תכניות ומטלות בקורס.
- הגשת המטלה נעשית אך ורק בעזרת מערכת המטלות המקוונת שבאתר הקורס.
- אל תשכחו לשמור את מספר האסמכתא שתקבלו מהמערכת לאחר ההגשה.

שאלה 1 (50%)

כתבו תכנית לחישוב מרחק בין שתי רכבות בסיום נסיעתן.
שתי הרכבות יוצאות בו-זמנית מאותו מקום לאותו כיוון.
התכנית תקרא מהקלט שהכניס המשתמש ארבעה מספרים שלמים. בתחילה את המהירות של הרכבת הראשונה (בקמ"ש כ-`int`) ואת זמן נסיעתה (בדקות כ-`int`), ואחר כך את המהירות של הרכבת השנייה (בקמ"ש כ-`int`) ואת זמן נסיעתה (בדקות כ-`int`).
התכנית תחשב את המרחק בין שתי הרכבות אחרי ששתי הרכבות סיימו את הנסיעות, ותדפיס על הפלט את המרחק הזה.
שימו לב שהתוצאה היא לא בהכרח מספר שלם (אבל היא חייבת להיות חיובית).

תזכורת מתמטית –

$$\text{מהירות} * \text{זמן} = \text{מרחק}$$

לדוגמא,

אם רכבת A נוסעת במהירות 90 קמ"ש במשך 30 דקות, היא עוצרת אחרי 45 ק"מ.
אם רכבת B נוסעת במהירות 120 קמ"ש במשך 20 דקות, היא עוצרת אחרי 40 ק"מ.
לאחר ששתי הרכבות סיימו את נסיעותיהן, המרחק ביניהן הוא 5 ק"מ.

ההדפסה תהיה בפורמט הבא :

אם הקלט הוא (הנתונים של רכבות A ו- B שלעיל) :

90 30 120 20

אז הפלט צריך להיות כזה בדיוק :

The distance between the trains is 5.0 km.

לשם קריאה מהקלט השתמשו במחלקה Scanner.

כדי להשתמש בה צריך לכתוב בראשית התכנית את השורה

```
import java.util.Scanner;
```

אפשר למצוא את הממשק של המחלקה Scanner בפרק Interactive Programs מהספר Java

Software Solutions שנמצא לאחר הסרטון 2.4, שם מובאות חלק מהשיטות.

הסברים על המחלקה והשימוש בה אפשר למצוא באתר הקורס בתוך "יחידה 2" ב"מדריכי עזר

וקישורים" בקובץ "מדריך לעבודה עם המחלקה Scanner לקבלת קלט מהמשתמש".

כדי לחשב את הערך המוחלט של מספר כלשהו x , ניתן להשתמש בשיטה `Math.abs(x)`, שהיא שיטה של Java שנמצאת במחלקה `Math`. כדי להשתמש בה אין צורך לייבא אף מחלקה, אלא לקרוא לה בשמה המלא `Math.abs(x)` כאשר במקום הפרמטר x כותבים את הביטוי שאת הערך המוחלט שלו רוצים לקבל. הפרמטר x של השיטה הזו יכול להיות מטיפוס שלם (`int`) או ממשי (`double`). השיטה מחזירה אותו טיפוס של הפרמטר.

לעזרתכם, כתבנו כאן חלקים מהמחלקה. עליכם להשלים את החסר (גם את התיעוד החסר).

```
import java.util.Scanner;

public class Train
{
    public static void main (String [] args)
    {
        // כאן אפשר להכניס תיעוד המסביר על התכנית

        Scanner scan = new Scanner (System.in);

        System.out.println ("Please enter 4 integers ");

        System.out.println ("Please enter the speed of train 1:");

        int v1 = scan.nextInt();

        System.out.println ("Please enter the time of train 1:");

        int t1 = scan.nextInt();

        // כאן עליכם להמשיך ...

    } // end of method main
} //end of class Train
```

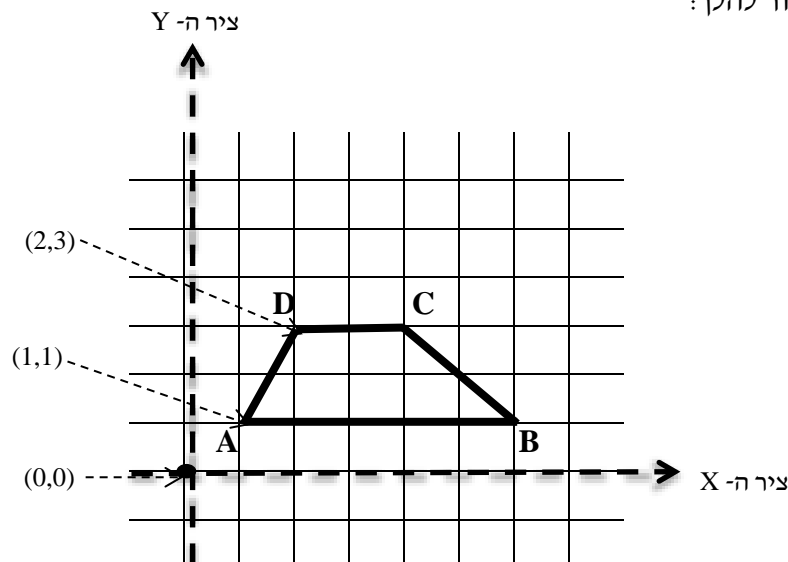
בשאלה זו אתם יכולים להניח שהקלט שניתן הוא תקין ושהוכנסו מספרים שלמים חיוביים ממש (לא כולל אפסים). אין צורך לבדוק זאת.

התכנית שכתבתם צריכה להיות במחלקה בשם `Train`. המחלקה `Train` תכיל שיטה אחת בלבד בשם `main`. אסור לכתוב שיטות נוספות.

שאלה 2 (50%)

כתבו תכנית לחשוב שטח והיקף של **טרפז** (trapezoid). **טרפז** הוא מרובע אשר לו שתי צלעות נגדיות מקבילות (אין שום תנאי על הצלעות האחרות). שימו לב שלפי הגדרת הויקיפדיה, הטרפז הוא מרובע בעל **לפחות** זוג אחד של צלעות נגדיות מקבילות. כך שגם מקבילית, מלבן וריבוע נחשבים טרפזים.

בשאלה זו נתייחס רק לטרפזים שהצלעות המקבילות שלו מקבילות גם לציר ה-X. ראו את האיור להלן:



תחילה עליכם לקלוט את נתוני הבסיסים של הטרפז. (הישרים בין A ל-B ובין C ל-D). כדי לעשות זאת עליכם לקלוט לכל בסיס את הנקודה השמאלית שלו ואת אורכו. כתבו את הקוד כך שהמשתמש יתבקש להזין פעם אחת את הקואורדינטות של הנקודה השמאלית של בסיס אחד וכן את אורך הבסיס ופעם שניה המשתמש יתבקש להזין את הקואורדינטות של הנקודה השמאלית של הבסיס השני וכן את אורך הבסיס. משתמש שירצה להזין את נתוני הטרפז שבאיור לעיל יצטרך להזין כך:

Please enter the left point coordinates of the base followed by its length:

1 1 5

Please enter the left point coordinates of the other base followed by its length:

2 3 2

כלומר, הנקודה השמאלית של בסיס אחד (A-B) היא (1,1) ואורכו הוא 5

הנקודה השמאלית של בסיס שני (D-C) היא (2,3) ואורכו הוא 2

הערה: שימו לב, שהמשפט "Please enter..." הכתוב כאן הוא מה שהתכנית אמורה להדפיס על הפלט כדי להסביר למשתמש אילו נתונים עליו להכניס. המשתמש אמור להכניס רק את המספרים הכתובים כאן: 1 1 5 ואחר כך את 2 3 2.

כתבו תכנית הקוראת מהמשתמש שישה מספרים שלמים (לתוך משתנים מטיפוס int). שני המספרים הראשונים יצינו קואורדינטת x ו- קואורדינטת y של הנקודה השמאלית של בסיס אחד של הטרפז; המספר השלישי יציין את אורך הבסיס הזה; שני המספרים הבאים יצינו קואורדינטת x ו- קואורדינטת y של הנקודה השמאלית של בסיס שני של הטרפז; המספר השלישי יציין את אורך הבסיס הזה. אין חשיבות לסדר בין הבסיסים.

שימו לב שהטרפז יכול להופיע בכל רביע במישור (כלומר, הקואורדינטות לא חייבות להיות מספרים חיוביים). כמו כן, אי אפשר להניח שהבסיס התחתון גדול או שונה באורכו מהעליון.

על תכניתכם לחשב ולהציג את שטח (area) והיקף (perimeter, circumference) הטרפז.

תזכורת מתמטית:

כדי לחשב מרחק בין שתי נקודות - (x_1, y_1) , (x_2, y_2) - השתמשו בנוסחה הבאה:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

על מנת לחשב שורש ריבועי של מספר, ניתן להשתמש בשיטה `Math.sqrt(x)`, שהיא שיטה של Java שנמצאת במחלקה `Math`. כדי להשתמש בה אין צורך לייבא אף מחלקה, אלא לקרוא לה בשמה המלא `Math.sqrt(x)` כאשר במקום הפרמטר x כותבים את הביטוי שממנו רוצים להוציא שורש ריבועי. הפרמטר x של השיטה הזו יכול להיות מטיפוס שלם (int) או ממשי (double). השיטה מחזירה מספר ממשי (גם אם השורש הריבועי של x הוא מספר שלם).

על מנת לחשב חזקה של מספר, ניתן להשתמש בשיטה `Math.pow(a, b)`, שהיא שיטה של Java שנמצאת במחלקה `Math`. כדי להשתמש בה אין צורך לייבא אף מחלקה, אלא לקרוא לה בשמה המלא `Math.pow(a, b)` כאשר במקום הפרמטר a כותבים את הביטוי שאותו רוצים להעלות בחזקה ובמקום הפרמטר b כותבים את החזקה. הפרמטרים a ו- b של השיטה הזו יכולים להיות מטיפוס שלם (int) או ממשי (double). השיטה מחזירה מספר ממשי (גם אם a בחזקת b הוא מספר שלם).

הנוסחה לחישוב שטח הטרפז היא:

$$\frac{height * (base1 + base2)}{2}$$

ההדפסה של הפלט חייבת להיות בדיוק לפי הפורמט הבא:

The area of the trapezoid is 7.0

The perimeter of the trapezoid is 12.0644

הערה: מה שכתוב כאן בגופן נטוי לא צריך להיות נטוי בפלט שלכם. כאן כתבנו זאת רק כדי להדגיש את המשתנים. כמובן שהנתונים כאן הם אלו שמתאימים לאיור לעיל. לכל טרפז יהיו תוצאות אחרות. גם רמת הדיוק יכולה להיות לפי מה שתדפיס התכנית - לדוגמא למי שיוצא 12.06449510224598 אין צורך ליפות את ההדפסה.

הנחות:

1. אתם יכולים להניח שהמשתמש אכן הכניס שישה מספרים שלמים ולא אותיות או תווים אחרים.
 2. אתם יכולים להניח שהמספרים שהוכנסו אכן מהווים קלט תקין לטרפז. שני המספרים הראשונים הם הנקודה השמאלית והמספר השלישי הוא האורך של הבסיס הראשון וכך גם לגבי הבסיס השני. אין ערבוב.
 3. אינכם יכולים להניח מי מבין הבסיסים יופיע ראשון. כלומר, יתכן והמשתמש יזין את נתוני הבסיס העליון קודם ויתכן שיזין את נתוני הבסיס התחתון קודם.
 4. אינכם יכולים להניח שהבסיס התחתון גדול או שונה באורכו מהעליון.
- לעזרתכם, כתבנו כאן חלקים מהמחלקה. עליכם להשלים את החסר (גם את התיעוד החסר).

```
import java.util.Scanner;

public class Trapezoid
{
    public static void main (String [] args)
    {
        // כאן אפשר להכניס תיעוד המסביר על התכנית

        Scanner scan = new Scanner (System.in);

        System.out.println("Please enter the left point
            coordinates of the base followed by its length:");

        int x1 = scan.nextInt();
        int y1 = scan.nextInt();
        int length1 = scan.nextInt();

        System.out.println ("Please enter the left point
            coordinates of the other base followed by its length:");

        int x2 = scan.nextInt();
        int y2 = scan.nextInt();
        int length2 = scan.nextInt();

        // כאן עליכם להמשיך...

    } // end of method main
} //end of class Trapezoid
```

התכנית שכתבתם צריכה להיות במחלקה בשם Trapezoid. המחלקה Trapezoid תכיל שיטה אחת בלבד בשם main. אסור לכתוב שיטות נוספות.

בשתי השאלות במטלה זו -

- הקפידו שפלט התוכנית יהיה בדיוק כפי שרשום בהנחיות! אין להוסיף הודעות על מה שנכתב מפורשות בהנחיות ויש לדייק (כולל אותיות גדולות/קטנות וסימני פיסוק) בהודעות המפורטות.
- אתם צריכים לכתוב את התכנית של כל אחת משתי השאלות במחלקה אחת (מחלקה אחת עבור כל שאלה בשמות Train ו-Trapezoid), והכל בשיטה main. אין להוסיף שיטות נוספות.
- **אסור להשתמש בלולאות ו/או במערכים!**

הגשה

1. הגשת הממ"ן נעשית בצורה אלקטרונית בלבד, דרך מערכת שליחת המטלות.
2. הפתרון לשאלה 1 כולל את הקובץ Train.java.
3. הפתרון לשאלה 2 כולל את הקובץ Trapezoid.java.
4. ארזו את שני הקבצים בקובץ zip (ולא rar) יחיד ושלחו אותו בלבד.
5. **אל תשכחו לשמור את מספר האסמכתא שקיבלתם מהמערכת לאחר ההגשה. אם לא קיבלתם מספר אסמכתא, סימן שההגשה לא התקבלה.**
6. שימו לב, אתם יכולים לשלוח שוב ושוב את המטלה במערכת, אם אתם רוצים לתקן משהו בה. כל הגשה דורסת את ההגשה הקודמת. **אבל עשו זאת אך ורק עד לתאריך ההגשה.** אחרי התאריך, ייחשב לכם כאילו הגשתם באיחור, גם אם ההגשה הראשונה היתה בזמן! כמו כן, אם המנחה הוריד כבר את המטלה שלכם מהמערכת, לא תוכלו לשלוח עותק מעודכן יותר, ולכן אם המנחה הוריד את המטלה שלכם מהמערכת לפני תאריך ההגשה, תוכלו לפנות אליו במייל ולשלוח את המטלה המעודכנת שלכם.

בהצלחה