

מטלת מנחה (ממ"ן) 14

הקורס: 20453 - מבוא למדעי המחשב ושפת Java א

חומר הלימוד למטלה: יחידות 3 - 4 נושאי המטלה: שימוש במחלקות נתונות וכתיבת מחלקות

מספר השאלות: 2 משקל המטלה: 5 נקודות

סמסטר: 2019 מועד אחרון להגשה: 22.12.2018

(ת)

מטרת מטלה זו היא להקנות לכם את עיקרי התכנות מונחה-העצמים.

מדינת ישראל רוצה למפות את הערים שבשטחה.

לצורך כך נגדיר שתי מחלקות:

המחלקה Point שתייצג קואורדינטה במפה;

המחלקה City שתייצג עיר;

שאלה 1 - 40 נקודות

בהרצאות הקורס של ד"ר אמיר גורן, הוגדרה המחלקה Point שמייצגת נקודה במישור, לפי

מערכת הצירים הקרטזית (Cartesian system) – (בהרצאות המיקומים הם int וכאן double).

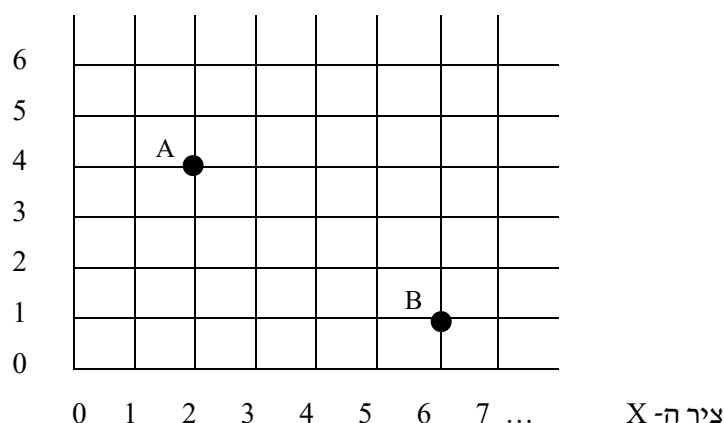
המחלקה Point שהוגדרה בהרצאות הכילה את התכונות הפרטיות (instance variables) הבאות:

- double _x – שמייצגת את המיקום על פני ציר ה-X;

- double _y – שמייצגת את המיקום על פני ציר ה-Y.

לדוגמא, הנה מסומנות שתי הנקודות $A = (2.0, 4.0)$ ו- $B = (6.0, 1.0)$ במרחב:

ציר ה-Y



בשאלה זו עליכם לכתוב מחדש את המחלקה Point. הפעם המימוש שלה יהיה לפי המערכת

הפולרית (Polar system).

המחלקה Point תייצג נקודה במישור ברביע הראשון בלבד .

למחלקה Point יש את התכונות הפרטיות (instance variables) הבאות:

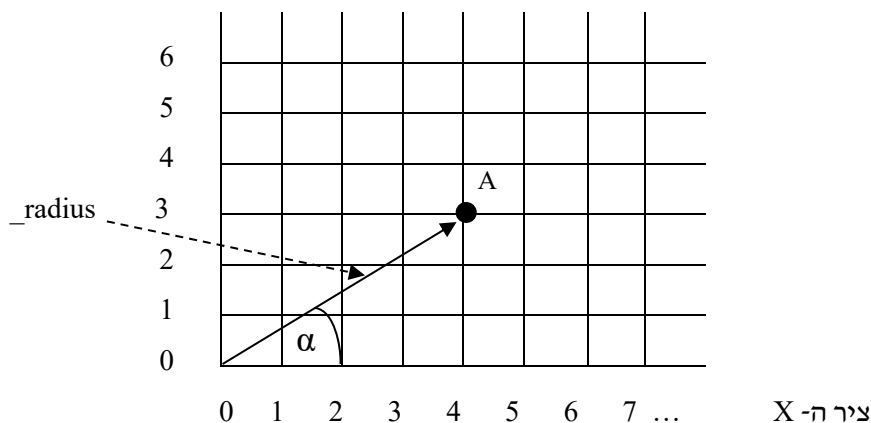
- `double _radius` – שמייצגת את אורך הוקטור מראשית הצירים עד הנקודה;
- `double _alpha` – שמייצגת את הזווית במעלות של הוקטור עם ציר ה- x .

לדוגמא, הנה מסומנת הנקודה A (שנמצאת בקואורדינטות (4.0,3.0) במרחב:

כאן אורך הוקטור שמחבר את הנקודה (0.0,0.0) עם (4.0,3.0) הוא 5.0

וערכה של הזווית α (alpha) הוא 36.87 מעלות (וברדיאנים - 0.64)

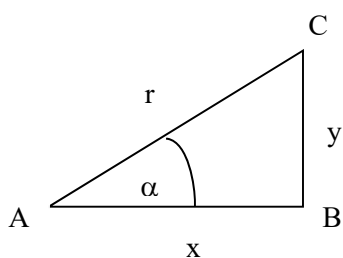
ציר ה- Y



תזכורת מתמטית קצרה

לדוגמא, נתון המשולש ישר הזווית הבא:

קדקודי המשולש הם A, B ו- C. צלעות המשולש הן x , y ו- r . (ראו תרשים להלן).



$$\sin(\alpha) = y/r$$

$$\cos(\alpha) = x/r$$

$$\tan(\alpha) = y/x$$

- מכאן, כאשר נתונים אורכי הצלעות x ו- y , אפשר לחשב את הזווית α על-ידי שימוש

בפונקציה ההפוכה \arctan . כלומר, $\alpha = \arctan\left(\frac{y}{x}\right)$ ערך הזווית בראדיאנים.

אם ערך ה- x הוא אפס אזי הזווית תהיה 90 מעלות.

- כאשר נתונים אורכי הצלעות x ו- y , אפשר לחשב את אורך הצלע r (היתר במשולש ישר-

זווית) על-ידי שימוש במשפט פיתגורס $r = \sqrt{x^2 + y^2}$

כזכור, על מנת לחשב שורש ריבועי של מספר, ניתן להשתמש בשיטה $\text{Math.sqrt}(x)$, שהיא שיטה של Java שנמצאת במחלקה Math . כדי להשתמש בה אין צורך לייבא אף מחלקה, אלא לקרוא לה בשמה המלא $\text{Math.sqrt}(x)$ 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).

בהמרה של ערך ממשי לשלם השתמש בפעולת עיגול $\text{Math.round}(x)$ המקבלת מספר ממשי x ומחזירה מספר שלם לפי כללי העיגול המקובלים.

כדי להימנע מאי דיוקים קטנים בחישובים של ממשיים, לדוגמא קבלת תוצאה 3.9999999994 במקום 4.0 השתמש בנוסחה: $\text{Math.round}(d*10000)/(\text{double})10000$ כאשר d מכיל את המספר הממשי שמעוניינים לעגל.

במחלקה Math תוכלו למצוא גם את השיטות \sin , \cos , \tan , atan (הוא \arctan), וגם את הקבוע Math.PI שהוא Π .

אפשר למצוא את ה-API של המחלקה Math בכתובת

<http://docs.oracle.com/javase/8/docs/api/java/lang/Math.html>

שימו לב:

המחלקה Math מתייחסת לזוויות ברדיאנים (radians) ולא במעלות (degrees). לכן, עליכם לעשות את השינויים המתאימים, ללא שימוש בשיטות toRadians או toDegrees של המחלקה.

להזכירכם:

$$\Pi = 3.14159... = \text{Math.PI} = 180^\circ$$

$$\text{Math.PI} / 2 = 90^\circ \text{ וכן הלאה}...$$

שוב, כיון שאנחנו מתייחסים במטלה זו רק לנקודות ברביע הראשון של מערכת הצירים, הזוויות האפשריות הן רק מ-0 עד 90° , כלומר מ-0 עד $\text{PI}/2$.

עליכם לכתוב את המחלקה Point (לפי המערכת הפולרית) לפי התיאור הבא:

שימו לב – ההתייחסות לנקודה היא לפי הפרמטרים של המערכת הקרטזית, כלומר הקואורדינטה בציר ה- x והקואורדינטה בציר ה- y , אולם המימוש הפנימי הוא לפי המערכת הפולרית.

לכן כל השיטות במחלקה בכלל לא יקבלו פרמטרים המתייחסים לתכונות לפי המערכת הפולרית. במימוש השיטות עליכם לדאוג להמרה הזו.

שימו לב שאינכם יכולים להגדיר תכונות נוספות על התכונות radius ו- alpha.

למחלקה Point הוגדרו שני בנאים (constructors):

- האחד - בנאי המקבל שני פרמטרים המהווים את ערכי התכונות שיהיו לנקודה.
`public Point(double x, double y)`
אם אחד הפרמטרים שהתקבל הוא שלילי, הוא צריך להיות מאותחל ל-0.
- השני - בנאי העתקה המקבל נקודה אחרת, ומעתיק את ערכיה.
`public Point (Point other)`

בנוסף, הוגדרו במחלקה השיטות:

- שיטות האחזור:
 - `double getX()` המחזירה את ערכה של קואורדינטת ה- x.
 - `double getY()` המחזירה את ערכה של קואורדינטת ה- y.
- השיטות הקובעות:
 - `void setX (double num)` המשנה את ערכה של קואורדינטת ה- x להיות num. אם num הוא מספר שלילי, הערך של x לא משתנה.
 - `void setY (double num)` המשנה את ערכה של קואורדינטת ה- y להיות num. אם num הוא מספר שלילי, הערך של y לא משתנה.
- השיטה `toString` שמחזירה את תוכן האובייקט כמחרוזת תווים לפי הייצוג המתמטי המקובל - (x,y). כלומר, להדפיס את הנקודה לפי מערכת הצירים הקרטזית. כך, המחרוזת (3.0,4.0) מייצגת את הנקודה שקואורדינטת ה- x שלה היא 3.0 וקואורדינטת ה- y שלה היא 4.0. שימו לב לדייק במחרוזת לפי הכתוב כאן. ללא רווחים וללא תווים נוספים. אין צורך להקפיד בעניין מספר הספרות העשרוניות שאחרי הנקודה.
- `boolean equals (Point other)` – שיטה שמקבלת נקודה כפרמטר ומחזירה האם הנקודה שעליה הופעלה השיטה והנקודה שהתקבלה כפרמטר זהות.
- `boolean isAbove (Point other)` - שיטה שמקבלת נקודה כפרמטר ומחזירה האם הנקודה שעליה הופעלה השיטה נמצאת מעל לנקודה שהתקבלה כפרמטר. (באיור הראשון, הנקודה A נמצאת מעל לנקודה B)
- `boolean isUnder (Point other)` - שיטה שמקבלת נקודה כפרמטר ומחזירה האם הנקודה שעליה הופעלה השיטה נמצאת מתחת לנקודה שהתקבלה כפרמטר. השיטה הזו משתמשת אך ורק בשיטה `isAbove` שהוגדרה לעיל.
- `boolean isLeft (Point other)` - שיטה שמקבלת נקודה כפרמטר ומחזירה האם הנקודה שעליה הופעלה השיטה נמצאת משמאל לנקודה שהתקבלה כפרמטר. (באיור הראשון, הנקודה A נמצאת משמאל לנקודה B)

- boolean isRight (Point other) - שיטה שמקבלת נקודה כפרמטר ומחזירה האם הנקודה שעליה הופעלה השיטה נמצאת מימין לנקודה שהתקבלה כפרמטר. השיטה הזו משתמשת אך ורק בשיטה isLeft שהוגדרה לעיל.
- double distance (Point p) – שיטה שמקבלת נקודה כפרמטר ומחזירה את המרחק בין הנקודה שעליה הופעלה והנקודה שהתקבלה כפרמטר. לעזרתכם, הנוסחה לחישוב מרחק בין הנקודה (x_1, y_1) , (x_2, y_2) היא $\sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}$
- void move (double dx, double dy) – המזיזה את הנקודה ב- dx על ציר ה- X וב- dy על ציר ה- Y. אם התזוזה מזיזה את הנקודה מחוץ לרביע הראשון של מערכת הצירים, הנקודה תישאר במקומה ולא תזוז.

עליכם לכתוב את המחלקה Point לפי ההגדרות לעיל.

הגדרות מדויקות לפי API תמצאו באתר הקורס בספר הדיגיטלי של יחידות 3-4, בתת-פרק של מטלה 12.

אתם יכולים להגדיר שיטות פרטיות נוספות על אלו שהוגדרו לעיל, אבל לא שיטות ציבוריות ולא תכונות נוספות.

מותר להשתמש אך ורק בשיטות הבאות מהמחלקה *Math*:

cos, sin, atan, round, pow, sqrt, PI וכן בקבוע *PI*.

שאלה 2 - 60 נקודות

המחלקה City מייצגת עיר במדינה.

למחלקה City התכונות הפרטיות (instance variables) הבאות:

- String _cityName – שם העיר
- Point _cityCenter – מיקום מרכז העיר
- Point _centralStation – מיקום התחנה המרכזית בעיר
- long _numOfResidents – מספר התושבים בעיר. חייב להיות חיובי או אפס.
- int _noOfNeighborhoods – מספר השכונות בעיר. חייב להיות חיובי ממש.

גם למחלקה City הוגדרו שני בנאים:

- האחד - המקבל פרמטרים עם ערכים לתכונות המתאימות: שם העיר, מיקום מרכז העיר (שני ממשיים שמייצגים את x ו- y), מיקום התחנה המרכזית (שני ממשיים שמייצגים את x ו- y), מספר התושבים (אם מספר התושבים שהתקבל הוא שלילי, מספר התושבים יוגדר להיות אפס) ומספר השכונות בעיר (אם מספר השכונות שהתקבל הוא שלילי או אפס, מספר השכונות יוגדר להיות 1).
- השני - בנאי העתקה, המקבל אובייקט מהמחלקה City ומעתיק את ערכיו.

כמו כן, הוגדרו פעולות האחזור (get) והפעולות הקובעות (set) לפי החתימות:

```
String getCityName()
Point getCityCenter()
Point getCentralStation()
long getNumOfResidents()
int getNoOfNeighborhoods()
void setCityName(String cityName)
void setCityCenter(Point cityCenter)
void setCentralStation(Point centralStation)
void setNumOfResidents(long numResidents)
void setNoOfNeighborhoods(int noOfNeighborhoods)
```

והשיטה toString שמחזירה את נתוני העיר (שם, מיקום מרכז העיר ותחנה מרכזית, מספר תושבים ומספר השכונות) כמחרוזת, לפי הפורמט לדוגמא הבא בדיוק:

City Name: Tel Sharon

City Center: (100.0,150.0) **הערה – אין צורך להתייחס למספר הספרות אחרי הנקודה**

Central Station: (45.0,36.0)

Number of Residents: 450

Number of Neighborhoods: 10

למחלקה City נוסף גם את השיטות הציבוריות:

- boolean addResidents(long residentsUpdate) שיטה המקבלת מספר תושבים חדשים שהצטרפו לעיר, מוסיפה אותם למספר התושבים הקיימים, ומחזירה true. השיטה צריכה לטפל גם במקרה בהם נגרע מספר התושבים (בעקבות הגירה של תושבים מהעיר). שימו לב, שמספר התושבים לא יכול להיות שלילי ולכן אם מספר התושבים שמהגרים מהעיר גדול ממספר התושבים בעיר, השיטה צריכה לאפס את מספר התושבים, ולהחזיר false.

- `void moveCentralStation(double deltaX, double deltaY)` שיטה המזיזה את התחנה המרכזית שבעיר למיקום אחר, השיטה מקבלת שני ממשיים `deltaX` ו-`deltaY` המורים בכמה להזיז את התחנה המרכזית בציר ה-X ובציר ה-Y בהתאמה. אם התזוזה מזיזה את התחנה מחוץ לרביע הראשון של מערכת הצירים, התחנה תישאר במקומה ולא תזוז.
- `double distanceBetweenCenterAndStation()` שיטה המחזירה את המרחק בין מרכז העיר ובין התחנה המרכזית.
- `City newCity(String newCityName, double dX, double dY)` שיטה שמקבלת מחרוזת תוים שמהווה שם, ועוד שני מספרים ממשיים `dX`, `dY`, ומחזירה עיר חדשה ששמה הוא ערך המחרוזת, מיקום מרכז העיר הוא במרחק `dX`, `dY` ממרכז העיר עליה היא מופעלת, מיקום התחנה המרכזית בעיר החדשה הוא במרחק `dX`, `dY` מהתחנה המרכזית עליה היא מופעלת, אין בה תושבים ויש בה שכונה אחת. גם כאן, אם התזוזה מזיזה את מרכז העיר או את התחנה המרכזית מחוץ לרביע הראשון של מערכת הצירים, הנקודה תישאר במקומה ולא תזוז.

עליכם לכתוב את המחלקה `City` לפי ההגדרות לעיל.

הגדרות מדויקות לפי API תמצאו באתר הקורס בתת-פרק "מטלה 12" שבספר הדיגיטלי של יחידות 3-4.

שימו לב, בכל שאלות המטלה :

- אסור להוסיף תכונות פרטיות למחלקות.
- מותר להוסיף שיטות פרטיות אבל לא ציבוריות.
- אין להשתמש במספרים בקוד. יש להוסיף קבועים (`final`) עבור כל מספר קבוע ולהשתמש בקבוע בקוד.
- בכל השיטות במטלה שמקבלות אובייקט כפרמטר אפשר להניח שמתקבל אובייקט שאותחל ואינו שווה ל-`null`.
- שימו לב לא לבצע aliasing במקומות המועדים.
- הגדרות מדויקות לבנאים ולשיטות הנדרשות לפי API תמצאו באתר הקורס.
- עליכם לתעד את כל המחלקות שתכתבו ב-API וגם בתיעוד פנימי. אפשר כמובן להשתמש בהערות ה-API שנמצאות באתר.

שימו לב ששמנו טסטרים לשתי המחלקות באתר הקורס. חובה שטסטרים אלו ירוצו ללא שגיאות קומפילציה עם המחלקות שלכם. אם יש שיטה שלא כתבתם, כתבו חתימה והחזירו ערך סתמי כדי שהטסטרים ירוצו עם המחלקות ללא שגיאות קומפילציה. אם הטסטרים לא ירוצו בגלל שגיאות קומפילציה הציון במטלה יהיה אפס.

הגשה

1. הגשת הממ"ן נעשית בצורה אלקטרונית בלבד, דרך מערכת שליחת המטלות.
2. הקפידו ששמות המחלקות והשיטות יהיו בדיוק כפי שמוגדר בממ"ן. **אחרת יורדו לכם הרבה נקודות!**
3. עליכם להריץ את הטסטרים שנמצאים באתר הקורס על המחלקות שכתבתם. שימו לב שהטסטרים לא מכסים את כל האפשרויות, ובפרט לא את מקרי הקצה. הם רק בודקים את השמות של השיטות במחלקות. מאד מומלץ להוסיף להם בדיקות
4. את התשובות לשאלות יש להגיש בשני קובצי Java הבאים : `City.java`, `Point.java`. אין צורך להגיש את קובצי ה-API שכתבתם.
5. ארזו את שני הקבצים בקובץ zip יחיד ושלחו אותו בלבד.

ב ה צ ל ח ה