

אוניברסיטת אריאל

מבני נתונים - סמסטר ב' תשפ"א

מטלה 4

הנחיות:

- מטלה זו הינה להגשה ביחידים. אין למסור ולקבל פתרון מתלמיד אחר. העתקה תגרור **לפסילה מלאה** של המטלה למעתיק והמועתק.
- יש להקליד את התשובות בקובץ טקסט או וורד (וכו') בשם Ex4. ניתן להוסיף איורים.
- למטלה מצורפות (הלינק למטה) מחלקות, יש להגיש אותן לאחר מילוי הקוד הדרוש יחד עם קבצי התשובות.
- יש להגיש את כל קבצי המטלה בקובץ ZIP (ולא כל דחיסה אחרת). שם קובץ ה-ZIP יהיה מספר ת.ז. של התלמיד בלבד (ולא כל שם אחר). אין להגיש קבצים או תיקיות מיותרים. סטייה מהנחיות אלו תגרור הורדה בציון.
- כל קבצי הקודים הקשורים למטלה זו נמצאים בלינק הבא:
https://github.com/saedAsa/DS_2021AU/tree/master/DS_2021AU/src/Ex4

שאלה 1:

- א. בהינתן ערימה מנימלית (minimum-heap) אשר כל אבריה שונים, איפה (באיזה מקום) אפשר למצוא את הערך הגדול ביותר בערימה? נמקי היטב.
- ב. נתונה המחלקה MaxHeap.java. השלימו את הפונקציה הסטטית mergeTwoHeaps אשר מקבלת שתי ערימות מקסימליות, ממזגת אותן לערימה מקסימלית אחת ומחזירה אותה.
- ג. השלימו את הפונקציה הסטטית getMinHeap שמקבלת ערימה מקסימלית, ומחזירה מערך שמייצג ערימה מנימלית שמורכבת מהאיברים של הערימה הנתונה.
- ד. מהי סיבוכיות כל אחת מהפונקציות שכתבתם למעלה? נמקי היטב.

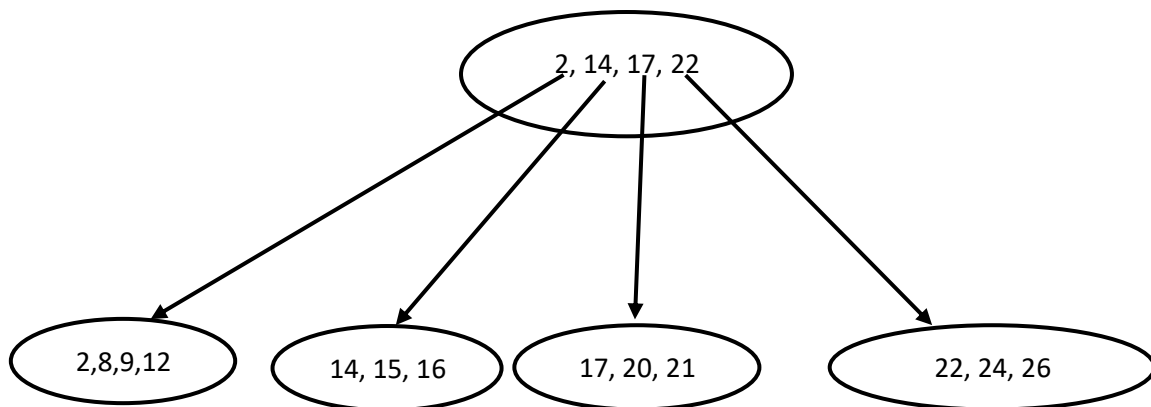
שאלה 2:

- א. השלימו במחלקה Ex4.java את הפונקציה ThreeSum המקבלת מערך של מספרים שלמים arr, ומספר מטרר target, ומחזירה רשימה של **כל** "השלשות" (מערכים בגודל 3) מהמערך הנתון **השונות אחת מן השנייה** אשר כל אחת מהן מקיימת את התנאים הבאים:
 1. המספר הקטן במערך נמצא במקום הראשון והגדול במקום האחרון.
 2. המספרים במערך שונים זה מזה
 3. סכום המספרים שווה ל- target
- ב. נתחו את סיבוכיות הפונקציה שכתבתם למעלה.

- **הפונקציה צריכה לרוץ באופן היעיל ביותר.**
- שימו לב שהפעולות contains ו-add של HashSet של java רצות בזמן $O(1)$ – למידע נוסף נא ללחוץ כאן.

שאלה 3:

נתון עץ B+ הבא:



כאשר דרגת העץ הינה 4 (הפרמטר b הינו 4).

מה תהיה תוצאת הוספת הערכים: 6, 7, 1, 18, 31 (משמאל לימין)? – ציירו את כל השלבים

שאלה 4:

בשאלה זו נרצה לקבץ נקודות במישור לקבוצות לפי מרחקן הזוויתי מנקודה כלשהיא.

נתונות המחלקות הבאות (שאינן לשנות שום דבר בהן):

- המחלקה StdDraw.java - מסייעת לייצג גרפית צורות גיאומטריות (כמו נקודה, מעגל, פוליגון וכו'...).

- המחלקה Ex4Utils.java המכילה את הפונקציות הבאות:

1. `public static Point[] generateRandomArray(int size)`

הפונקציה מחזירה מערך של נקודות אקראיות בגודל size.

2. `public static Color[] generateColorsArray(int size)`

הפונקציה מחזירה מערך של צבעים אקראיים בגודל size (כל צבע מייצג נקודה).

3. `public static double angleFrom(Point to, Point from)`

הפונקציה מחזירה את המרחק הזוויתי בין שתי הנקודות from ו-to.

4. `public static void showPoints(UnionFind uf)`

הפונקציה מציגה על המסך את הנקודות שיש במשתנה uf כך שכל קבוצת נקודות תוצג בצבע אחר.

5. `public static int countUniqueGroups(UnionFind uf)`

הפונקציה מחזירה את כמות הקבוצות הזרות שיש במשתנה uf.

כמו כן, נתונה המחלקה UnionFind.java **שאותה יש להשלים**.

המחלקה מייצגת אוסף של קבוצות זרות של נקודות.

א. השלימו את פעולת הבנאי שמקבלת את כמות הנקודות שיש להחזיק (size) וזווית (angle) - שלפיה נרצה לקבץ את הנקודות.
הפעולה מאתחלת את אוסף הקבוצות ע"י יצירת size נקודות אקראיות ומקבצת אותן לקבוצות זרות כל שכל נקודה שייכת לקבוצה משלה (נקודה מס' i תשויך לקבוצה מס' i), ולאחר מכן, תקבץ את הנקודות לפי הזווית (ראו סעיף ד').

ב. `public int Find(int p)`

הפונקציה מקבלת אינדקס של נקודה ומחזירה את מס' הקבוצה שאליה שייכת.

ג. `public static void Union(int ind1, int ind2)`

הפונקציה מקבלת שני אינדקסים ind1 ו-ind2, ומאחדת את הקבוצות שאליהן שייכות.

ד. `public static void UnionByAngularDist (Point p)`

הפונקציה מאחדת את הנקודות לפי הזווית ביניהן לבין הנקודה p באופן הבא:

- כל הנקודות שהזווית שלהן בין 0-angle יהיו באותה קבוצה.
- כל הנקודות שהזווית שלהן בין $angle-2*angle$ יהיו באותה קבוצה.

וכן האלה...

לחישוב המרחק הזוויתי בין שתי נקודות יש להשתמש בפונקציה `angleFrom`.

ה. `public void increaseAngle(int d)`

מגדילה את הזווית ב-d מעלות, ומאחדת את הנקודות באותו אופן כמו סעיף ד'.

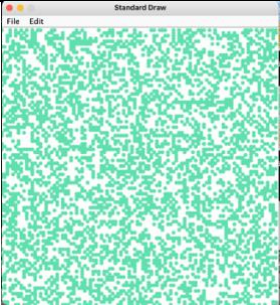
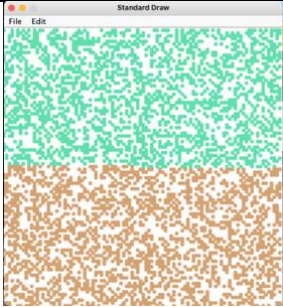
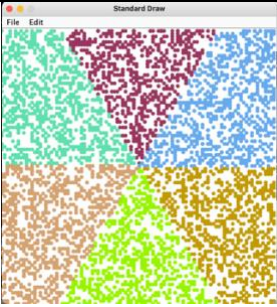
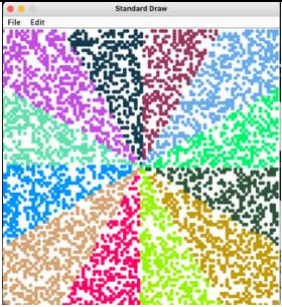
- שימו לב שתוצאת איחוד של שתי נקודות השייכות לקבוצות זרות הינה איחוד שתי הקבוצות שלהן.**

דרישות:

- יש ליישם את הפונקציות יעיל ככל האפשר.
 - יש לנתח את זמני הריצה של כל הפונקציות שכתבתם.**
- לבדיקת עבודתכם יש להפעיל את פונקציית ה-main שיש במחלקה `UnionFindSimpleTester.java`.

- בכל לחיצת space הזווית תגדל.

דוגמאות הרצה בעמוד הבא...

אחרי הגדלת הזווית ב- 30 מעלות 3 פעמים	אחרי הגדלת הזווית ב- 30 מעלות פעמיים	אחרי הגדלת הזווית ב- 30 מעלות	angle = 30
			

דוגמאות להרצות :

- מס' הנקודות בדוגמה למעלה הינו 5000.
- הנקודה P שממנה נמדדת הזווית בדוגמה למעלה הינה הנקודה (50,50) – מרכז המסך.

תיהנו,

בהצלחה !