**תרגיל בית 1**

**שאלה 1**

ב. לא תזרק אף חריגה של של clone() (CloneNotSupportedException) במחלקה Shape מכיוון שהמחלקה implements מ cloneable וגם אנחנו קוראים בתוך המתודה לsuper.clone() כך שמחלקת האב((Object שמטפלת לא תזרוק חריגה מהסוג הנ"ל (או שתטפל בחריגה הנזרקת) אם כל המחלקות היורשות implements מ cloneable.

ד. מכיוון שאין מתודות Mutetors במימוש המחלקה Color לכן הערך ההתחלתי לא ניתן לשינוי מכאן הוא immutable דבר זה משפיע על מימוש Shape בכך שאין צורך לעשות clone ל Color מכיוון שגם ככה לא ניתן לשינוי.

**שאלה 2**

ה. הצעתו של הסטודנט לא נכונה ואנחנו לא מסכימים איתה מכיוון שלמשל מימוש setsize של מחלקת LocationChangingCircle תהיה יותר מצומצמת מהמימוש של setsize ב LocationChangingOval (יהיה צריך להיות מלבן חוסם בעל אורך וגובה שווים- ריבוע) ולכן המפרט של המחלקה יהיה גם יותר חלש ולכן לא יקיים את עקרון ההחלפה של ליסקוב ולא יהיה true subtype.

ו. עקרונית ניתן לעשות זאת, ע"י קיבוע ערכי הגזרה(ל360) כאשר בתוך המחלקה של האליפסה נעשה את מתודת הצעד כך שתשנה את ערכי המיקום של הסקטור וכמובן שלא נעשה שום שימוש במתודת הצעד של הסקטור.

ז. גם כאן באופן עקרוני יהיה ניתן לממש קשר כזה שיהיה java subtype באופן שהמחלקה היורשת תדרוס את פונקצית הmutator ולמעשה תהפוך אותה ללא רלוונטית למשל ולכן היא תהיה immutable. אך קשר כזה לא יהיה true subtype מכיוון שהמפרט של המחלקה היורשת חלש יותר ולא מקיים את עקרון ההחלפה של ליסקוב.

ח. בוודאי שניתן לייצר אובייקט כזה שמהווה קשר של has a shape כך שshape הוא mutable והאובייקט יהיה imutable שפשוט לא יהיו מתודות כלשהן שישנו אותו וכן שהוא מוגדר כprivate שאין גישה אליו ללא המתודות(למשל מלבן גדול שמורכב מהרכבה של שדות של מלבנים קטנים שלא ניתנים לשינוי) אך אין כאן באמת משמעות של java/true subtype מכיוון שמדובר בהרכבה ולא הורשה.

**שאלה 3**

ד. בפעם שהשתמשנו בטיפוס shape השתמשנו כדי להשתמש במתודת הdraw שהיא מתודה אבסטרקטית של מחלקת shape שאותה יורשים כל תתי המחלקות ונותנים לה מימוש שונה (לכל תת מחלקה) כך שלמעשה אנו עושים שימוש בפולימורפיזם כדי להריץ כל פעם את מתודת draw של תת המחלקה הספציפי. בפעם שהשתמשנו בטיפוס animatable אשר תת המחלקות משתמשות בו כinterface עשינו זאת כדי לגשת למתודת step אשר מחלקת shape לא מכירה אך המחלקות היורשות שלה כן ולכן היינו צריכים לעשות casting כדי להשתמש בפולימורפיזם של animatable . כמו שנכתב, המנגנות שמאפשר זאת בשפת java הוא פולימורפיזם ובפרט גם Interface.

ו. המחלקה Animator יורשת מהמחלקה javax.swing.JFrame על מנת לאפשר את הופעת המסך (GUI) וזה מתבטא בקוד המחלקה למשל באמצעות שימות בשימוש מתודות שעברו בהורשה כמו שמובא למשל בקוד ב: getContentPane().getGraphics() , setJMenuBar(menuBar).

המחלקה Animator מממשת ממשק של java.awt.event.ActionListener כדי לעשות )לחייב) שימוש במתודות המאפשרות לקשר בין תצוגת המסך לבין המשתמש למשל כאשר מכניסים צורה חדשה או מאפשרים תזוזה על המסך באמצעות מימוש: ( public void actionPerformed(ActionEvent evt כאשר נקלט event אז ActionListener שולח ActionEvent ל actionPerformed אותו אנו מממשים כדי שיבצע את הפעולה שעלינו לעשות.

ז. השינוי שיידרש לעשות הוא רק לשנות את היצירה של הקונטיינר לרשימה וכן לעשות יבוא כנדרש וזהו. זאת מכיוון שהמימוש התבסס על שימוש באיטרטור אשר יכול לעבור גם על מבני נתונים אחרים באמצעות שימוש במתודות שבהם הוא תומך.