**שאלות על המימוש – נמרוד מחלב 315230185**

1. דוגמה ליחסים:
   1. **הורשה** – UndirectedGraph ו- DirectedGraph יורשים מGraphBase את התכונות שלו והמימושים שלו לפעלות השונות.
   2. **הפשטה** – IGraph מייצג את ממשק הגרפים **בלי לחשוף/לציין** **את** **צורת המימוש הספציפית** ובכך "מפשיטה" את הממשק הגרפים למשתמש.
   3. **יחס הכלה** – GraphBase מכיל כשדה (וגם הבנים שיורשים ממנו את השדה) את- SortedMap<V, SortedSet<V>>.
   4. **דריסה של מתודה** - **UndirectedGraph** **דורס** את **AddEdge** שממומש בGraphBase **על מנת להוסיף למימוש.**
   5. **פולימורפיזם** – הרב צורניות **מתקבלת** **בtests** **GraphTestAbstract** כאשר הן מתבצעות **תוך** **הסתכלות** על **המנשק** **IGraph** **בלבד**, בלי התייחסות למימוש הספציפי.
   6. **אין דוגמה להעמסה.**
2. **מבחינה טכנית ניתן להגדיר הורשה** בין DirectedGraph ל- UndirectedGraph כאשר Undirected יירש מDirected אבל **התקינות של הדבר תלויה בשימוש הספציפי של התכנית**, בחרתי שלא לעשות ככה מכמה סיבות:
   1. למעשה directed וundirected בקוד הם טיפוסים שונים, המתודה getGraphType צריכה להחזיר את הסוג שלהם, בכל אחד מהם היא מחזירה סטרניג שונה, מה שרומז לנו שהמימוש צריך להיות נפרד.   
      אז למשל אם **נגדיר** **מתודה** **שמקבלת** **directed** **graph** נצפה שנקבל **התנהגות** **של** עצם **directed** **graph** ולא שאולי נקבל מאחורי הקלעים נקבל undirected graph – שוב **תלוי בדרישות של התוכנית עצמה**.
   2. עוד סיבה היא שהשדה sortedmap יכול להיות private במימוש שלי ותוך כדי לקבל את הטיפוס הספציפי שלו מundirected וdirected בניפרד, אם נגדיר יחס הורשה בניהם נהיה חייבים להשתמש באותו מימוש לsortedmap בשניהן.
3. בקטע קוד רואים שמוסיפים לאותו set **שני עצמים שונים של גרפים רייקים**, למעשה המתודה **equals שאותה הset בודק תחזיר false** ולכן שניהם יתווספו לקבוצה **והתצואה תהיה "2",** כי שניהם מצביעים למקום אחר בזיכרון.

על מנת לגרום לקטע קוד להחזיר 1 **יש לדרוס את equals ואת** **hashcode** ולממש אותן על פי השדות של המחלקה, למשל **לבדוק שלכולם יש אותם קודקודים ואותן קשתות בדיוק**, ניתן זאת כבר במחלקה האבסטרקתית.