תכנות ותכן מונחה עצמים – תרגיל 2

מגישים:

גילי קרני 305200420

גל שחף 201285566

# שאלה 1

## סעיף א' – תיעוד חיצוני

בחרנו בפעולות הבאות ע"מ לממש את ההפשטה:

1. אתחל גרף נקי.
2. הוספת צומת עם label (שחור/לבן)
3. הוספת קשת עם label
4. החזר רשימת צמתים (שחורים/לבנים)
5. החזר רשימת labels של הורים של צומת לפי label
6. החזר רשימת labels של ילדים של צומת לפי label
7. החזר label של ילד לפי label קשת וlabel הורה
8. החזר label הורה לפי label קשת ו- label ילד

בשביל למנוע שכפול קוד, החלטנו להוסיף את המתודות הפנימיות הבאות:

1. הוספת צומת (מקבלת label וצבע)

כמו כן, הוספנו מחלקה בשם ColoredVertex, אשר מממשת את ההפשטה של צומת. למחלקה זו הפעולות הבאות:

1. הוסף הורה לפי label
2. הוסף ילד לפי label
3. האם הורה קיים לפי label
4. האם ילד קיים לפי label
5. האם קשת נכנסת קיימת לפי label
6. האם קשת יוצאת קיימת לפי label
7. האם הצומת שחור
8. האם הצומת לבן
9. החזר רשימת labels של הורים
10. החזר רשימת labels של ילדים
11. החזר label ילד לפי label קשת
12. החזר label הורה לפי label קשת

התלבטנו האם לממש גם מחלקה של Edge או להתייחס לכל קשת רק בתור ה-label שלה. בשל המניעים הבאים החלטנו לא לממש מחלקה ייעודית:

1. אין דרישה בהפשטה להחזיר את כל הקשתות של הגרף, או אף את כל הקשתות של צומת, ועל כן אין צורך לשמור מידע על כל הקשתות בצורה מרוכזת.
2. קשת מוגדרת חח"ע ע"י מוצא, יעד ו-label. ע"מ לשמור מוצא ויעד בקשת, ובמקביל לשמור את הקשתות המחברות כל צומת, עלינו לבצע ייבוא (import) לולאתי.
3. בהמשך לנימוק 2: אמנם ניתן לשמור במחלקה Edge רק את ה-label-ים של המוצא ושל היעד, אבל שמנו לב שלפי ההפשטה שמידע זה הינו redundant, מכיוון שבכל חיפוש של ילד או הורה לפי קשת מתקבל גם פרמטר של label הורה או label ילד (בהתאמה), כך שניתן באמצעות label צומת זה ו-label קשת למצוא בצורה יעילה את הצומת המבוקש גם ללא אחזקת הנתונים באובייקט מסוג Edge.

לדעתנו, פעולות אלו מספקות למימוש ההפשטה. הן מאפשרות בניית גרף ע"י הוספת צמתים (לפי צבעים) וקשתות, אחזור רשימת צמתים לפי צבע, אחזור הורים וילדים של צומת, אחזור צומת לפי הורה/ילד וקשת. פעולות אלו מאפשרות לקבל גם מידע כללי על הגרף, וגם בניית מסלולים ותתי- גרפים מתוך הגרף.