

## מערכות 2 מטלה 2

נועה פישמן 319055430

noa.fishman@gmail.com

### Graph

המחלקה מייצגת את האובייקט גרף שמכיל בתוכו מטריצה מרובעת שהיא מטריצת השכנויות של הגרף, משתנה ששומר את גודל הגרף (כמות הקודקודים) ומשתנה ששומר את כמות הצלעות. בהמשך למטלה הקודמת השארתי את פונקציית load כמו שהיא הייתה לפני בנוסף גם את הפונקציות getSize | getWeight השארתי כמו שהם. התבקשנו להוסיף למחלקה אופרטורים שעליהם אפרט בהמשך אך כדי להקל על המימוש שיניתי את הפונקציה print כך שהיא לא מופעלת על הגרף שרוצים להדפיס אלא מפעילים את הפונקציה ונותנים לה את המשתנה שרוצים להדפיס. בנוסף, הוספתי פונקציית update שמטרתה לעדכן את הנתונים של הגרף לאחר שמופעל עליו אופרטור כלשהו שעלול לשנות את כמות הצלעות הפונקציה מעדכנת את כל הנתונים הנחוצים של האובייקט. וגם הוספתי את הפונקציה getEdge שמחזירה לנו כמה צלעות יש לגרף. כמו בתרגיל הראשון גם כאן אני מתייחסת לכל הגרפים בתור גרפים מכוונים ככה שאם גרף אינו מכוון המטריצה המייצגת אותו היא מטריצה סימטרית ועדיין כל צלע תיספר פעמיים. כעת אסביר מעט על כל פונקציה.

### **getEdge**

הפונקציה מחזירה כמה צלעות יש לגרף הנוכחי ככה המשתנה של האובייקט יכול להישאר פרטי וכך אנו בטוחים שלא ניתן לבצע עליו שינויים.

### **Update**

הפונקציה מופעלת אחרי אופרטורים חישוביים וסופר שוב כמה צלעות יש לגרף ומעדכן את הפרמטר באובייקט. ככה שאם האופרטור גרם לשינוי בכמות הצלעות המידע באובייקט יישאר מעודכן.

### **printGraph**

הפונקציה עובדת אותו הדבר כמו במטלה הראשונה רק במקום לבצע הכל על המשתנה אותו רוצים להדפיס כאן אני מקבלת את המשתנה שרוצים להדפיס בתור פרמטר ומחזירה מחרוזת שמייצגת את הגרף אותה אני רוצה להדפיס. בנוסף מדפיסה את המחרוזת המחוזרת.

### **אופרטור +**

האופרטור מופעל על גרף אחד ומקבל גרף שני הוא מחבר בין השניים (מחבר בין המריצות המייצגות אותם) ומהמטריצה החדשה שנוצרה מייצר גרף חדש ומחזיר אותו

### **אופרטור -**

האופרטור מופעל על גרף אחד ומקבל גרף שני הוא מחסר בין השניים (מחבר בין המריצות המייצגות אותם) ומהמטריצה החדשה שנוצרה מייצר גרף חדש ומחזיר אותו

### **אופרטור +=**

האופרטור מופעל על גרף אחד ומחבר אליו גרף נוסף לתוך הגרף הראשון. (מוסיף את המטריצה של הגרף השני לתוך המטריצה של הגרף הראשון) לבסוף מעדכן את הגרף הראשון כדי לעדכן את מספר הצלעות במידה והשתנה ומחזיר רפרנס לגרף המקורי.

## **אופרטור =**

האופרטור מופעל על גרף אחד ומחסר ממנו גרף נוסף. (מחסיר את המטריצה של הגרף השני מהמטריצה של הגרף הראשון) לבסוף מעדכן את הגרף הראשון כדי לעדכן את מספר הצלעות במידה והשתנה ומחזיר רפרנס לגרף המקורי.

## **אופרטור + (אונארי)**

האופרטור מקבל גרף ומחזיר את הגרף כפול 1 כלומר מחזיר את הגרף המקורי.

## **אופרטור - (אונארי)**

האופרטור מקבל גרף ומחזיר את הגרף כפול (-1) כלומר, מכפילים את המטריצה המייצגת ב(-1) ומקבים את הגרף כך שכל המשקלים בו שינו סימן.

## **אופרטור \***

האופרטור מופעל על גרף אחד ומקבל גרף נוסף ומכפיל ביניהם. האופרטור עושה זאת בעזרת כפל מטריצות, ראשית הוא מוודא שהמטריצות (הגרפים) אכן באותו הגודל ולאחר מכן מכפיל בין המטריצות ומהמטריצה החדשה שקיבלנו מייצרים גרף חדש ואותו מחזירים.

## **אופרטור ++**

האופרטור מופעל על גרף ומגדיל את המשקלים של כל הצלעות שלו ב 1. כלומר, לכל איבר במטריצה המייצגת מוסיפים 1. לבסוף מעדכנים את כמות הצלעות של הגרף ומחזירים את הרפרנס שלו.

## **אופרטור -**

האופרטור מופעל על גרף ומקטין את המשקלים של כל הצלעות שלו ב 1. כלומר לכל איבר במטריצה המייצגת מחסירים 1. לבסוף מעדכנים את כמות הצלעות של הגרף ומחזירים את הרפרנס שלו.

## **אופרטור /=**

האופרטור מופעל על גרף ומקבל פרמטר אשר נחלק את כל הצלעות של הגרף בפרמטר זה. הפונקציה עוברת על המטריצה המייצגת ומחלקת כל איבר בה בפרמטר. הפונקציה מעדכנת את הגרף ואת כמות הצלעות בהתאם ומחזירה את הגרף.

## **אופרטור \*=**

האופרטור מופעל על גרף ומקבל פרמטר אשר נכפיל את כל המשקלים של הצלעות בגרף בפרמטר זה. הפונקציה עוברת על המטריצה המייצגת ומכפילה את כל האיברים בה בפרמטר שקיבלנו. לבסוף מעדכנת את הגרף והצלעות בהתאם ומחזירה את הגרף המעודכן.

## **אופרטור ==**

האופרטור מופעל על גרף ומקבל גרף נוסף ומטרתו היא לבדוק האם הם שווים. ראשית נבדוק האם הם באותו הגודל ואם לא נחזיר false לאחר מכן נבדוק האם המטריצות המייצגות שלהם שוות אם כן נחזיר true ואם לא נבדוק, אם גרף אחד לא גדול מהגרף השני והוא גם לא קטן ממנו אז נחזיר true אחרת אחרי שבדקנו את כל האופציות לשיוויון נחזיר false.

## **אופרטור !=**

האופרטור מופעל על גרף ומקבל גרף נוסף. הפונקציה מפעילה על שני הגרפים את האופרטור של השוויון ואם מקבלת שהם שווים מחזירה false אחרת מחזירה true.

## אופרטור <

האופרטור מופעל על גרף ומקבל גרף נוסף ומחזיר האם הגרף עליו הופעל האופרטור קטן מהגרף השני שקיבלנו. ראשית הפונקציה בודקת האם הגרף מוכל בגרף שקיבלנו, עושים זאת בעזרת ריצה על כל תתי המטריצות האפשריות בגודל של הגרף הקטן, ובודקים האם המטריצה של גרף הקטן מוכלת בתוך המטריצה של הגרף הגדול. אם נמצאה תת מטריצה זהה נחזיר true ואם לא נמצאה וסיימנו לעבור על כל התת מטריצות מגודל זה נבדוק האם לגרף שעליו הופעל האופרטור יש פחות מצלעות מלגרף שקיבלנו. אם כן נחזיר true ואם לא נבדוק האם יש לו יותר קודקודים, אם כן נחזיר true אחרת, נחזיר false כי אף אחד מהתנאים לא מקיים שהגרף עליו הופעל האופרטור קטן מהגרף השני.

## אופרטור >

האופרטור מופעל על גרף ומקבל גרף נוסף ומחזיר האם הגרף עליו הופעל האופרטור גדול מהגרף השני שקיבלנו. ראשית הפונקציה בודקת האם הגרף שקיבלנו מוכל בגרף עליו הופעל האופרטור, עושים זאת בעזרת ריצה על כל התת מטריצות האפשריות בגודל של הגרף הקטן, ובודקים האם המטריצה של גרף הקטן מוכלת בתוך המטריצה של הגרף הגדול. אם נמצאה תת מטריצה זהה נחזיר true ואם לא נמצאה וסיימנו לעבור על כל תתי המטריצות מגודל זה נבדוק האם לגרף שקיבלנו יש פחות צלעות מלגרף שעליו הופעל האופרטור. אם כן נחזיר true ואם לא נבדוק האם יש לו יותר קודקודים, אם כן נחזיר true אחרת, נחזיר false כי אף אחד מהתנאים לא מקיים שהגרף עליו הופעל האופרטור גדול מהגרף השני.

## אופרטור >=

האופרטור מופעל על גרף ומקבל גרף נוסף ומחזיר האם יש ביניהם קשר של גדול או שווה. הפונקציה בודקת עבור שני הגרפים האם הם שווים או שהגרף עליו הופעל האופרטור גדול מהגרף השני ואם אחד מהם מתקיים מחזירה true אחרת תחזיר false.

## אופרטור <=

האופרטור מופעל על גרף ומקבל גרף נוסף ומחזיר האם יש ביניהם קשר של קטן או שווה. הפונקציה בודקת עבור שני הגרפים האם הם שווים או שהגרף עליו הופעל האופרטור קטן מהגרף השני ואם אחד מהם מתקיים מחזירה true אחרת תחזיר false.

## אופרטור <<

האופרטור מקבל גרף ורפרנס של פלט ומכניס לפלט ייצוג של הגרף. כלומר, הפונקציה עוברת על הגרף ממירה אותו למחרוזת ומכניסה אותה לתות הרפרנס שקיבלנו ומחזירה אותו.

## Demo

המחלקה דמו מראה לנו פונקציה ראשית כלשהי בה נעשה שימוש באופרטורים שבנית. בהוראות המטלה קיבלנו דמו ששימש עבורי כבסיס לדמו הזה, עליו הוספתי עוד פעולות ושינית את המטריצות כדי להראות עוד יכולות ומקרים שהאופרטורים מצליחים להתמודד איתם.

## Test

המחלקה טסט היא מחלקה שבה יש פונקציות אשר בודקות את האופרטורים שבנית. גם כאן לקחתי את המחלקה שקיבלנו שיניתי מעט והוספתי עוד הרבה בדיקות משלי על כל האופרטורים. עבור כל אופרטור ניתן לראות לפחות בדיקה אחת שמראה איך הוא מתמודד עם משקלים חיוביים ושליליים, מקרים שבהם חלק מצלעות עשויות להתאפס וחלק להתווסף. בנוסף ניתן לראות שעבור כל אופרטור שעשוי לשנות את כמות הצלעות בדקתי שאכן מספר הצלעות של הגרף התעדכן בהתאם.

## **Makefile**

מימשתי את מייקפייל כמו שמימשתי במטלה הראשונה. כדי לעשות מייק לדמו צריך לכתוב `make Demo` וכדי להריץ אותו צריך לכתוב `./Demo`. וכדי לעשות מייק לטסטים צריך לכתוב `make Test` וכדי להריץ אותם צריך לכתוב `./Test`.

## **Algorithms**

את המחלקה הזאת לא שיניתי מהמטלה הראשונה. הבנתי שכאשר במחלקת Graph עבור כל אופרטור שיכול לקרות בו שינוי אני מעדכנת אותו בהתאם, וכאשר כל הגרפים מוגדרים כראוי אין סיבה שהמימוש של הפונקציות שם ישתנה.