

## תרגיל בית 3

כיתבו מחלקה בשם `Member` המייצגת משתמש ברשת חברתית. המשתמש יכול לעקוב / להפסיק לעקוב אחרי חברים אחרים, וכן לראות אחרי כמה חברים הוא עוקב / כמה עוקבים אחריו. בנוסף המערכת זוכרת כמה חברים פעילים יש במערכת.

מצורף לתרגיל זה קובץ `main.cpp` עם תוכנית ראשית ובה כמה בדיקות. עליכם להוסיף את כל הקבצים הדרושים כדי שהתוכנית הראשית הזאת תתקמפל ותעבוד כפי שמוסבר בגוף הקוד. יש לכתוב בדיקות-יחידה נוספות כדי לוודא שהמחלקה שלכם מתמודדת עם מקרים נוספים. מותר לכם להוסיף מחלקות נוספות לפי הצורך.

**הגשה:** צרו מאגר (`repository`) חדש בגיטהאב והעלו לשם את הקבצים בתיקיה הראשית.

הוסיפו למאגר, בתיקיה הראשית, קובץ בשם `ids.txt` (אותיות קטנות בלבד) הכולל את מספרי תעודות הזהות של המגשים - כל מספר בשורה נפרדת.

הגישו בטופס-ההגשה קישור-שיבוט למאגר - הקישור שרואים כשלוחצים על הכפתור `clone` בגיטהאב.

**בדיקה אוטומטית:** אנחנו נבצע את הפקודות הבאות ממחשב עם לינוקס:

1. `git clone <שלכם>`
2. נעתיק לתוך התיקיה שלכם תוכנית ראשית משלנו, הכוללת בדיקות אוטומטיות נוספות וחישוב ציון.
3. `clang++-5.0 -std=c++17 *.cpp`
4. `./a.out`

אתם יכולים לפתור את התרגיל בכל סביבת-פיתוח שאתם רוצים, אבל לפני ההגשה, וודאו שהפקודות האלו רצות בלי שגיאות על מחשב לינוקס אחר כלשהו. אם אין לכם `clang` תשתמשו ב-`g++` במקום. שימו לב: תוכנת הבדיקה שלנו מורידה את כל המאגר, ולכן יש לפתוח מאגר חדש עבור כל תרגיל בית (אלא-אם-כן הוא המשך של תרגיל קודם).

### דגשים:

- יש לחזור על החומר של ההרצאות לפני שמתחילים לכתוב, ולהשתמש בו לפי הצורך.
- יש לתכנת בהתאם לכללים בסיסיים של הנדסת תוכנה, בפרט: חלוקה לקבצים, תיעוד בעברית או באנגלית, שמות משמעותיים למשתנים, ובדיקות-יחידה (`unit-test`).
- יש לתכנן את המחלקה כך שיהיה אפשר להוסיף לה בתרגילים הבאים אלגוריתמים נוספים בגרפים, כגון: חישוב מספר עוקבים-של-עוקבים, מציאת מעגל של עוקבים, וכו'.
- מותר להשתמש בתכונות מתקדמות של שפת `C++` גם אם עדיין לא נלמדו בהרצאות.
- אין להעתיק תרגילים שלמים מסטודנטים אחרים. מותר להיעזר בקטעי קוד מהאינטרנט, אולם **יש לציין בבירור את המקור**, לוודא שהקוד עובד, ולוודא שאתם מבינים למה הוא עובד.