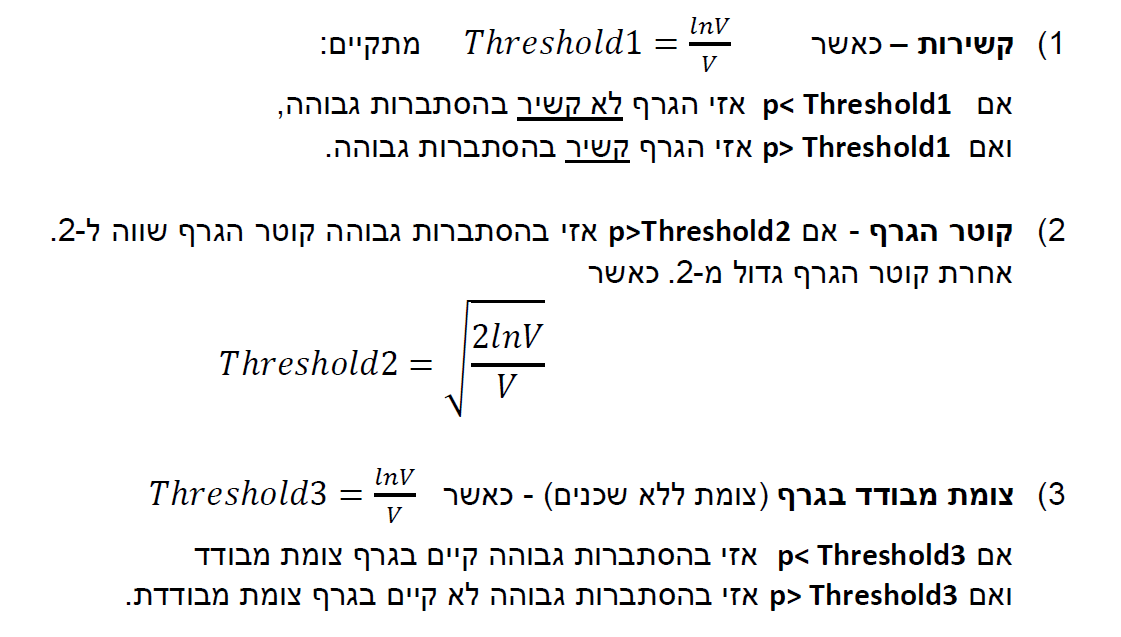
אלגוריתמים 2 – מטלה 1

קובץ README:

תאריך הגשה: 13.4.21, מגיש: ירדן אנקונינה )יוליה אישרה להגיש לבד), ת"ז :205415664

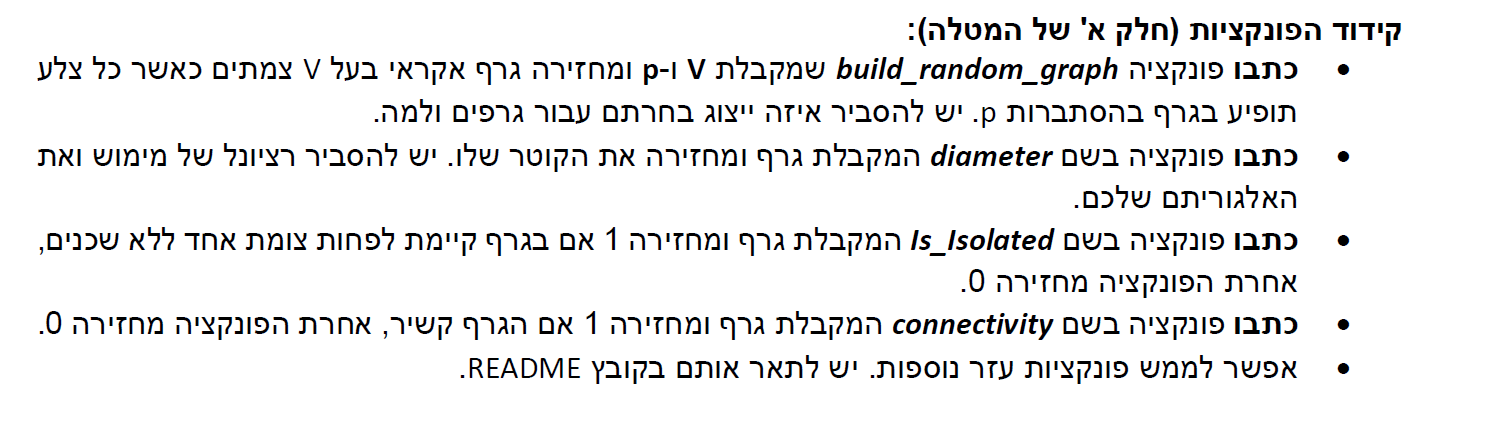
במטלה זו התבקשנו ליצור גרפים אקראיים במודל ארניש -ריניה (G(n, p)), בעלי 1000n= קודקודים על פי הסתברות P מסוימת ולבדוק האם מתקיימות 3 התכונות הבאות בגרפים שיצרנו:



**המסקנה** לאחר כתיבת התוכנית והסימולציות שנועדו לבדוק את התכונות הנ"ל, ניתן להסיק שאכן תכונות אלו מתקיימות (פירוט בהמשך).

ראשית אסביר את הקוד שכתבתי:

חלק א' :



**הסבר ייצוג הגרף:**

בחרתי לייצג את הגרפים במטלה ע"י רשימת סמיכויות (Adjacency list) וזה בעקבות הכמות צלעות שיש לי בגרף. מכיוון שההסתברויות להיווצרות צלעות הם נמוכות (סביבות 0.1 ומטה) אז הגרף שלי יחסית "דליל" בצלעות ולכן להשתמש במטריצת סמיכויות תהיה בזבוז מקום. והדרך שבה בחרתי יותר חסכונית - בעיקר שמדובר בגרפים גדולים (מטריצה -V^2 מקום, רשימת סמיכויות-E+V (.

**הסבר של פונקציית diameter :**

קלט : גרף, פלט :קוטר.

כדי למצוא את הקוטר ראשית בדקתי שהגרף קשיר מאחר שאם הוא לא קשיר הקוטר של הגרף הוא אינסוף – בדיקה זו ביצעתי ע"י פונ' BFS שארחיב עליה בהמשך.

לאחר מכן שידוע שהגרף קשיר, ביצעתי עבור כל קודקוד בגרף BFS (שימוש בBFS מכיוון שהוא מסוגל להחזיר את הdistance() של כל קודקוד מהsource) – שהחזיר לי את המרחק הגדול ביותר בגרף מהקודקוד מקור, לאחר מכן מציאת הקוטר הוא המקסימום בין כל המרחקים.

פונ' BFS : מקבל קודקוד מקור ומחזירה מרחק גדול ביותר מקודקוד מקור / האם גרף קשיר.

ללא מערך אבות או מערך צבעים כי אין צורך, מימוש אלג' BFS, עם מערך visited ,distance וqueue.

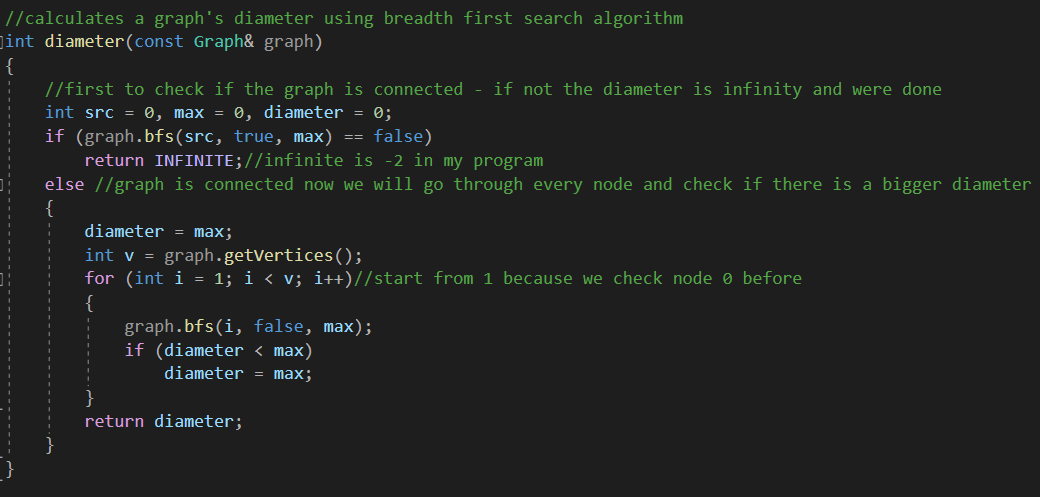
הסיבוכיות היא - עבור כל קודקוד עוברים על כל הצלעות שלו מוסיפים לתור ומחשבים את המרחק לכל השכנים, לא חוזרים על קודקודים פעמיים.

חישוב מרחקים ופעולות תור ומערך ביקורים הוא קבוע .

לאחר מכן אני עובר בלולאה על מערך המרחקים ובודק את המקסימום - .

לסיכום פונקציית BFS - .

כעת, מכיוון שאני משתמש בBFS לכל קודקוד בגרף ועוד פעולות קבועות שאינן בעלות חשיבות, סיבוכיות הזמן של פונ' diameter היא :



סיבוכיות שאר הפונקציות :

**Is\_isolated:**

עוברת על כל רשימה ברשימת סמיכויות ובודקת אם היא ריקה (גודל ==0),סיבוכיות זמן - .

**Build random graph:**

עוברים על כל זוגות הקודקודים כלומר בחירת זוג מתוך n קודקודים – C(n,2) = , ומגרילים מספר רנדומלי אם הוא גדול מהסתברות P אז ניצור צלע בין שני הקודקודים, סיבוכיות זמן - = .

**Connectivity *:***

*נבדוק האם הגרף קשיר ע"י הרצת* bfs *פעם אחת, אם בסוף ההרצה יש קודקוד במערך* visited *שלא ביקרו בו אז זה אומר שאין דרך להגיע אליו כלומר הגרף אינו קשיר,* סיבוכיות זמן *- .*

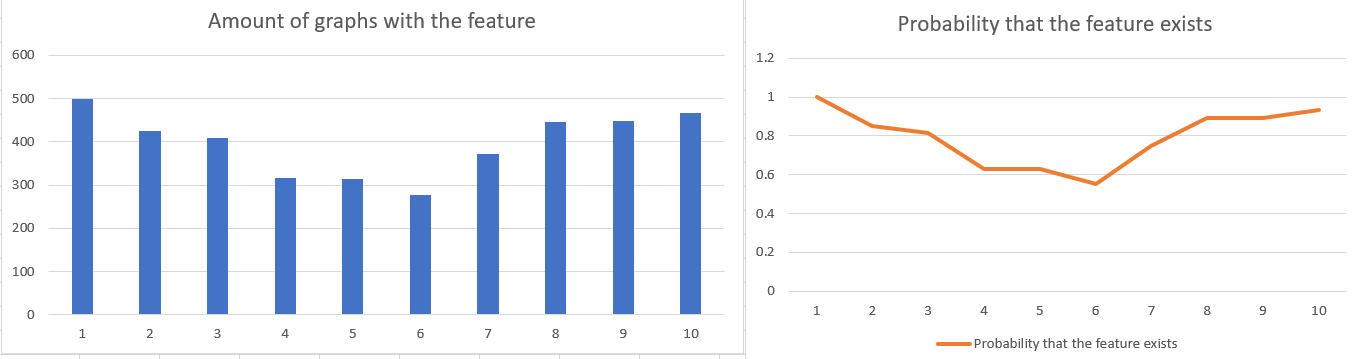
תוצאות הבדיקות

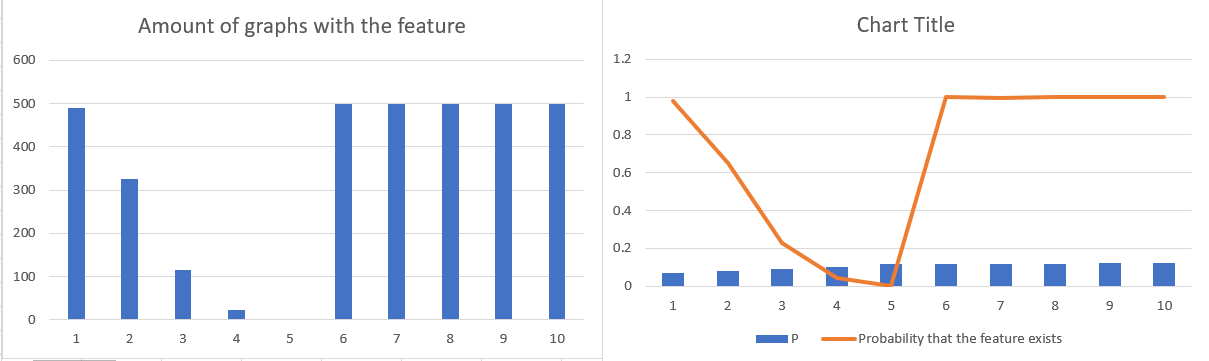
מצורפים קבצי EXCEL ,פירוט על הקבצים :

Simulation1A – כמה גרפים ענו על התכונה כלומר שהם קשירים וגם הסתברות P גדולה מthreshold או שהם לא קשירים וההסתברות P קטנה מthreshold.

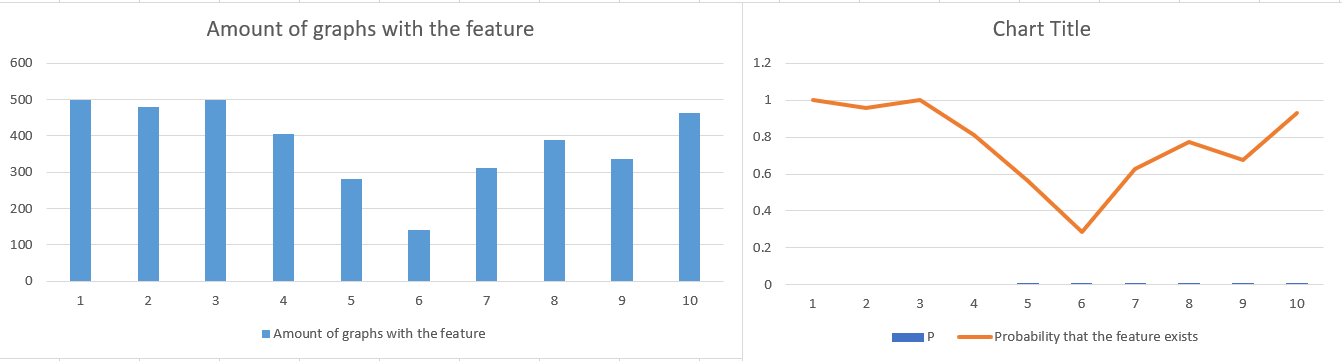
Simulation2A - באותו אופן כמה ענו על תכונה 2 בהתחשב להסתברות P.

Simulation3A - באותו אופן כמה ענו על תכונה 3 בהתחשב להסתברות P.

Simulation1A

Simulation2A

Simulation3A



כפי שניתן לראות בקבצים לשלושת התכונות ניתן לומר ש - התכונה *אכן* מתקיימת ככל שמתרחקים מהthreshold אז הסיכוי שהיא תתקיים גדל, כאשר מתקרבים לthreshold מאוד אז התכונה לא בהכרח מתקיימת, כמובן שהגרפים יראו אחרת עבור הסתברויות שונות שנבחר מסביב לthreshold.

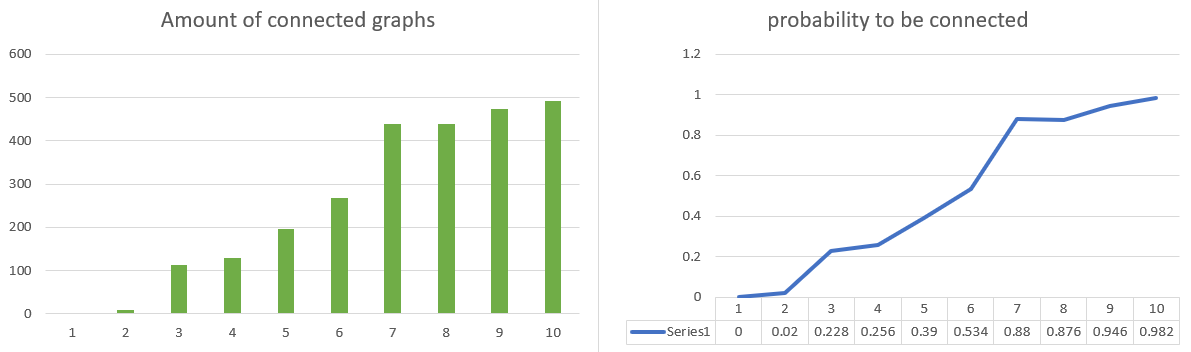
קבצים נוספים : (בדיקת סימולציה נוספת עבור כל תכונה – גרפים רנדומליים שונים מאשר ה3 הקודמים)

Simulation1B – מראה את מספר הגרפים הקשירים, ומחשב את ההסתברות להיות קשיר(חלוקה ב500 גרפים).

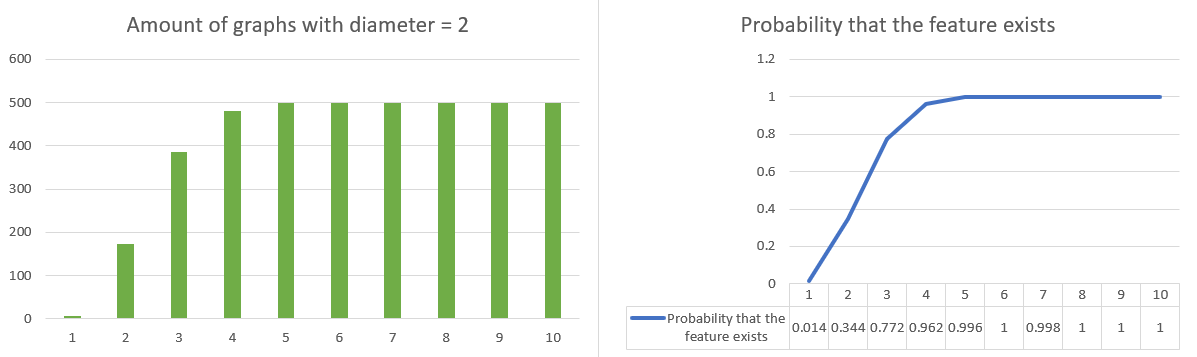
Simulation2B – מראה את מספר הגרפים עם קוטר = 2, ומחשב את ההסתברות לכך(חלוקה ב-500).

Simulation3B – מראה את מספר הגרפים עם קודקוד ללא שכנים, ומחשב את ההסתברות לכך(חלוקה ל500).

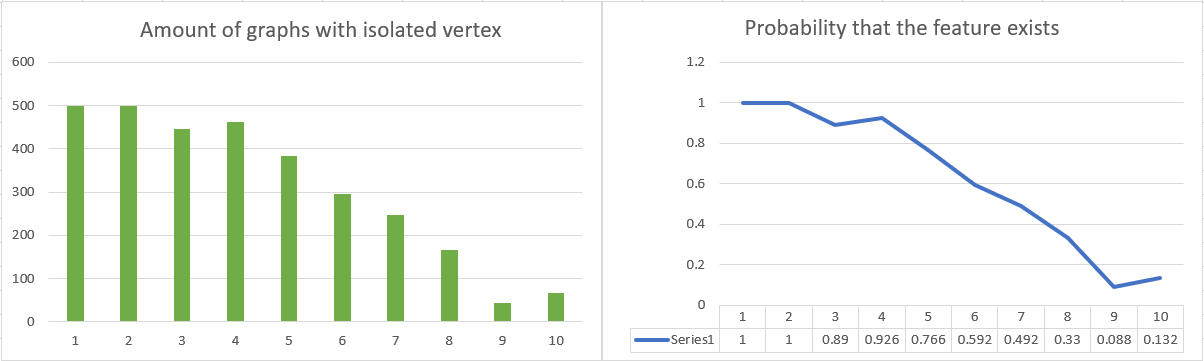
Simulation1B



Simulation2B



Simulation3B



כפי שניתן לראות בתכונה 1ו-2 הסיכויים נמוכים לפני הthreshold והסיכוי גבוהה אחרי הthreshold וככל שנתרחק הסיכויים האלו יתגברו (threshold נמצא בין 5 ל-6).

בתכונה 3 ניתן לראות באותו אופן שלפני הthreshold קיים סיכוי גבוהה לגרף בודד, ולאחר מכן הסיכוי קטנים.

**לסיכום**

ניתן לראות מהבדיקה שעשינו שאכן התכונות מתקיימות עבור גרפים אקראיים במודל ארניש -ריניה, וככל שנתרחק מהthreshold כך הסיכוי ייגבר להצלחת התכונה.

שמות קבצי קוד מצורפים :

main.cpp

functions.h

functions.cpp

Graph.h

Graph.cpp