חלק ג – מחקר

עבור חלק זה יצרנו סקריפט פייתון – Measurements.py שאחראי על הגדרת packet loss, הרצת תוכניות ואיסוף הנתונים, ספציפית בעזרת דגלי auto ו-format שיצרנו למטרה זו.

לא נוסיף הסבר לסקריפט Measurements.py כאן, קיימות בו הערות לעיונכם.

לאחר ההרצה אנחנו מקבלים את התוצאה הבאה:

A screenshot of a computer screen

Description automatically generated

נענה על השאלות שמועלות במטלה:

1. ניתן לראות ש-cubic נתן תוצאות טובות ב-packet loss נמוך וגבוה, במיוחד לפי רוחב הפס שהוא הגיע אליו – בערך פי 2 ללא אבוד ובערך פי 1.25 עם איבוד גבוה.

כמו כן ניתן לראות שבשילוב פרוטוקולים שונים מתקבלת תוצאה טובה יותר באיבוד נמוך אך הפער מצטמצם באיבוד גבוה, ב-5% וב-10% ניתן לראות שהמרכיב המשמעותי הוא הפרוטוקול של השולח דווקא.

1. מימוש RUDP שלנו משיג ביצועים קרובים ל-TCP באיבוד נמוך (בחלק מהמקרים) אך כשהאיבוד עולה הוא נשאר מאחורה, כנראה מכיוון שלא מימשנו flow control מתקדם ולכן כל חבילה שהולכת לאיבוד מכריחה המתנה של ה-timeout.
2. נעדיף RUDP במקרים של איבוד נמוך מכיוון שהוא מהיר יותר בשליחה בודדת, אך מכיוון שמנגנון flow control שלנו פשוט כאשר מגיעים לאיבוד גבוה עדיף TCP.