**מבוא מורחב למדעי המחשב – תרגיל בית מספר 1**

**מגישים:**

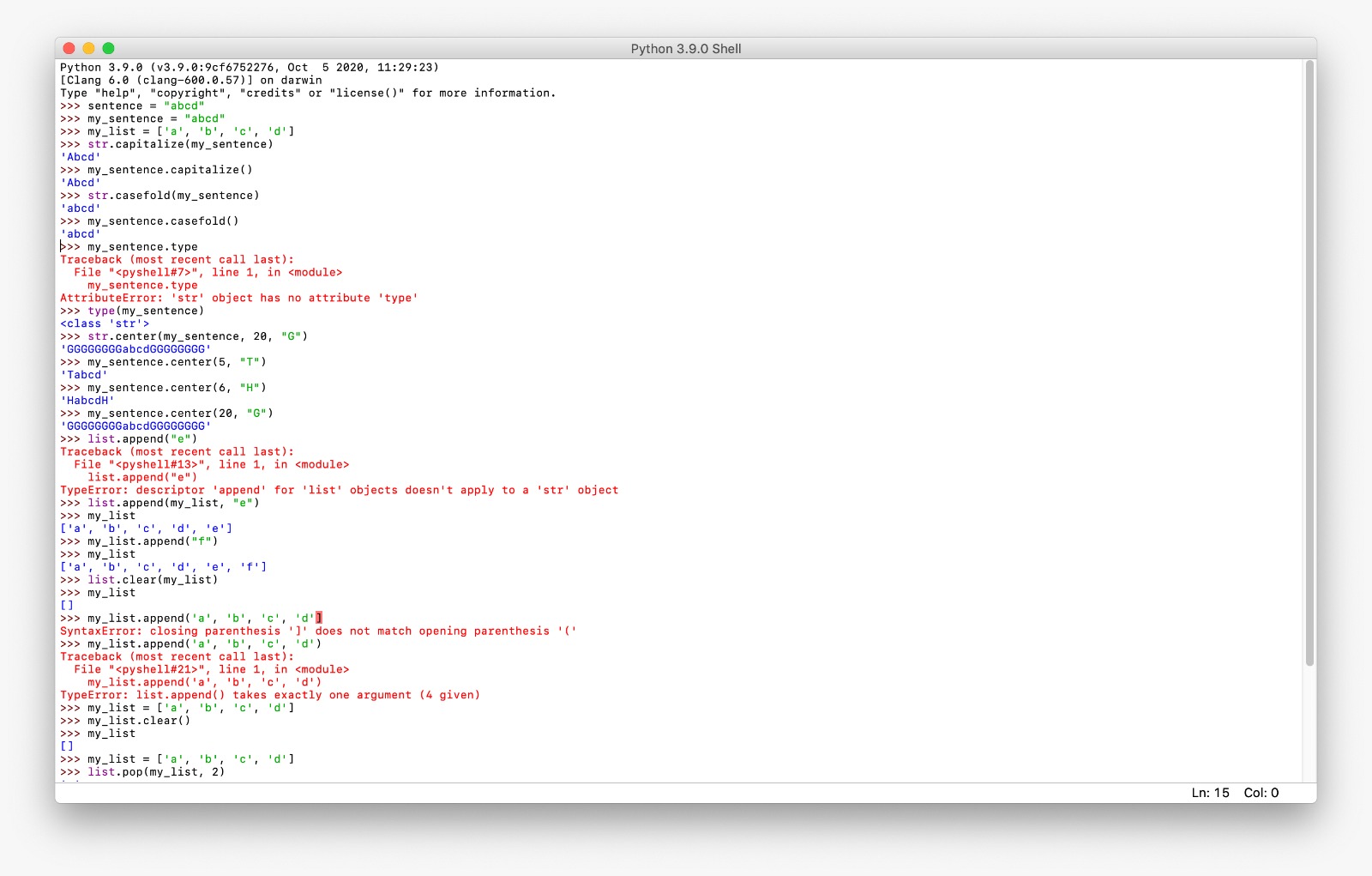
**Tomer Mildwoth – 316081355**

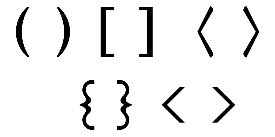
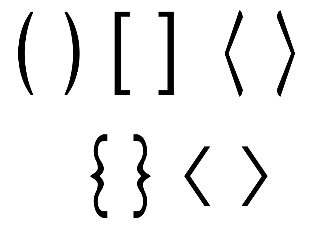
**Itay Hirschel – 315852566**

**Amir Liberman – 213468093**

**Asaf Yitzhaki – 206783532**

**שאלה 1**

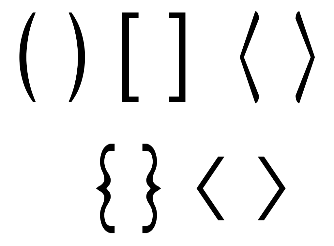
****



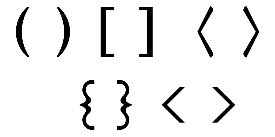
הגדרת משתנה מסוג string, והגדרת list-

שעליהם נעבוד

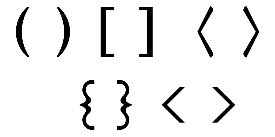
פונקצייתcapitalize() על המחרוזת

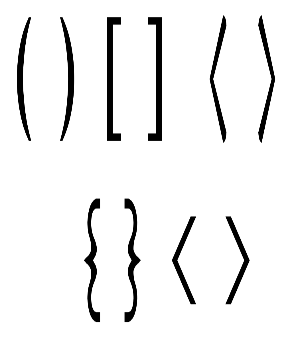
****

פונקצייתcasefold() על המחרוזת

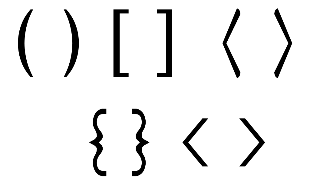


פונקצייתcenter() על המחרוזת

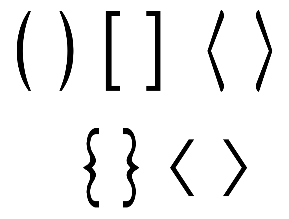


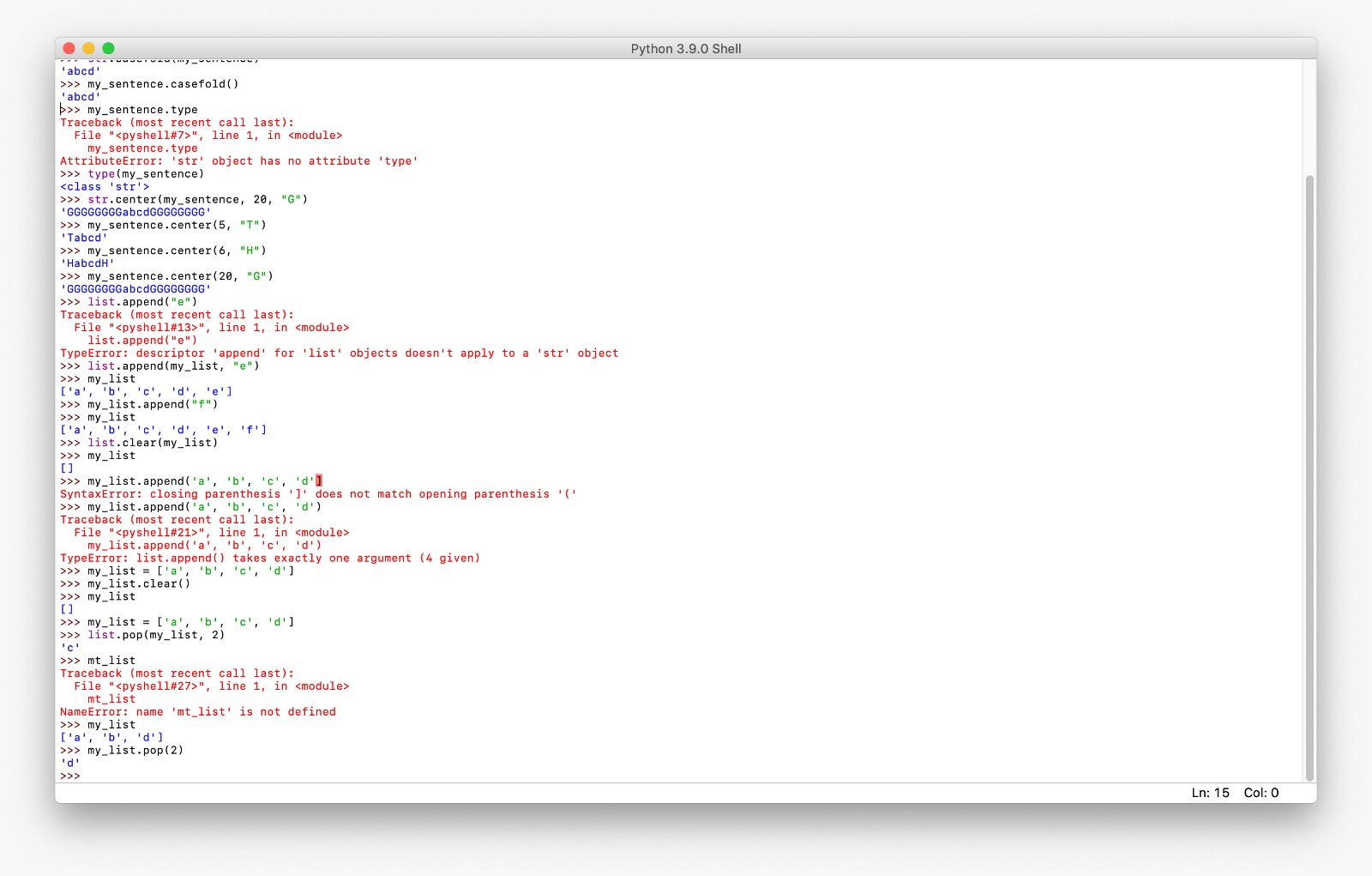
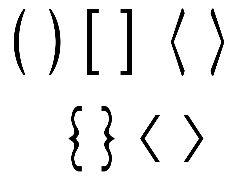
****

פונקצייתappend() על הרשימה

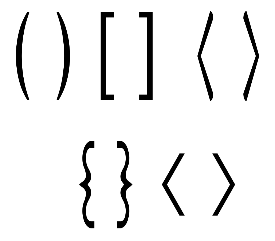
****

פונקצייתclear() על הרשימה

****

****

פונקציית pop()על הרשימה

****

**שאלה 2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| total | val | ID[i] | I | Iteration |
| 1 | 1 | ‘1’ | 0 | 1 |
| 5 | 2 | ‘2’ | 1 | 2 |
| 8 | 3 | ‘3’ | 2 | 3 |
| 16 | 4 | ‘4’ | 3 | 4 |
| 21 | 5 | ‘5’ | 4 | 5 |
| 24 | 6 | ‘6’ | 5 | 6 |
| 31 | 7 | ‘7’ | 6 | 7 |
| 38 | 8 | ‘8’ | 7 | 8 |

**ID = "12345678"**

**ID = "31608135"**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| total | val | ID[i] | I | Iteration |
| 3 | 3 | ‘3’ | 0 | 1 |
| 5 | 1 | ‘1’ | 1 | 2 |
| 11 | 6 | ‘6’ | 2 | 3 |
| 11 | 0 | ‘0’ | 3 | 4 |
| 19 | 8 | ‘8’ | 4 | 5 |
| 21 | 1 | ‘1’ | 5 | 6 |
| 24 | 3 | ‘3’ | 6 | 7 |
| 25 | 5 | ‘5’ | 7 | 8 |

**שאלה 3**

**סעיף א'**

|  |  |  |
| --- | --- | --- |
|  | sol 1 | sol 2 |
| 2^200 | 2.67E-05 | 9.00E-06 |
| 2^400 | 4.72E-05 | 9.40E-06 |
| 2^800 | 0.0001632 | 2.00E-05 |
| 2^1600 | 0.0005474 | 9.29E-05 |

לפי ניתוח הנתונים והגרף, ככל שהקלט גדול יותר, כך זמן הרצת הפונקציה עולה. בפתרון 1 קצב הגידול גבוה יותר מפתרון 2 וככל שהקלט גדול יותר, כך יעילות פתרון 2 מתבטאת באופן משמעותי יותר.

**סעיף ב'**

|  |  |  |  |
| --- | --- | --- | --- |
| number | sol 1 | sol 2 | sol 3 |
| 2^200 | 2.67E-05 | 9.00E-06 | 4.50E-06 |
| 2^400 | 4.72E-05 | 9.40E-06 | 4.60E-06 |
| 2^800 | 0.0001632 | 2.00E-05 | 7.80E-06 |
| 2^1600 | 0.0005474 | 9.29E-05 | 2.52E-05 |

הפתרון השלישי אכן יעיל יותר מבחינת זמן ריצה בהשוואה לשני הפתרונות הראשונים. קצב הגידול נמוך יותר משמעותית לאותם קלטים ביחס לפונקציות הקודמות.

**סעיף ג'**

נבדוק עבור קלטים באורך דומה עם כמות אפסים שונה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | sol1 | sol2 | sol 3 | number of zeros |
| 10^801-10 | 1.34E-03 | 1.05E-04 | 5.01E-05 | 1.00 |
| 10^801-3^788 | 1.43E-03 | 7.22E-05 | 5.94E-05 | 25 |
| 10^800 | 1.29E-03 | 1.01E-04 | 4.38E-05 | 800.00 |

נראה כי למספר האפסים בקלט אין השפעה משמעותית על זמן הריצה ביחס ליעילות הפונקציה ומספר הספרות בה. כך אנו רואים, כי מספר הספרות הכלול בקלט מהווה חלק משמעותי יותר בהשפעה על זמן הריצה מאשר ממספר האפסים לאותו מספר. כלומר, כמות האיטרציות שהפונקציה מבצעת היא המשפיעה העיקרית על יעילות הזמן ריצה.

**סעיף ד'**

הזמן שייקח להריץ את הלולאה הוא בערך 15^10\*4 בשנים. חישבנו את זה ע"י לקיחת מספר האיטרציות שלקח לנו להריץ לולאת for בפתרון 2 והזמן שלקח לנו לביצוע פעולה זאת. כך, חילקנו את 100^2 בכמות האיטרציות שלקח לנו בפתרון 2 עבור 200^2, והכפלנו בזמן ההרצה של פתרון 2 עבור 200^2 ולבסוף המרנו משניות לשנים.

בשונה, מהפתרון השני המספר כאן אינו מומר למחרוזת ובקוד זה הלולאה רצה על כל המספרים עד המספר הנתון, לעומת הפתרון השני בו הלולאה רצה על כל ספרה.

מהסעיפים הקודמים הסקנו שזמן הרצת הלולאה תלוי בעיקר בכמות האיטרציות ולכך הפער משמעותי - כי כמות האיטרציות גדולה משמעותית.