

## תשובות חלק א:

סעיף א' / שאלה 5

### DivideLogFile:

**סיבוכיות זמן- $O(n)$ :** חלוקת הקובץ לרשימה של רשימות  
מחרוזות סה"כ מעבר על הקובץ פעם אחת כאשר  $n$  הוא אורך הקובץ.  
**סיבוכיות מקום –  $O(n)$ :** רשימת הרשימות באורך  $n/size$  \* רשימה באורך  $n=size$

### Foreach(Frequencycounting):

**סיבוכיות**

**זמן- $O(n)$ :** מעבר בכל איטרציה על קבוצת שורות מתוך ה  $n$  הכללי והכנסה שלהם למילון לפי קוד השגיאה.  
סה"כ מעבר על כל שורה פעם אחת והכנסה למילון. כל פעולות ההכנסה  $O(1)$ : ContainsKey, Trim, Split =  
מעבר על כל השורות.  $O(n)$  =

**סיבוכיות מקום –  $O(n)$ :** רשימת מילונים שסה"כ האורך שלהם יחד הוא  $n$  הכי הרבה במידה שלכל שורה יש קוד שגיאה שונה.

### SumAllErrors:

**סיבוכיות**

**זמן –  $O(n)$ :** מעבר על כל המילונים והחזרה של מילון אחד עם סך כל השכיחויות, הכי הרבה במידה ולכל שורה יש קוד שגיאה שונה  $O(n)$  =  
**סיבוכיות מקום –  $O(n)$ :** מילון שהאורך שלו הכי הרבה  $n$  במידה ושלכל שורה יש קוד שגיאה אחר .

### PrintTopN:

**סיבוכיות זמן  $O(n \log n)$ :** מיון המילון הכללי לפי מספר השכיחויות.

הדפסה של  $N$  השכיחויות הגדולות, מקסימום  $N=n$ .

**סיבוכיות מקום –  $O(n)$ :** איחסון  $N$  הגדולים, במקרה הגרוע  $N=n$ .

סה"כ

**סיבוכיות זמן**  $O(n \log n) = O(n) + O(n \log n) + O(n) + O(n) + O(n)$  :

סה"כ

**סיבוכיות מקום**  $O(n) = O(n) + O(n) + O(n) + O(n)$  :

**סעיף ב:**

**שאלה 3:** אגדיר משתנה מסוג Datetime שהוא יוודא את השעה הנוכחית,

בנוסף אגדיר 2 משתנים count,sum שהם יעלו את המונים בהתאם לנתונים כל עוד ההודעה שייכת לשעה הנוכחית אחרת יבצעו את פעולת הממוצע ויוסיפו את הנתון לקובץ.

הפתרון בהנחה שהמידע מגיע לפי סדר כרונולוגי, אחרת אפשר להסתמך על הפתרון הקודם שהוא להגדיר 2 מילונים שהמפתח זה התאריך והספרה של השעה והערך במילון אחד נספר כמות השורות לשעה זו ובמילון השני,

נצברים הערכים.