תשובות למשימות- הסבר במילים

חלק א'

סעיף א'

1. סיבוכיות זמן: פונקציית פיצול הקבצים: סיבוכיות כוללת של אורך הקובץ Log הגדול.

פונקציית ספירת כמות השגיאות בכל קובץ: סיבוכיות כוללת של מספר השורות בכל קובץ Log קטן.- מספר השורות של הקובץ הגדול לחלק למספר התהליכונים שיכולים לרוץ בו זמנית המחשב.

הפונקציה הראשית למציאת N השגיאות השכיחות: זימון 2 הפונקציות הקודמות- הסיבוכיות הגדולה היא O(L)- מספר השרות בקובץ הגדול, ובנוסף שליפת N השגיאות השכיחות – O(MlogM) M הוא מספר סוגי השגיאות- מיון המילון על מנת לשלוף את N השכיחות.

1. סיבוכיות מקום:

ב MAIN יצרתי רשימה של N השגיאות השכיחות- O(N)

בפונקציה של פיצול לקבצים – בזיכרון נשמרת רק שורה אחת בכל רגע ולכן זה O(1) כי כל הקבצים נוצרים על המחשב עצמו.

פונקציה של ספירת השגיאות לכל קובץ: O(E) – יצירת מילון למספר סוגי השגיאות השונות.

בפונקציה הראשית למציאת השגיאות- יצירת מילון של כל סוגי השגיאות מכל הקבצים O(E) וכן יצירת מילון זמני של שגיאות לכל קובץ ויצירת רשימה של N השגיאות O(N). סה"כ: O(N+M).

סעיף ב'

3. הייתי יוצרת מילון ראשי שהמפתחות שלו יהיו התאריכים של הימים עם שעות עגולות, והערכים בתוכו:

אובייקט שמכיל 3 תכונות:

* משתנה SUM
* משתנה COUNT
* מילון שמפתחותיו יהיו הדקות והשניות והערכים בתוכו יהיו אובייקט שיכיל כל שורה שהתקבלה, שדות: .value, Timestamp ובנוסף 2 משתנים: SUM ן COUNT לטיפוליות.

וכך בכל שורה שהייתה מתקבלת, ראשית הייתי בודקת אם התאריך שלה -בפורמט שגוי או שהערך שלה , הוא לא מספרי- הייתי מתעלמת מהשורה הזו ולא מוסיפה אותה למאגר.

ואם זה היה תקין, אז הייתי לוקחת את הדקות שלה ובודקת אם קיים ערך כזה במילון, אם כן נכנסת אליו ונכנסת למילון הפנימי שלו- של הדקות, ובודקת אם קיים ערך זהה למשתנה הדקות הנוכחי, אם כן – זה אומר שיש כפילויות בתאריכים ולכן נעשה ממוצע בין כל הValue .

נוסיף אותו לSUM ונעלה את הCOUNT ונעשה ממוצע ביניהם- ונכניס את הממוצע לValue של הרשומה הזו.

אם לא קיים ערך כזה של הדקות, נוסיף אותו כמפתח במילון ונשנה את המונה והסכום בהתאם לערכים שהתקבלו.

לאחר כל הוספת רשומה נעלה את המונה ואת הסכום הכוללים של כל שעה, על מנת שנוכל לחשב בזמן אמת את הממוצע של הערכים לפי שעות של ימים.

וכך בכל רגע נתון ניתן לחשב את הממוצע לפי שעות- לבדוק אם הCOUNT גדול מ 0 ולחלק את הSUM בCOUNT .

4. **היתרונות של פורמט Parquet לאחסון בזיכרון**:

**-** חיסכון במקום: בפורמט הזה הנתונים נשמרים באופן יותר חסכוני בגלל שהם דחוסים וזה תופס פחות מקום בזיכרון.

**-** טעינה מהירה בזיכרון: אפשר לגשת למידע שבעמודות בלי לטעון את כל הקובץ וזה מקצר את זמן הטעינה בזיכרון.

**-** שומר על סוג הנתונים: בפורמט הזה נשמרים הנתונים לא רק כטקסט אלא כמספר , ערך בוליאני תאריך וכו ואז לא צריך לשנות את הנתונים לאחר שטוענים אותם לזיכרון

כשזה נתונים גדולים- הפורמט הזה נח לשמירת הנתונים באופן חסכוני וחכם.

חלק ב'