

ראשית, ביצירת הטבלה אנחנו נבנה שני מערכים של mutex אחד שאחראי על השורות, ואחד אחראי על העמודות.

בנוסף לכך, יש mutex שמוודא שפעולת הוספת עמודה והוספת שורה הן פעולות אטומיות ולא יהיו שתי שורות שיתווספו באותו הזמן.

נשמור גם ערך עבור מספר השורות ומספר העמודות בטבלה.

לבסוף נייצר דגל גלובלי שאומר האם הוצבה מגבלה עבור כמות החיפושים המקבילים על הטבלה וגם נייצר Semaphore בסיסי שרק "שומר מקום" כרגע.

פונקציות getCell ו setCell לא מסובכות לכן אין מה להרחיב בהן.

:searchString

ראשית נבדוק האם הדגל LimitSet עלה, במידה וכן צריך לעשות WaitOne. נעבור על כל התאים בטלה, ונראה האם הערך שאנחנו מחפשים קיים בטבלה. במידה והדגל אכן עלה, נשחרר את Semaphore.

:exchangeRows

ראשית ננעל את שתי השורות המדוברות ע"י נעילתן במערך mutex של השורות. באמצעות משתנה זמני, נחליף בין מיקום הערכים בתאים של השורות המדוברות. לבסוף נשחרר את הנעילה.

*האלגוריתם הזה עבור exchangeCols.

:searchInRow

ראשית נבדוק האם הדגל עבור LimitSet עלה ונפעל בהתאם. לאחר מכן ננעל את השורה שבה אנחנו מחפשים במערך mutex של השורות ונבדוק האם הערך קיים בשורה.

במידה והערך קיים, נגדיר את col להיות הערך של העמודה שבה מצאנו את הערך.

*האלגוריתם הזה עבור searchInCol.

:addRow

ראשית ננעל את mutex add_row בכדי שהפעולה תהיה אטומית. נייצר Spreadsheet חדש שבו יש שורה אחת יותר מהמקורי, כמו כן ניצור גם מערך חדש של מנעולים עבור השורות, ונעתיק לתוכו את התכולה של המערך הקודם. נבקש נעילה עבור כל השורות אחרי השורה שאנחנו מוסיפים תחתיה שורה חדשה. עבור כל השורות עד השורה שאנחנו נוסיף תחתיה שורה חדשה (כולל), נעתיק את התכולה של התאים. כשנעבור את השורה, ונשאיר שורה אחת ריקה (החדשה) נעתיק את התכולה של כל שאר השורות. לבסוף נשחרר את המנעול על השורות, והמנעול של add_row.

*האלגוריתם הזה עבור addCol.

:setConcurrentSearchLimit

נגדיר עבור SearchLimit מנעול מסוג Semaphore חדש בגודל הנתון, ובנוסף לכך נעלה את דגל LimitSet.

:save

נבצע שמירה בסגנון csv, נעבור על כל הערכים בכל שורה, ובינייהם נכניס ", " ואז נרד שורה ונעבור לשורה הבאה.

:Load

נטען את קובץ csv ונייצר עבורו Spreadsheet חדש בגודל המתאים וכמו כן נייצר מערכים של מנעולים עבור השורות והעמודות.

סיכום מנעולים:

Semaphore:

מנעול אחד מסוג Semaphore שמטרתו היא לוודא שלא נעבור את כמות החיפושים המקביליים.

Mutex:

קיים מנעול עבור כל עמודה, ועבור כל שורה. כמו כן שני מנעולים אחד עבור addRow ואחד עבור addCol.

מההנחה שיש n שורות ו- m עמודות, נקבל כי סה"כ יש לנו $n + m + 2$ מנעולים מסוג Mutex.