רעיון 1

צריך לממש עץ AVL, שלכל בן יהיה מצביע לאבא.

צריך לממש מילון בעזרת AVL ככה שיהיה בו גם מצביע למשכורת הכי גבוהה.

מילון חברות ממומש בעזרת עץ AVL, ממוין לפי ה-ID של החברה.

לכל חברה יהיה מילון של המשכורות השונות של העובדים שלה ובכל מילון כזה יהיה מילון של העובדים עם המשכורת הזו ממויין לפי ID. וגם יהיה לה מילון של עובדים שממויין לפי ID.

מילון כל העובדים הקיימים ממוין לפי ID, וגם מילון כל העובדים הקיימים ממויין לפי Salary.

העץ יהיה גנרי עבור איברים שונים

מילון של חברות שמעסיקות עובדים ממויין לפי CompanyID בשביל GetHighestEarnerInEachCompany. מה הקטע? אנחנו חשבנו שבגלל שהסיבוכיות של להוציא עובד מהמערכת היא , אז זה דופק אותנו כי אין שם שום דבר לגבי k. אבל יצאנו טמבלים ובעצם בעץ הזה יכולות להיות מקסימום חברות, לכן, אפשר להוציא משם בlog(n).

רעיון לסיור בפונקציה GetNumEmployeesMatching:  
אנחנו מוצאים את המינימום והמקסימום. מתחילים מהמינימום. כל עוד הוא בטווח אנחנו בודקים אם קיים לו בן ימני. אם כן אנחנו עושים עליו סיור In-order, אחרת נעלה לאבא. אם עלינו מבן ימני נמשיך למעלה, אחרת נבצע על כל הימני. באותו אופן לשמאלי.

חברה

* ID
* Value
* Highest earner
* NumEmployees
* Employee\_Dict

עובד

* ID
* Company\_ID
* Salary
* Grade