מבני נתונים 1

234218

212148894

סאלח חסאן

הוגש ע"י:

כולל בונוס הדפסה:

לפני בונוס הדפסה:

נא להחזיר לתא מס':

ציון:

שם

שם

מספר זהות

מספר זהות

1

תרגיל רטוב

מספר

המבנה שלנו יורכב מהמבנים הבאים:

Class Employee

{

Int ID;

Int salary;

Int grade;

Int group\_id;

}

Class Company

{

Int ID;

Int value;

Int num\_of\_employees;

Shared\_ptr<Employee> highest\_earner;

AVL employees\_by\_grade;

AVL employees\_by\_id;

}

Class Industry

{

Int num\_of\_workers;

Int num\_of\_companies\_with\_employees;

Shared\_ptr<Employee> highest\_earner;

AVL workers\_by\_id;

AVL workers\_by\_grade;

AVL companies;

AVL companies\_with\_employees;

}

**Init() –**

נקצה מבנה של Industry, אשר ניתן לבצע זאת ב O(1) כך :

1. ניצור 4 עצי AVL ריקים כפי שראינו בהרצאה ב O(1).
2. נאתחל את כל7 השדות של Industry ב O(1) אשר בהתחלה המספרים מאותחלים לאפס ו champion מאותחל ל nullptr.ו נאתחל כל אחד אחד מהעצי AVL בעץ ריק שיצרנו למעלה. ובסוף נעשה cast ל void\* ונחזיר את המבנה כנדרש ב O(1).

**AddCompany()-**

נבדוק שהנתונים שקיבלנו תקינים. אחרת, נחזיר שגיאה מתאימה מסוג INVALID\_INPUT.

נחפש את המזהה של החברה שקיבלנו בפונקצייה בתוך עץ החברות שיש לנו בתוך מבנה של Industry ב O(log(k)) ואם נמצא חברה קיימת עם מזהה זה נחזיר שגיאת FAILURE. אחרת, נקצה node חדש עבור החברה החדשה ו נכניסו לעץ ב O(log(k)) כפי שנלמד בכיתה,אם נתקלנו בשגיאות הקצאה נחזיר ALLOCATION\_ERROR אחרת נסיים הפעולה ב O(log(k)) ונחזיר SUCCESS.

**AddEmployee()-**

נבדוק שהנתונים שקיבלנו תקינים. אחרת, נחזיר שגיאה מתאימה מסוג INVALID\_INPUT.

נחפש את העובד בעץ DS->workers\_by\_grade אשר לוקחת O(log(n)) כפי שנלמד בכיתה.

ונחפש את החברה בעלת מזהה CompanyID בתוך העץ של

DS->companies אשר פעולה זאת לוקחת O(log(k)) ונסמן חברה זאת ב company\_to\_find.

אם לא קיימת חברת זאת או אם קיים העובד הזה אז נחזיר FAILURE.

אחרת, נקצה עובד חדש בעל הנתונים שקיבלנו מהפונקציה ונוסיף העובד לעצים הבאים:

1. DS->workers\_by\_id פעולת זו לוקחת O(log(n)) כפי שראינו בכיתה.
2. DS->workers\_by\_grade פעולת זו לוקחת O(log(n)) כפי שראינו בכיתה.
3. Company\_to\_find->employees\_by\_grade פעולת זו לוקחת O(log(n)) כפי שראינו בכיתה.

אם חלה שגיאת זכרון בהכנסות נחזיר ALLOCATION\_ERROR.

ונבדוק אם העובד החדש הפך להיות highest\_earner של החברה בה הוא עובד או של כל המערכת ונעדכן המביעים בהתאם ב O(1).

אחר כך נבצע: (DS->num\_of\_workers)++,

Num = (company\_to\_find->num\_of\_employees)++

ונבדוק אם Num == 1. אם כן אז נוסיף את company\_to\_find לעץ DS->companies\_with\_employees פעולת זו לוקחת O(log(k)) כפי שראינו בכיתה. אחרת, לא נעשה כלום.

נחזיר SUCCESS בסוף.

בסה"כ ביצענו הפעולה ב O(log(k)+log(n)) כנדרש.

RemoveEmployee()-

נבדוק שהנתונים שקיבלנו תקינים.אחרת, נחזיר שגיאה מתאימה מסוג INVALID\_INPUT.

נחפש את העובד בעל מזהה EmployeeID בתוך DS->workers\_by\_id ב O(log(n)) ונסמן עובד זה ב

RemoveCompany() –

GetCompanyInfo() –

GetEmployeeInfo() –

IncreaseCompanyValue() –

PromoteEmployee() –

HireEmployee() –

AcquireCompany() –

GetHighestEarner() –

GetAllEmployeesBySalary() –

GetHighestEarnerInEachCompany() –

GetNumEmployeesMatching() –

Quit() –