

به نام خدا

تحلیل زمانی بخش پیدا کردن کانکشن ها / پروژه نهایی / علی ابراهیمی / 993613001

در این پروژه از یک HashMap (در کلاس LinkedIn) برای نگاشت هر User به Id اش استفاده شده است. هر User نیز خود یک HashSet از connectionId ها دارد. بنابراین با داشتن id می توان در $O(1)$ به خود یوزر و همینطور ست کانکشن هایش دسترسی پیدا کرد.

تابع زیر، کانکشن های user را تا n درجه پیدا کرده و درون مپ connections قرار می دهد. در این مپ کلید-یوزر و ولیو-درجه است.

```
private void getDegrees(User user, HashMap<User, Integer> connections, int degree, int n) {  
    if (n == 0)  
        return;  
  
    for (String id : user.getConnectionId()) {  
  
        User connection = users.get(id);  
        Integer deg = connections.get(connection);  
  
        if (deg == null || degree < deg) {  
            connections.put(connection, degree);  
        }  
  
        getDegrees(connection, connections, degree: degree + 1, n: n - 1);  
    }  
}
```

این تابع بازگشتی است. هر بار ست کانکشن های user پیمایش می شود. به کمک مپ، هر بار یک کانکشن را در $O(1)$ بررسی میکنیم. شرط بودن نشان میدهد برای اولین بار این کانکشن را visit کرده ایم. شرط $degree < deg$ بررسی میکند درجه واقعی کانکشن کمترین مقدار باشد.

سپس همین روند را برای تمامی کانکشن ها در n لول تکرار میکنیم. اگر تعداد این افراد را n فرض کنیم (این n با پارامتر ورودی تابع متفاوت است)، هر کاربر در $O(1)$ درجه آن بررسی می شود. بنابراین پیچیدگی کلی $O(n)$ خواهد بود.