

Programlama Laboratuvarı III. Proje Ödevi

Tuna Kömür
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli, Türkiye
kmrtuna9@gmail.com

Osman Aldemir
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli, Türkiye
osmanaldir@gmail.com

Özetçe—Birçok kullanıcı ve özellikleri ile birlikte oluşturulmuş json dosyasını okuyup her kullanıcının kendi "tweet" özelliği içindeki en çok geçen geçen kelimeyi, o kullanıcının ilgi alanı kabul ederek ortak ilgi alanına sahip kullanıcıları graph yapısı üzerinde eşleştirilip gösterildiği bir proje geliştirilmiştir.

I. AMAÇ

Bu projenin amacı birden fazla kullanıcının tweetlerinden, takipçilerinden ve ortak ilgi alanlarını bulup ortaklık sayesinde kullanıcıları eşleştiren bir programdır. Akademik olarak da veri yapılarını öğreten hepsini kendimiz yazarak pekiştirmemizi sağladı.

A. Başlamadan Önce

Yazılım projemize başlamadan önce ciddi bir araştırma serüveninden geçtik. Hangi veri yapısı daha uygun hangisini kullanıp kullanmayacağımızı karar vermek çok zamanımızı aldı.

Bu süreçte birçok internet sitesi ve projede kullandığımız yazılım dilinin kurslarından faydalandık. Çeşitli araştırma ve fikir geliştirmelerin sonunda ilerleyiş stratejimizi belirleyerek projemize başladık.

B. Başlangıç

Projenin ilk aşamasında kendimize ait bir data seti hazırladık. Sonrasında hangi veri yapılarını kullanacağımıza karar verdik ve gerekli sınıfları tanımlayarak projenin isterlerine hazır hale getirdik.

C. İlerleyiş

Projenin gelişme sürecine bakacak olursak ilk olarak kendimize ait olan data setinin yazıldığı Json dosyasından datalar doğru mu diye terminal ekranına bastırdık. Her bir kullanıcının bilgilerini Json dosyasından aldıktan sonra kendi oluşturduğumuz ArrayList'te tuttuk.

Oluşan userlara erişmek içinde userlarımızı "Key", "Value" olarak hashmap ekledik. Sonrasında projenin diğer isterlerini yapmak içinde tekrar bir araştırma yaparak yeni algoritmalar geliştirdik.

II. GİRİŞ

Projenin ilk aşamasında hangi veri yapılarını kullanacağımızı karar vermekle geçti. Birden fazla kaynaktan internet sitesinden yararlandık. İlk olarak faker kütüphanesinden yararlanarak kendimize bir data seti hazırladık. Bu data setini de Json dosyasına yazdık.

Sonraki adımda ise Json dosyalarını okumak için Json kütüphanesinden yararlandık. Daha sonra isterlerdeki şartlara yerine getirmek için kullanacağımız veri yapılarının sınıflarını oluşturduk ve bu sınıfların birbiri ile bağlantılarını inceleyerek ilerideki adımlarımızı planladık.

Sınıflar:

Main: Bütün kodu burada çalıştırdığımız birbirinden gerekli metotları barındıran diğer classlardan nesneler oluşturduğumuz kodun temel yapı taşı olan class. Bu classta isterlerin hepsini gerçekleştiriyoruz.

HashMap: Bu class userlarımızı ve bilgilerini tutmaya yarayan bir class. Userlarımızı nasıl tutuyor derseniz HashMap classında bulunan put fonksiyonu ile tutuyoruz. "Key" ve "Value" şeklinde burada userlarını ve bilgilerini tutuyoruz.

myArrayList: Bu sınıfta Userları, userların tweetlerini, Followerslarını, Followinglerini bu sınıftan arraylist gibi çalışan bir nesne üretip burada kaydettik.

User: Her bir userın bilgisini tutan bir classtır. Userın ismini, kullanıcı adını, takipçilerini, takip ettiklerini, dilini, bölgesini, takipçi sayısını, takip ettiklerinin sayısını ve en son olarak da tweetlerinin isterlere uygun değişkenlere atıldığı classtır.

Graph: Kullancılar arası bağlantıyı göstermek için oluşturduğumuz bir classtır.

III. YÖNTEM

Projemizin ilk adımında Bir Data Seti oluşturduk. Data seti oluştururken de Faker kütüphanesinden yararlandık. Bu kütüphane bizde istediğimiz isterlere istediğimiz sayıda bir Data oluşturduk. Oluşturduğumuz bilgileri ya metin dosyasında tutacaktık ya da Json formatında tutacaktık. İkisinin farkına baktıktan sonra proje için bilgilerin Json fotmatında daha uygun olduğuna karar verdik. Artık elimizde hazır bir data seti vardı.

IV. KATKILAR

Tuna Kömür:

Kütüphane araştırma. Faker kütüphanesinin kurulumu, data seti üretmeyi Json formatından bilgi okuma, Json kütüphanesini kurma hangi veri yapılarının kullanılmasına karar verme Graph kodunu yazma. Takipçi ve takip edilen ilişkisini graph'da gösterme.

Osman Aldemir:

Kodun temel algoritmasını geliştirme. Hangi veri yapılarının kullanılması gerektiğini bulma. Arraylist, hashmap ve user klaslarını oluşturma, kullanıcıların ilgi alanlarına göre eşleme, belirli hashtag ve anahtar kelimeleri tweetlerde arama ve bulma. Uml diyagramı oluşturma.

V. SONUÇ

Sonuç olarak Faker kütüphanesi yardımıyla nasıl data seti kullanacağımızı öğrendik. Txt dosyası ile Json dosyası arasındaki farkı öğrendik. Json kütüphanesinin ne işe yaradığını öğrendik. Json dosyasından verilerin nasıl çekildiğini öğrendik. Arraylist veri yapısını kendimiz yazarak bilmediğimiz kullanımlarına hakim olduk.

Graph'ın, düğümün ve kenarın ne olduğunu anladık. Hashmap'de veri saklamayı ne için kullanıldığını ve bu verilere tekraradan nasıl ulaşacağımızı öğrendik. Kısacası bu proje bize önemli veri yapılarının işleyişini, ne amaçla kullanıldıklarına ve bize bir probleme hangi veri yapısını kullanırsak en kısa yoldan çözüme ulaşıracağımızı öğretti.

KAYNAKLAR

- [1] <https://fakerphp.github.io/>
- [2] https://www.w3schools.com/java/java_hashmap.asp
- [3] https://www.w3schools.com/java/java_arraylist.asp
- [4] <https://www.baeldung.com/java-graphs>
- [5] <https://ioflood.com/blog/json-to-java-object/#:~:text=Converting>

Diğer adımda bu data setini koda aktarıp okutmamız lazımdı. Bunun için de hazır olan Json kütüphanesini kurup bu adımı da başarıyla tamamladık. Okuduğumuz bilgileri bir for döngüsü içinde Json dosyasının uzunluğuna kadar döndürdük. Her okuduğu bilgiyi onu tutan bir değişkene atadık. Fakat takipçiler, takip edilen ve tweetler birden fazla string olduğundan dolayı onlar için for döngüsü içinde yeni bir for döngüsü açtık.

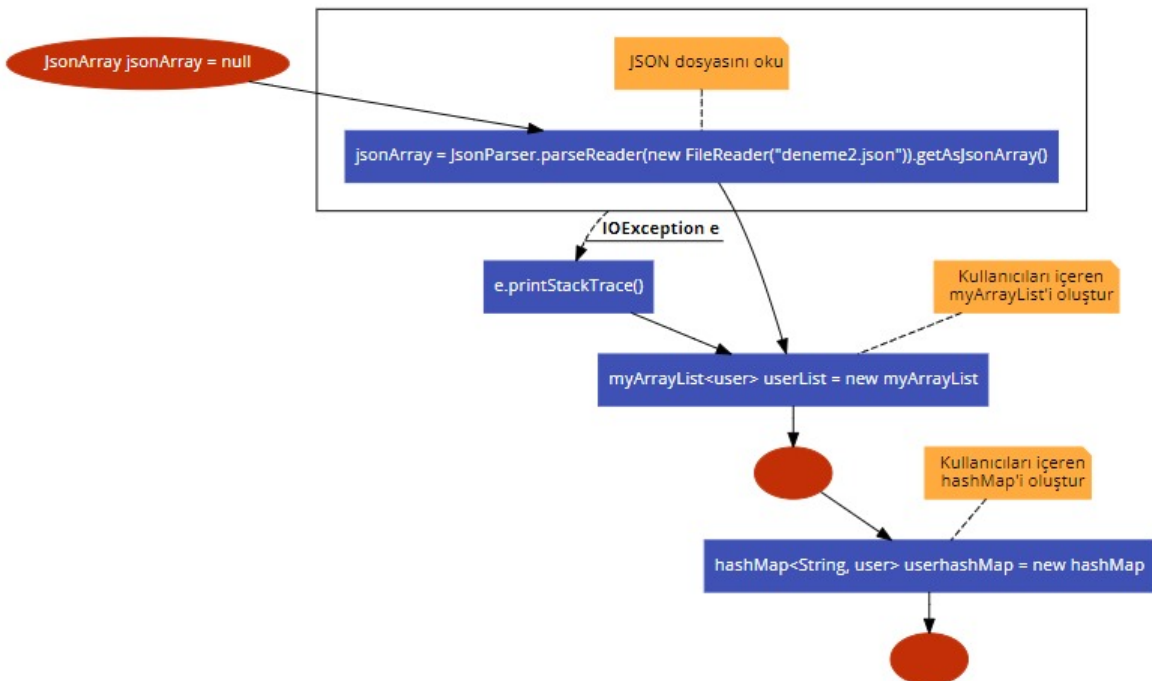
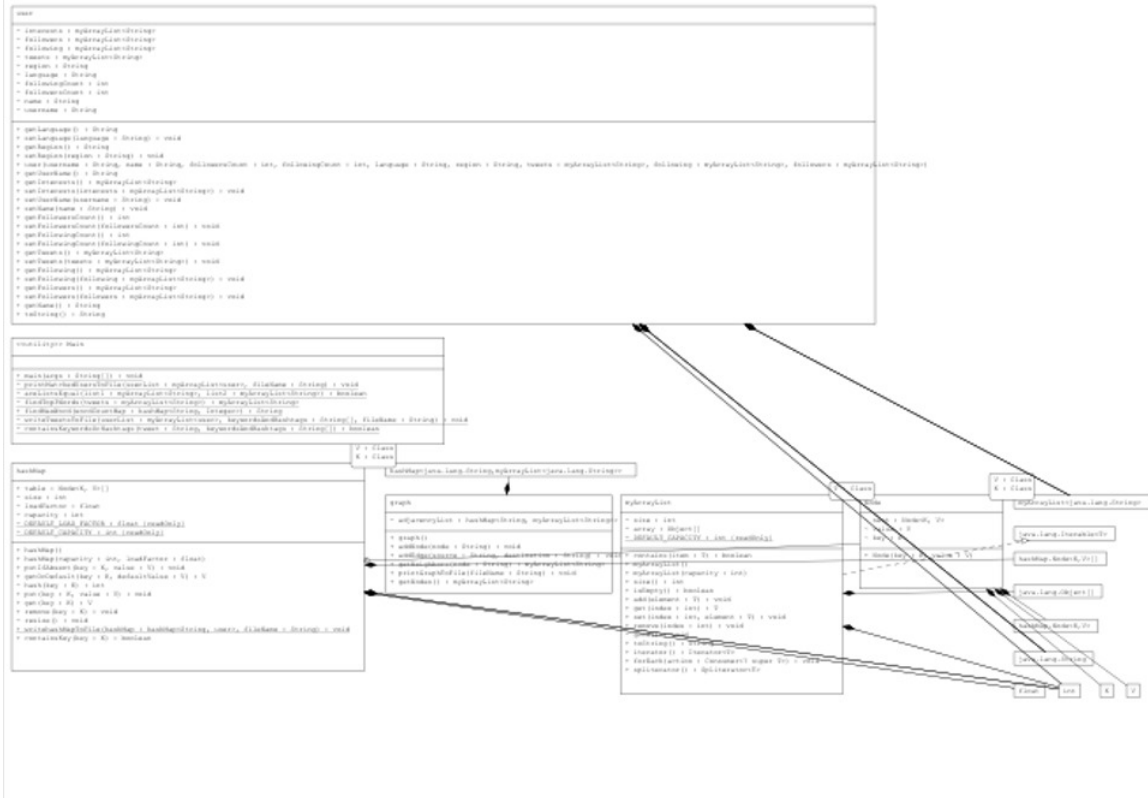
Tweet için myArraylist sınıfından String tipinde tweets, takipçiler için myArraylist sınıfından String tipinde followers, takip edilenler içinde myArraylist sınıfından String tipinde following adında üç yeni liste tanımladık ve her user için bunlar teker teker çalışıp bu verileri Arraylistte tuttuk. Bütün bilgileri aldığımıza göre artık bir user sınıfından bir user oluşturduk ve parametre olarak datadan okuduk bütün bilgileri gönderdik.

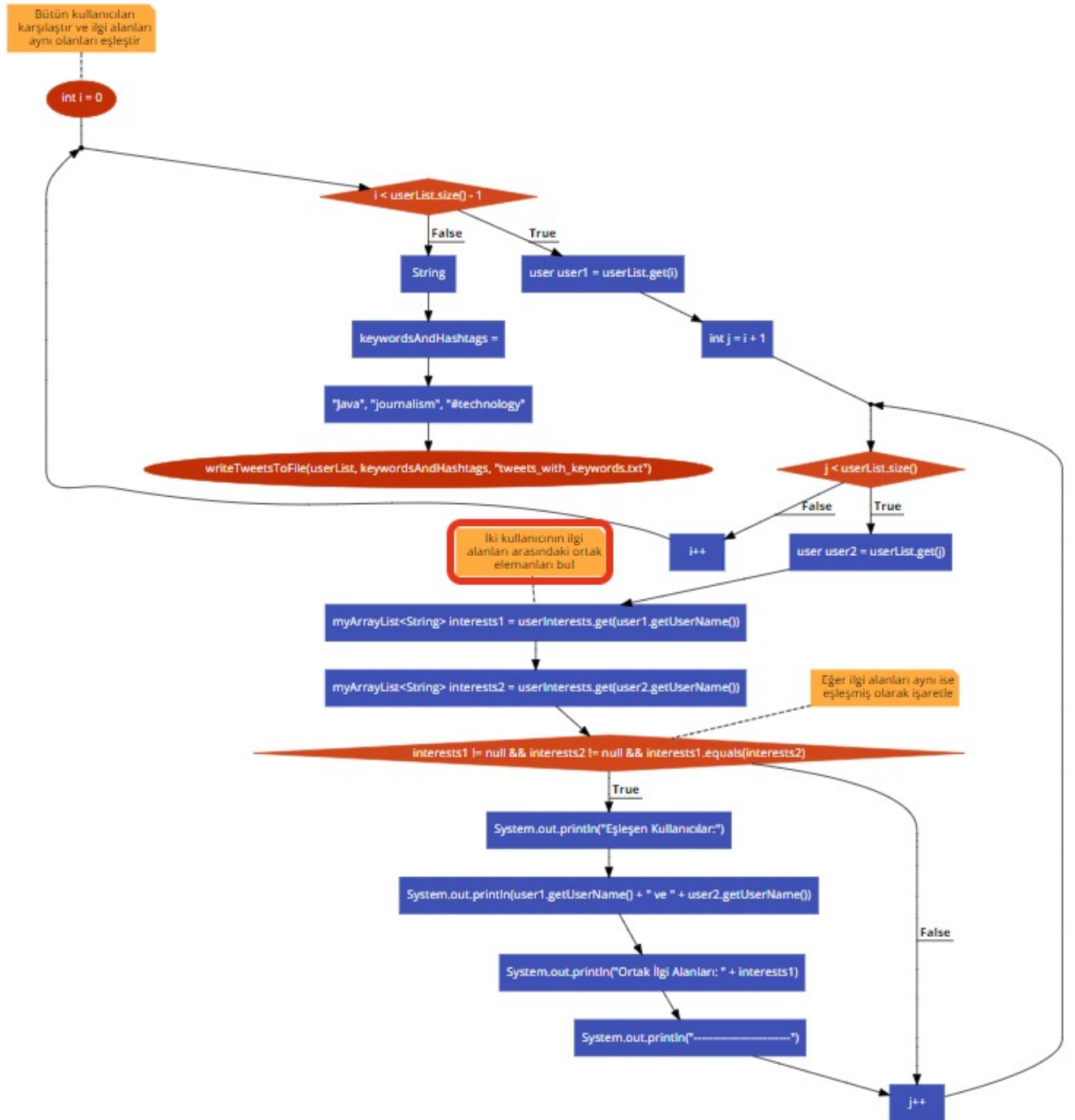
Sonraki adımda userları tutacak bir listenin olması lazımdı. Bunun için oluşturduğumuz myArrayList clasından user tipinde bir userlist oluşturduk. Daha sonra bu userlarıma istediğimiz zaman ya da bilgisine ulaşmak için hashmap sınıfından hashmap<String,user> userhasmap şeklinde bir userları tutacak hashmap nesnesi oluşturduk. Oluşturduğumuz hashmap'i görmek için fazladan bir metod ekledik ve bu metod hashmap'i bir txt dosyasına kaydettik bu sayede hasmap'i de göstermiş olduk.

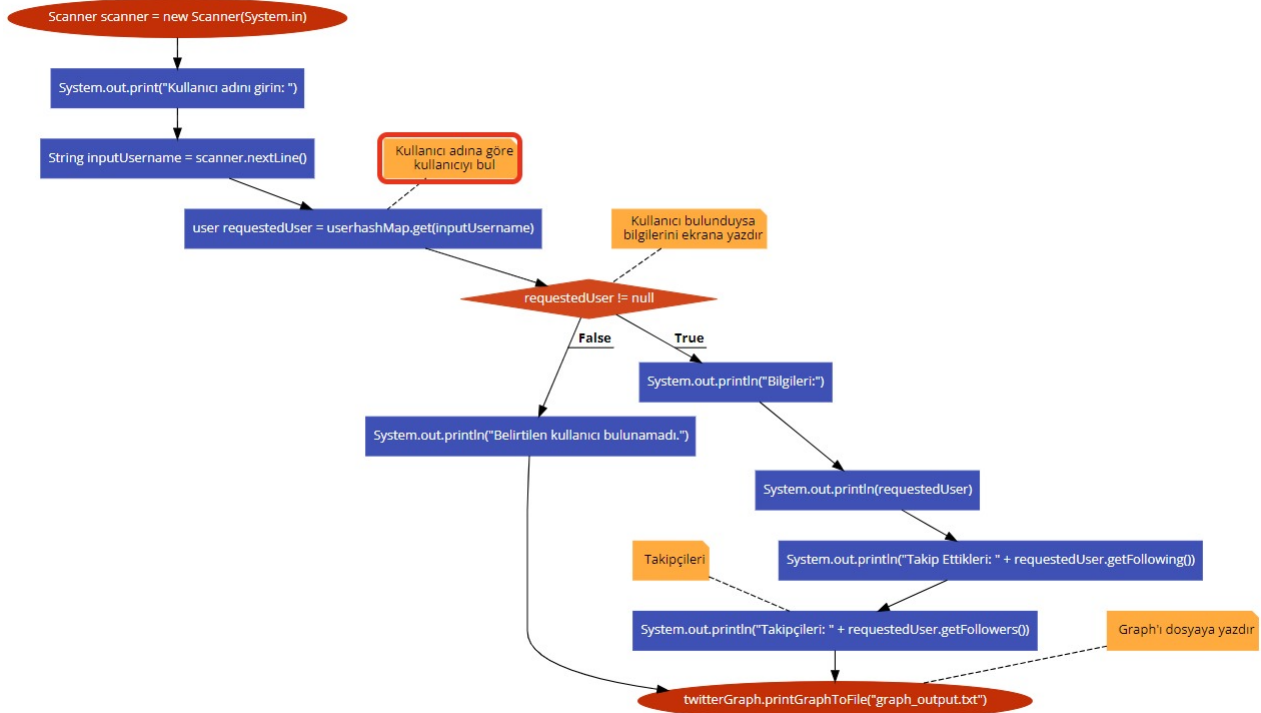
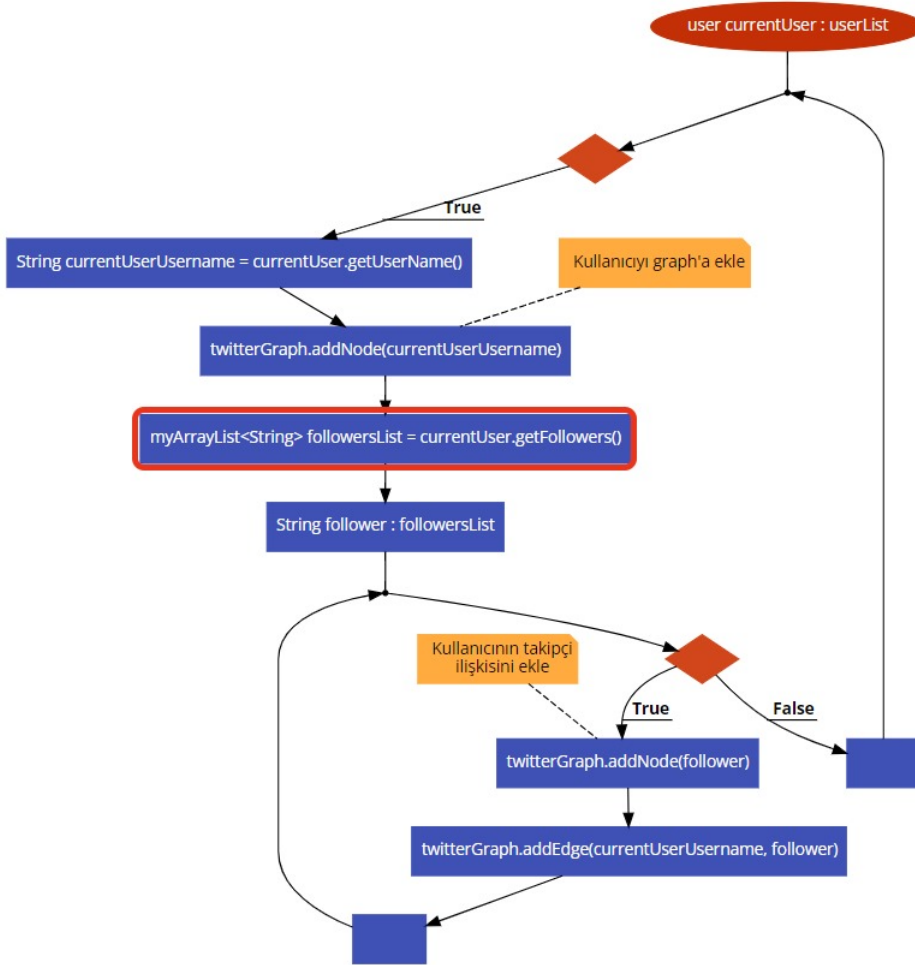
Projenin sonraki isteri kullanıcı nesneleri arasındaki takipçi-takip edilen ilişkilerini kullanarak bir graf oluşturulması kısmını yapmaya başladık. İlk olarak Graph sınıfında twitterGraph adında bir nesne oluşturduk. Eğer kullanıcılar arası bir ilişki olursa bunu Graph nesnesi içinde bulunan addNode metoduyla bir düğümde tuttuk. Graph'ta kimin kimle bağlantısını göstermek içinde yeni bir metod yazdık. Bu metod graph'ı bir txt dosyasına yazıyordu bu sayede de graphı ekranda görmüş olduk. Diğer bir adım ortak ilgi alanları olan kullanıcıları eşleştirmekti. Bunun için kullanıcıların tweetleri içinde en çok geçen 3 kelimeyi buldurduk.

Sonrasında 2 tane for döngüsü açarak eğer ilgi alanları eşit olan varsa bunları bir txt dosyasına yazdırmasını sağladık. Ayrıca kodu çalıştıran kişi hangi userın bilgilerine ulaşmak isterse onu terminale yazarak istediği userın bütün bilgilerine ulaşmasını sağladık. Kullanıcıların tweet içeriklerinde belirli anahtar kelimeleri ve hashtagleri içeren tweetler içinde mainde hastagleri ve anahtar kelimeleri oluşturduk. Eğer herhangi bir kullanıcının içinde bu hastagleri ve anahtar kelimeli bulursa o userın ismini ve bu kelimelerin geçtiği tweetleri teker teker txt dosyasına yazdırdım ve bu şekilde bu hastagleri ve anahtar kelimeleri içeren tweetleri görmüş olduk.

VI. DENEYSEL SONUÇ







VII. HATALAR

```
Eşleşen Kullanıcılar:
tuna41 ve hozgorkey
Ortak İlgi Alanları: [bir, ve]
-----
Eşleşen Kullanıcılar:
tuna41 ve gumuspala.sinem
Ortak İlgi Alanları: [bir, ve]
-----
Eşleşen Kullanıcılar:
osman67 ve eakay
Ortak İlgi Alanları: [bir, iin, ve]
-----
Eşleşen Kullanıcılar:
osman67 ve xkunter
Ortak İlgi Alanları: [bir, iin, ve]
```

```
Eşleşen Kullanıcılar:
stephencain ve eric85
Ortak İlgi Alanları: [journalism, journalism, journalism]
```

```
1 Eşleşen Kullanıcılar:
2 burcu06 ve eakay
3 Ortak İlgi Alanları: [bu, ve, bir]
4 -----
5 Eşleşen Kullanıcılar:
6 burcu06 ve akal.ada
7 Ortak İlgi Alanları: [bu, ve, bir]
8 -----
9 Eşleşen Kullanıcılar:
10 atakan.kocoglu ve eakay
11 Ortak İlgi Alanları: [ve, ve, ve]
12 -----
13 Eşleşen Kullanıcılar:
14 atakan.kocoglu ve xkunter
15 Ortak İlgi Alanları: [ve, ve, ve]
16 -----
17 Eşleşen Kullanıcılar:
18 atakan.kocoglu ve solmaz.derin
19 Ortak İlgi Alanları: [ve, ve, ve]
20 -----
21 Eşleşen Kullanıcılar:
22 atakan.kocoglu ve nabadan
23 Ortak İlgi Alanları: [ve, ve, ve]
24 -----
```