

# **KOCAELİ ÜNİVERSİTESİ BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMLAMA LAB. I- 3. Proje**

**Seda Nur EKİCİ-  
200201050 Mehmet  
Hüdayi YILDIRIM  
190201044**

## **Özet**

*Bu rapor Programlama Laboratuvarı 1 Dersinin 3.projesini açıklamak ve sunumunu gerçekleştirmek amacıyla oluşturulmuştur. Raporda projenin tanımı, isterleri, yapım aşaması kullanılan araç ve yöntemler, kod parçacıkları vb. Bulunmaktadır. Proje aşamasında yararlanılan kaynaklar raporun son bölümünde bulunmaktadır.*

## **• Giriş**

*Bu projenin amacı, verilecek olan excel dosyalarında bulunan kişi bilgilerini kullanarak soy ağacı oluşturmak ve bu soy ağacı üzerinden çeşitli işlemler yapmaktır. Proje kapsamında; birçok aileye mensup kişilere ait,*

*çeşitli bilgiler içeren 4 excel dosyası verilecektir. Verilen bu dosyalardaki ailelerin birbiriyle çeşitli şekillerde kesişimi olacaktır. Dosyaların program tarafından okunup içindeki bilgilerin kullanılarak kişiye ait soy ağacının oluşturulması projenin en temel hedefidir. Kişi için sınıf oluşturulacaktır. Kişi sınıfı tc-no (id), ad, soyadı, doğum tarihi, anne adı, baba adı, kan grubu ve meslek özellikleri, kızlık soyadı ve cinsiyet ile oluşturulacaktır. Excel içerisindeki her satır ağaçta bulunacak bir düğümü oluşturacaktır. İlk düğümde bulunan kişilerin anne ve babaları için ekstra bir düğüm oluşturulmayacaktır (ağaç ilk satırdaki kişilerden başlayacak).*

*Bu proje kapsamında soy ağacı kullanılarak bazı problemlerin çözülmesi istenmektedir. Çocuğu olmayan düğümlerin listesinin yaş sıralamasına göre kaydedilmesi istenmektedir. Kaydetme adımında yapılan sıralama (sort) işleminin adım adım gösterilmesi beklenmektedir. Üvey kardeşler bulunarak harf sıralamasına göre kaydedilmesi istenmektedir. Kaydetme adımında yapılan sıralama (sort) işleminin adım adım gösterilmesi beklenmektedir. Kan grubu A olanların listesi kaydedilerek gösterilmesi istenmektedir. Soyunda aynı mesleği yapan çocuklar veya*

*torunların gösterilmesi istenmektedir. (Baba mesleğini, dede mesleğini devam ettiren kişiler ve mesleğini devam ettirdiği ataları). Soy ağacında aynı isme sahip kişilerin ismi ve yaşları gösterilmesi istenmektedir.*

*Kullanıcıdan alınacak 2 tane isim girdisinden sonra büyük olan kişinin küçük olan kişiye yakınlığı gösterilmesi istenmektedir.*

*Kullanıcıdan alınan kişi bilgisi ile o kişiye ait soy ağacının gösterilmesi istenmektedir. Verilmiş olan excel dosyalarına ek olarak kişinin soy ağacında bulunanların kişi ile yakınlık derecelerinin yazılması beklenmektedir. Soy ağacının kaç nesilden oluştuğu bulunması istenmektedir. (Ağacın maksimum derinliği bulunmalıdır.)*

*Kullanıcıdan alınan isim girdisinden sonra o isimden sonra kaç nesil geldiğinin bulunması istenmektedir. Tüm isterler için en iyi ve en kötü senaryo karmaşıklık hesabı yapılması istenmektedir.*

*Soy ağacında istenilen isterlerin farklı bir renkle anlaşılır şekilde gösterilmesi istenmektedir. Ayrıca kaydedilen veri yapısının da gösterilmesi istenmektedir. Kişilerin cinsiyetine göre farklı renk veya şekillerin kullanılması istenmektedir.*

## • İlerleyiş ve Yöntem

## 1. Başlamadan Önce

*Bu projeye başlamadan önce proje dokümanında verilen bilgileri derleyip kullanabileceğimiz metodları araştırdık. Kütüphaneden ve internet üzerinden yararlanabileceğimiz kaynakları araştırdık. Javada iç içe excel sayfası okumayı, aile ağacını nasıl oluşturabileceğimizi araştırdık.*

## 2. Başlangıç

*Öncelikle bize verilen excel dosyasını sayfa sayfa okutmayı öğrendik. Apache.poi kütüphanesi yardımıyla her sayfayı tek tek okuduk. Sayfadaki her bir hücreyi tek tek ad, soyad, dogumTarihi, uyeninEsi, anneAdı , babaAdı, kanGrubu, meslek, evliBekar, kızlıkSoyadı ve cinsiyet özelliklerine atadık. Bu değerleri oluşturduğumuz ağaca ekledik. Switch-case döngüsü ile isterleri kullanıcının isteğine göre listelemesini sağladık.*

*Ağac classında arama ve ekleme fonksiyonlarını oluşturduk. Aile ağacını bu classta oluşturduk. AğacNode ve AileUyesi classlarında üyelerin ve aralarındaki akrabalık ilişkilerini tanımladık.*

## • Son Söz

*Bu proje bize çok fazla bilgi birikimi sağladı ve farklı bir bakış açısı kazandırdı. Projeyi yaparken*

*çok fazla araştırma yapıp  
eksiklerimizi kısa zamanda  
tamamladık ve projede bizden  
istenenleri elimizden gelen en iyi  
şekilde yapmaya çalıştık.*

## • Kaynakça

<https://github.com/karar78/networking>

<https://emrecelen.com.tr/java-treemap-nedir/>

<https://therenegadecoder.com/code/tree-data-structure-family/#java-tree-syntax>

<https://github.com/alfiememo/FamilyTree>

<https://nerdbook.wordpress.com/2018/03/28/agac-veri-yapisi/#:~:text=%C3%96rne%C4%9Fin%2C%20%C3%A7ocuk%2C%20karde%C5%9F%20d%C3%BC%C4%9F%C3%BCm%2C,%C3%B6zel%20a%C4%9Fa%C3%A7%20%C5%9Fekilleri%20de%20%C3%A7%C4%B1kar%C4%B1labilir>

[Apache POI Tutorial Part1 - How To Read Data From Excel #ApachePOI](#)

<http://tanertemelce.blogspot.com/2013/05/java-ile-excel-okuma-ve-excele-yazma.html>

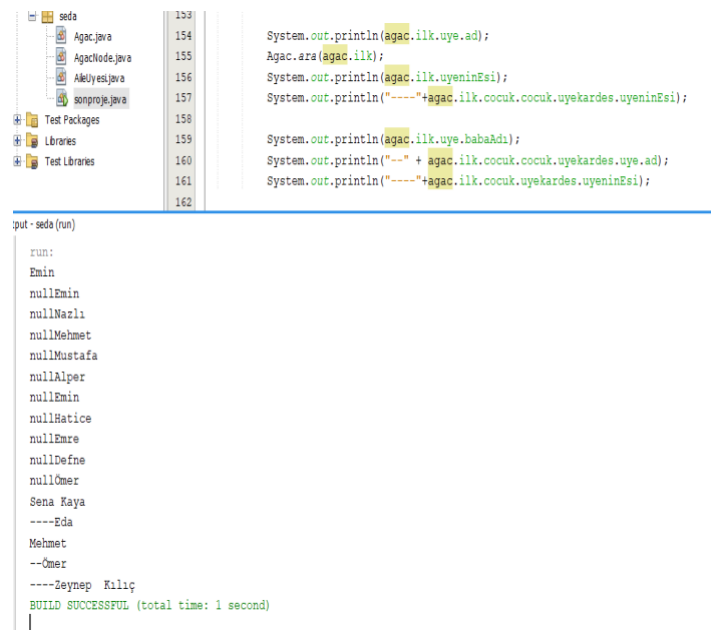
<http://java4f.blogspot.com/2014/05/apache-poi-java-ile-excel-den-veri-okumak.html>

## • Görüntüler

```
run:
hangi isteri istersiniz:
1:1.ister
2:2.ister
3:3.ister
4:4.ister
5:5.ister
6:6.ister
7:7.ister
8:8.ister
9:9.ister
10:10.ister
```

```
3
3. ister:
```

```
su kişinin SenaA(-)-- esitt
su kişinin NazlıA(-)-- esitt
su kişinin ÖmerA(+)-- esitt
su kişinin HaticeA(+)-- esitt
su kişinin İremA(+)-- esitt
BUILD SUCCESSFUL (total time: 2 seconds)
```



```
153
154 System.out.println(agac.ilk.uye.ad);
155 Agac.ara(agac.ilk);
156 System.out.println(agac.ilk.uyeninEsi);
157 System.out.println("-----" + agac.ilk.cocuk.cocuk.uyekardes.uyeninEsi);
158
159 System.out.println(agac.ilk.uye.babaAdi);
160 System.out.println("---" + agac.ilk.cocuk.cocuk.uyekardes.uye.ad);
161 System.out.println("-----" + agac.ilk.cocuk.uyekardes.uyeninEsi);
162

gput - seds (run)

run:
Emin
nullEmin
nullNazlı
nullMehmet
nullMustafa
nullAlper
nullEmin
nullHatice
nullEmre
nullDefne
nullÖmer
Sena Kaya
----Eda
Mehmet
--Ömer
----Zeynep Kılıç
BUILD SUCCESSFUL (total time: 1 second)
```

## •Akış Şeması

