**T.C.**

**CELAL BAYAR ÜNİVERSİTESİ HASAN FERDİ TURGUTLU TEKNOLOJİ FAKÜLTESİ**

VERİ YAPILARI DERSİ

PROJE ADI: MEZUN ÖĞRENCİ BİLGİ SİSTEMİ

Taner Yıldırım

153802052

Serkan Kumru

152802042

Turgutlu 2017

TESLİM TARİHİ : 26.05.2017

**1 – *Problemin Kısa Tanımı***

Mezun bilgi sistemi mezun olan öğrencilerin bilgilerini iş başvurularında aranan kriterlere göre sıralama, bilgileri iş yerine iletmek ve bu bilgilerin istenilen yeterlilik düzeyine göre kişileri kişi adına göre ikili ağaca atarak belirli bir düzen ve istenen kriterlere göre sıralamak , inorder , postorder , preorder olarak gezinme , eleman sayısını bulma ,derinliği bulma gibi işlemleri gerçekleştirmek.

***Kullanılan sınıfların ve metotların kısa açıklaması***

**Basvuru sınıfı:** İlana başvuru yapanlar ve başvuru yapan kişilerin uygunluğunu kontrol eder.

**Kişi sınıfı:** Bu sınıf içerisinde kullanıcı numarası ve kullanıcı bilgilerinin alınmasını sağlayan sınıf.

**Egitim sınıfı:** Bu sınıf içerisinde kişinin okuduğu okullar, okullara başlangıç ve bitiş tarihleri ve not ortalaması tutulur.

**Staj sınıfı:** Bu sınıf içerisinde kişinin iş deneyimlerini tutan sınıf.

**İlan sınıfı:** İlan sınıfı içerisinde şirkete ait iş ilanı özellikleri ve ilana başvuran kişiler vardır.

**Sirket sınıfı:** İlan sınıfı içerisinde şirkete ait bilgiler ve ilanla ilgili bilgiler tutulur.Ayrıca Hash Table’a eklenen her ilan için 10 boyutlu bir Heap sınıfı da tanımlanmıştır.Şirket ilanlarıda bulunur.

**HashEntry sınıfı:** Bu sınıf içerisinde Object tipinde bir veri, verinin anahtar ve değer olarak obje oluşturulduğu sınıftır.

**İkiliAramaAgaciDugumu sınıfı:** İkili Arama Ağacı sınıfı içerisinde ikili arama ağacına ait ve bizim sistemimiz için gerekli olan metotlar bulunmaktadır. Bu metotlar;

*ElemanSayisi:* Ağaçta olan kaç kişi olduğunu bulmamızı sağlar.  
*DugumleriYazdir:* Bu metot çağırıldığı yerdeki sıralamaya göre bütün kişileri yazdırır.   
*Ziyaret:* Bu metota sıralamalarda(InOrder vb.) root’u göndeririz.  
*InOrder:* InOrder gezinmeye göre ağaçta gezinmemizi sağlar.  
*Ekle:* Bu metot içerisinde gelen kişinin adına göre ağaca ekleme işlemi yapar.

*Güncelle:* Bu metot içerisinde ağaçta bulunan kişilerin bilgilerinin güncellenmesi işlemi yapar.  
*IsmeGoreListele:* Gelen ada göre ikili arama ağacında sınıf içerisinde tanımladığımız tipi private olan *IsmeGoreListeleInt* metodunu içerisinde çağırarak ağaç içerinde gezinerek ada göre işlemini gerçekleştirmeyi sağlar.

*NotOrtalamasiDoksandanBuyukler:* Bu metot kişinin not ortalaması 90 ve üstünde olan kişileri listeler.

ingilizceBilenler*:* Bu metot İngilizce bilen kişileri listeler.

*\* Yukarıdaki metotlar sadece sistemde kullanılan metotlardır.*

**LinkedListHashEnty sınıfı:** Bu sınıf içerinde tipi Object olan deger ve tipi int olan Anahtar üyelerini bulundurur. Ve constructor’unda gelen değerleri ilgili yerlere atama işlemi yapar.

**HashMap sınıfı:** Bu sınıf constructor’unda LinkedHashEntry tipinde dizi boyutu verilecek şekilde tanımlama yapılıyor. Sınıftaki metotlar;

*GetIlan:*Gelen key değerine göre size HashTabledaki şirket ve ilan bilgisini döner.

*AddIlan:* Gelen ilana göre hashing işlemi yapar ve tabloda o indis boşsa ekleme işlemini gerçekleştirir. Boş değilse next işlemi yaparak ekleme işlemini gerçekleştirir.

**HeapDugumu sınıfı:** İçerisinde Kişi tipinde eleman değişkeni ve double tipinde uygunlu puanı tutar.

**HeapSort sınıfı:** İçerisinde sort algoritmasına göre sıralama yapmaktadır.

**Heap sınıfı:** İçerisinde tipi HeapDugumu olan HeapArray isimli bir dizi tutar. Constructor’unda gelen değer dizinin boyutu olur. Burada Heap’le ilgili metotlar ve sistem için gerekli olan metotlar vardır. Bu metotlar;

*Insert:* Gelen kişiyi sırasıyla Heap’e eklememize yarar.

*MoveToUp:* Heap’e ağaç şeklini vermemizi sağlar.

*RemoveMax:* En yüksek değeri (uygunluk puanı en yüksek olanı) döndürmemizi sağlar.

*MoveToDown:* Yeni root’u belirler.

*DisplayHeap:* Heap’teki bütün değerleri listeler.

**KULLANILAN VERİ YAPILARININ AÇIKLAMASI**

HashTable, sistemde mevcut olan ilan numarası ve şirket bilgilerine kişilerin başvurmasını ve uygun kişinin işe alınmasını sağlamakta kullanıldı.

Heap, her iş ilanına başvuran kişileri ve uygunluk puanını tutmak için kullanıldı.

İkili Arama Ağacı, kişilerin ağaca eklenmesi ve gerekli ikili arama ağacı işlemlerinin yapılması için kullanıldı.

LinkedList, HashTable’ın bulunan indisi doluysa next yaparak gelen bilgileri eklemede kullanıldı.

**KULLANICI KATOLOĞU**

Program load olduğunda eleman.txt dosyasından kişiler ikili arama ağacına isim sırasına göre çekiliyor.Şirket.txt dosyasından da şirket bilgileri hash table’a id sırasına göre key üretilerek ekleniyor.

Form ekranındaki Kayıt Ol butonu ile yeni bir kişi ikili arama ağacına ekleniyor.

Güncelle butonu ile ağaçtaki kişi ismine göre bulunup güncelleme yapılıyor.

Sil butonu ile ağaçtaki kişi ismine göre bulunup silinme yapılıyor.

Form load olduğunda İşe Başvurma listboxı dolduruluyor. Kişi kullanıcı adı ve işin ıd sini girerek Başvuru butonu ile ilana başvurabiliyor.

İşe Başvurma listboxından seçilen ilana başvuranların tamamı heapten çekilerek İlana Başvuranların Listesi listboxında listeleniyor.

En Uygun Elemanı İşe Al butonu ile de uygunluk puanı en yüksek olan kişi Heapten çekiliyor.

İş yeri kaydı için Kayıt Ol butonuna basılıyor.İşyeri kaydı yapılırken ilan da aynı zamanda kaydediliyor.

Ortalaması 90 üstü olan kişiler ile İngilizce bilenlerin kodları ikili arama ağacında düzenlenmiştir.

**Ekran çıktıları:**





