

## BLM-305 İLERİ PROGRAMLAMA TEKNİKLERİ

### DÖNEM ÖDEVİ RAPORU

#### Proje:

**İsim:** File Manager

Proje ödevinde Java programlama dilini ve frameworklerini kullanarak her hangi bir konuda proje geliştirilmesi amaçlanmıştır.

Bu şartlar doğrultusunda seçtiğim konunun amacı:

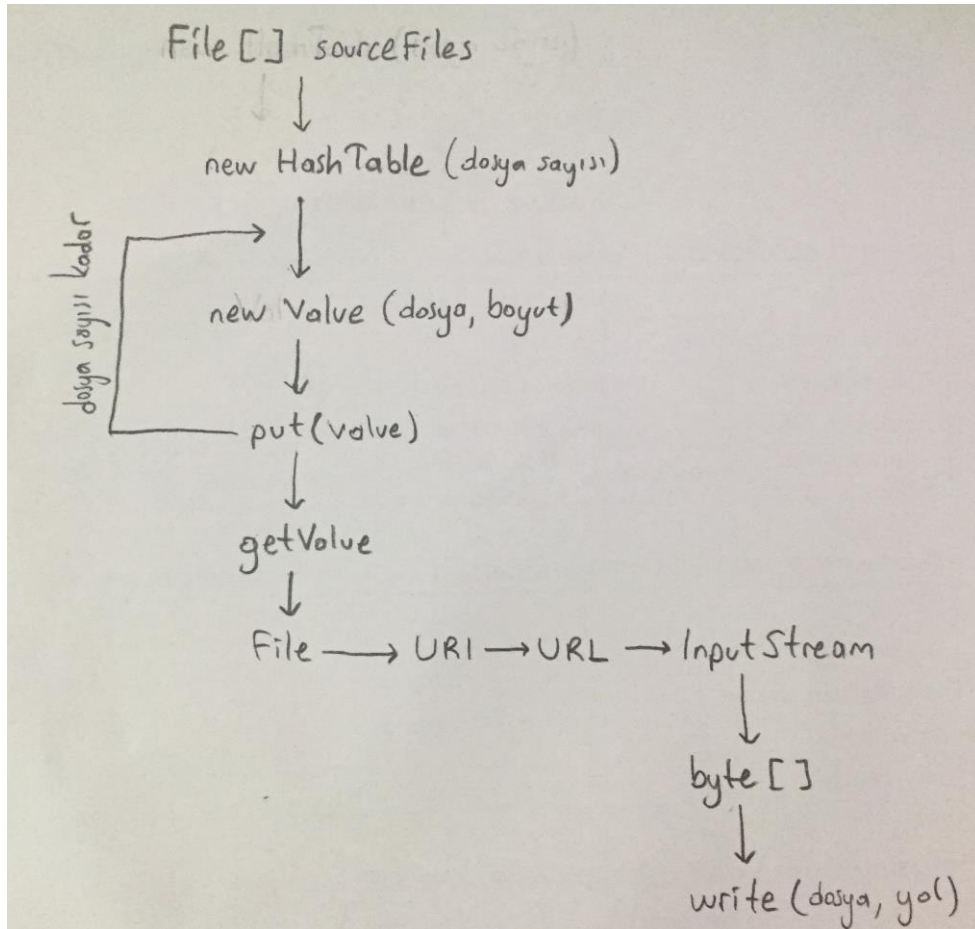
- Bir klasör içerisinde bulunan dosyalar arasında aynı dosyadan birden fazla bulunması olayının önüne geçmek ve hafıza kullanımını daha verimli hale getirebilmek.
- Bu fikir, eski telefonumdaki resimleri bilgisayara yedeklerken karşılaştığım söz konusu problemten dolayı ortaya çıktı. Aynı resimden birden fazla olması ve bu durumun, teker teker ayıklandığı takdirde büyük zaman kaybına yol açacağını bildiğim için böyle bir programın yararlı olabileceğini düşündüm.

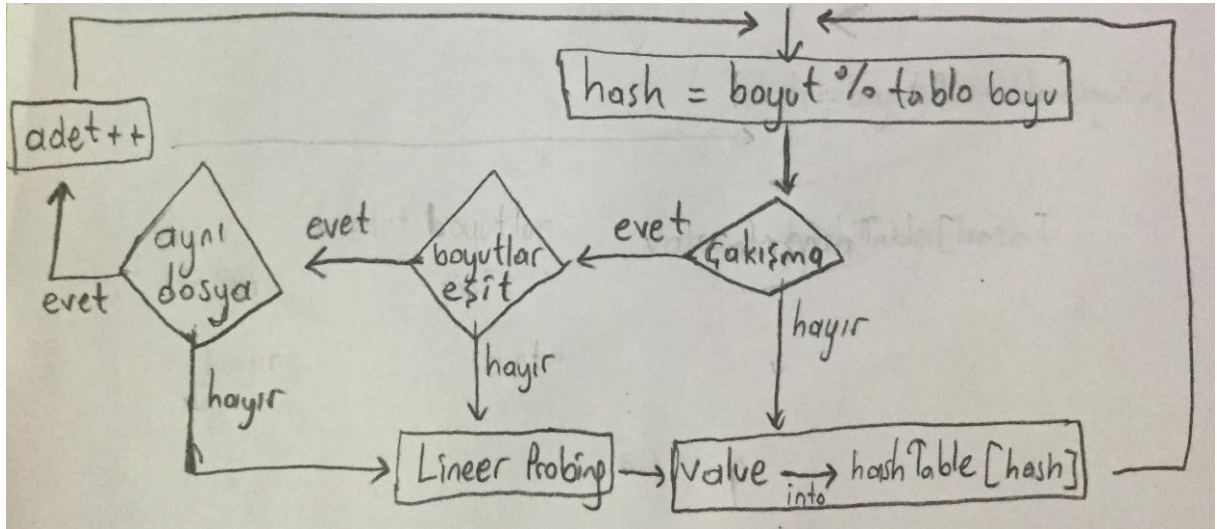
#### Çalışma Prensipleri

- Dosyaların düzenlenmesi ve saklanması için kaynak klasördeki dosyalar kadar uzunluğa sahip bir HashTable ( ) nesnesi oluşturulur.
- Kaynak klasörden okunan her dosya için bir Value( ) nesnesi oluşturulur.
- Value ( ) nesnesi içerisinde 2 temel bilgi saklar; dosyanın kendisi ve boyutu.
- Daha sonra bu nesne oluşturulan HashTable a eklenir. Ekleme işlemi birkaç adımda gerçekleşir:
  - Dosyanın boyutu bir hash fonksiyonuna tabi tutulur ve table daki yeri belirlenir.
  - Çakışma olduğu durumda 2 ihtimal vardır: birebir aynı dosya olması ve ya aynı boyutta farklı bir dosya olması.

- Bu ihtimallerin belirlenebilmesi için ilk olarak çakışma olan yerde konumlanmış dosyanın orijinal boyutu ile yeni eklenen dosyanın orijinal boyutu karşılaştırılır.
- Eğer boyutlar aynı ise bu dosyaların birebir aynı olup olmadıkları kontrol edilir.
- Dosyaların birebir aynı olduğu tespit edilirse işlem yapılan konumdaki Value nesnesinin adet özelliği 1 artırılır.
- Eğer boyutları aynı fakat birbirinden farklı dosyalar ise çakışma olduğu takdirde “Linear Probing” yöntemi kullanılarak bir sonraki boş konuma yerleştirilir.
- HashTable’a ekleme işlemleri bittiğinde temel anlamda dosyaların kopyalarından kurtulmuş oluruz.
- Kullanıcı tarafından belirlenen hedef klasöre hash tablosundaki dosyalar yazılır. Bu işlem şu aşamalarla gerçekleştirilir:
  - Tabloda bulunan bütün value nesneleri bir diziye atılır.
  - Sıra ile dosyalar önce URI sonra URL ye dönüştürülür.
  - Daha sonra InputStream ile bir byte dizisine okunur.
  - Byte haline getirilmiş dosyalar ise OutputStream ile hedef klasöre yazılır.

#### Algoritma Şeması:

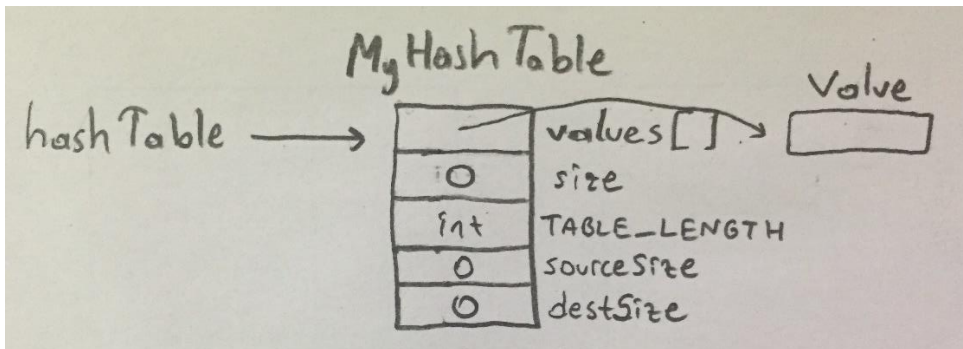
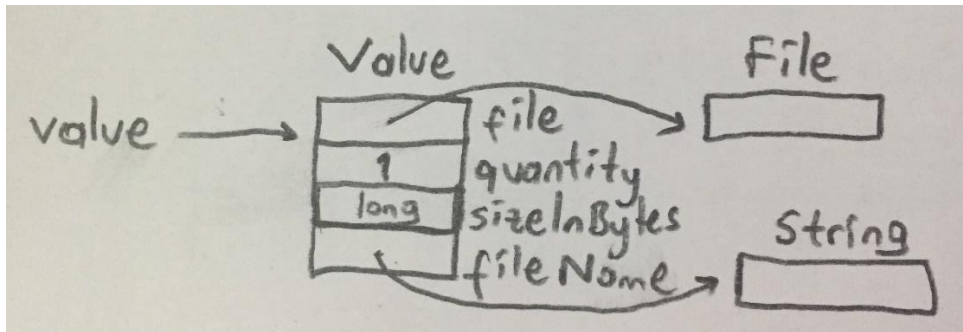




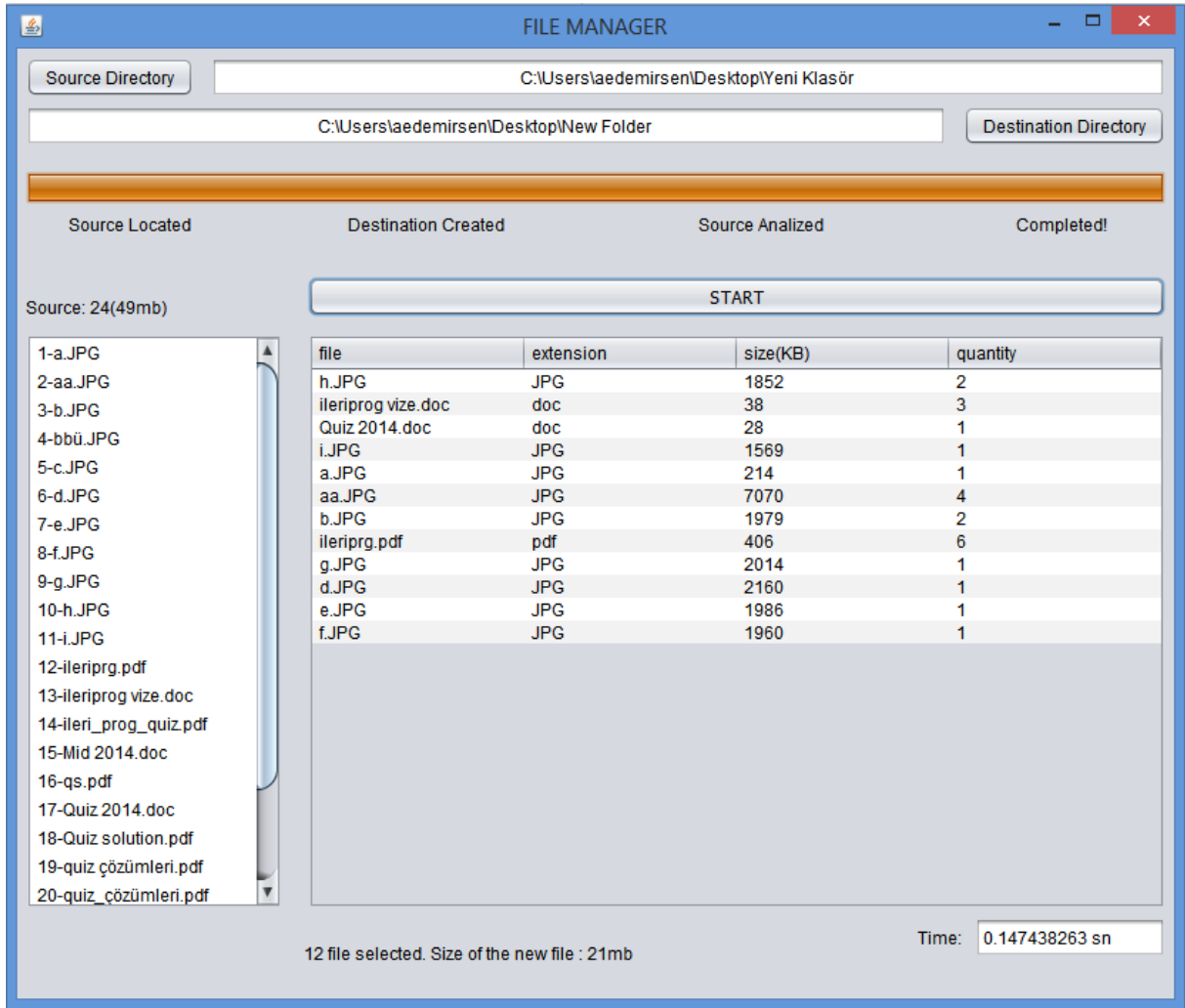
### Proje içerisinde kullanılan Sınıflar:

- JFileManager
- MyHashTable
- FileOperation
- Value

### Nesne Diyagramları:



## Ekran Resmi:



## Kaynak Kod:

<https://github.com/aedemirsen/FileManager>

<http://paste.ubuntu.com/23675008/>