

# PROGRAMLAMA LABORATUVARI 1

## 1. PROJE

Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü  
Programlama Laboratuvarı 1-1.Proje

Hakan AKGÜN  
180202103

Gökalp GÖZÜBOL  
180202084

### 1.Giriş

Bu doküman Programlama Laboratuvarı 1 dersi 1. Projesi için çözümü açıklamaya yönelik oluşturulmuştur. Dokümanda projenin tanımı, çözüme yönelik yapılan araştırmalar, kullanılan yöntemler, proje hazırlanırken kullanılan geliştirme ortamı ve kod bilgisi gibi programın oluşumunu açıklayan başlıklara yer verilmiştir. Doküman sonunda projeyi hazırlarken kullanılan kaynaklar bulunmaktadır

### 2.Temel Bilgiler

Program Visual Studio 2022 üzerinde C++ program dili ile yazılmıştır.

Programa entegreli “Üniversite” klasörü kullanılmıştır.

### 3.Proje Tanımı

Bir veri tabanı kullanmadan üniversitede bulunan bölümleri ve bölümlerde okutulan derslerin içeriklerini gösteren bir uygulama oluşturulması hedeflenmiştir.

#### 3.1.Proje İsterler

- Kullanıcı kelime ve etiket araması yapabilmelidir. Aranılan kelime etiket olabileceği gibi etiket olmayan bir kelime de olabilir.
- Kullanıcı bir etiketin adını değiştirebilir.

- Ana klasörün içerisinde bir output.txt dosyası olmalıdır ve programın tüm çıktıları bu dosyaya yazdırılmalıdır. Çıktılarda tüm etiketler listelenmeli, toplamda kaç adet olduğu bilgisi yer almalıdır.

- Tüm kontrollerin sağlanması için bir menü oluşturulmalıdır. Menüden kullanıcı arama yapabilmeli, etiketi ve o etikete ait txt dosyasının adını güncelleyebilmeli ve dosyaya yazma işlemi yapabilmelidir.

Arayüzler hakkında birkaç görsel;

```
Ana menu
1- Arama
2- Guncelleme
3- Dosyaya Yazma
0- Cikis
3
```

```
Yetim Etiketler:
Ahmet ERDİL
Ahmet SAYAR
Alev MUTLU
Görsel Programlama
Hikmet Hakan GÜREL
Nesneye Yönelik Programlama
ProgramlamaLab
VeritabanıYönetimi
asa
İstenen Etiketler:
Bilgisayar Mühendisliği
Bilgisayar Mühendisliğiö
Burak Ünner
Endüstri Mühendisliği
Kimya Mühendisliği
Matematik_II
NesneyeYönelikProgramlama
Programlama_II
Tarih
Yazılım Mühendisliği
output
Çevre Mühendisliği
Press any key to continue
```

• 2. resimde dosyaları iterate ettikten sonra çıktı için *output.txt* dosyası oluşturuyoruz. Doysa üzerine etiket listesini yazmak için öncelikle başlığı yazıyoruz. Ardından etiket listesindeki her bir etiketin ismini utf8'e dönüştürüp dosyaya yazıyoruz ve yanına da bu etikete kaç kere erişildiğini ekliyoruz. Bu şekilde bütün etiketleri dosyaya yazdıktan sonra boş bir satır daha ekleyip yetim etiketleri eklemeye geçiyoruz. Bunu için benzer şekilde öncelikle bir başlık ekliyoruz. Sonrasında etiket listesindeki her bir etiket için etiket ismindeki boşlukları alttire ile değiştiriyoruz. Sonrasında eğer txt listesinde bu etiket yok ise bu, etiketin yetim etiket olduğunu gösterir, dosyaya bu etiketin ismini yazıyoruz.

### 3.2 Yapılan Örnek Fonksiyonlar ve Bazı Source Code Files

**Dokuwiki.cpp** : Uygulama kodlarının bulunduğu source code file.

**KelimeArama()**: Kelime arama fonksiyonu içerisinde öncelikle dosya sistemi üzerinde tüm dosyaları recursive olarak iterate eden *filesystem* kütüphanesinden *recursive\_directory\_iterator* fonksiyonunu kullanıyoruz. Bu şekilde elde ettiğimiz tüm dosya isimleri üzerinde öncelikle uzantısının *.txt* olup olmadığını kontrol ediyoruz. Sonrasında satır numaralarını kaydetmek için bir değişken tanımladıktan sonra dosyayı açıyoruz. *getline* fonksiyonunu kullanarak dosyayı satır satır okuyoruz. Her bir satır okuma işleminde öncelikle satır sayısını bir arttırıyoruz. Sonrasında okunan satırdaki Türkçe karakterleri kaybetmemek amacıyla okunan veriyi *widestring* çeviriyoruz. Sonrasında bu satırda argüman olarak gelen etiketi regexe dönüştürerek arama yapıyoruz. String fonksiyonlarında olduğu gibi regex işlemlerinin hepsinde *wideregex* kullanıyoruz(*wregex*, *wsregex\_iterator*). Bulunan her eşleşme için dosya adını ve satır numarasını ekrana yazdırıyoruz. Sonrasında dosyayı kapatıyoruz ve *filesystem* üzerindeki sonraki dosyaya geçiyoruz. Bu şekilde fonksiyon tamamlanmış oluyor.

**EtiketArama()**: Etiket arama fonksiyonunda gelen etiket stringinin başına ve sonuna köşeli parantezler

ekleyerek doğrudan *Kelime arama* fonksiyonunu çalıştırıyoruz.

### Örnek Fonksiyon

```
void EtiketArama(wstring etiket)
{
    etiket = L"\\\\" + etiket + L"\\\\";
    KelimeArama(etiket);
}
```

Etiket Listesi	Tekrar Sayısı
Ahmet	1
Alev	1
Matematik	1
Matematik_I	2
Matematik_II	2
Programlama	3
ProgramlamaLab	1
Programlama_I	3
Programlama_II	2
asa	1

Yetim Etiketler:

Ahmet  
Alev  
Matematik  
Programlama  
ProgramlamaLab  
asa

### 4. Kısaca Program Çalışma Yöntemi

Program temel akışında while loop içerisinde sürekli olarak ekrana ana menüyü yazdırıp kullanıcıdan girdileri alarak fonksiyonları çalıştırıyoruz, sonrasında *pause* komutu ile çıktıdan sonra ekranın sabit kalmasını sağladıktan sonra ana menüyü tekrar ekrana yazdırıyoruz ve bu şekilde *Çıkış* komutu gelene kadar devam ediyoruz.

Ana menü içerisinde Arama, Güncelleme ve Çıktı oluşturmak üzere 3 temel fonksiyon bulunuyor. Arama seçimi yapıldığında ise Kelime olarak arama, Etiket olarak arama ve Etiketlerin listelenmesi için 3 alt fonksiyon bulunuyor.

## 5. Akış Şeması









