

Programlama Laboratuvarı I

proje I

Furkan Demirsoy. Kocaeli Üniversitesi Bilgisayar Mühendisliği

ÖNSÖZ

Bu proje Kocaeli Üniversitesi Bilgisayar mühendisliği programlama laboratuvarı I dersi 1. projesidir. Bu raporda bu projeyi nasıl yaptığım, yaparken nelerden faydalandığım ve karşılaştığım hataları nasıl çözdüğüm gibi birçok konuda bilgilendirme yapılcaktır

I. GİRİŞ

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

A. Proje istekleri

İç içe klasörlerden oluşan Üniversite isimli bir klasördeki .txt uzantılı dosyaların içindeki etiketlerin tespit edilmesi ve bu etiketleri kendi isimlerinde bir.txt uzantılı dosya olup olmadığı durumuna göre sınıflandırılması ve kullanıcıdan girilen bir kelimenin etiket mi yoksa bir kelime mi olduğunu tespit edilmesi ve de kullanıcıdan girilerek bir etiketin ve onun varsa dosya adının güncellenebilmesi istenmektedir. Ayrıca sınıflandırılan bütün etiketlerin output.txt isimli bir dosyaya ne kadar geçtiği bilgisiyle beraber kullanıcıya gösterilmesi ve kendi isminde bir txt dosyası olmayan etiketlerin de eğer istenirse txt dosyalarının oluşturulabilmesi istenmektedir.

B.

Projenin C ya da C++ dilleriyle yazılması ayrıca türkçe karakterlere de dikkat edilmesi istenmektedir.

C.

Projede ana menüden tüm etiketlerin ve yetim etiketlerin listelediği output dosyasını görmek. Etiket güncellemesi gibi işlemleri yapabilmek. Ayrıca kelime ve etiket araması yapabilmek için bir arayüz oluşturulması istenmektedir.

D.

Projede listelenmesi ve aranması için etiket formatları bulunmaktadır:

Etiket formatı: Eğer bir kelime etiket ise formatı şu şekillerde olmalıdır: [[kelime]], [[kelime1 kelime2]] Hatalı etiket formatları: [[kelime], [kelime]], [kelime], kelime vb.

E.

Ayrıca projede yetim etiket ve istenen etiket isminde 2 tip etiketin de bulunması istenmektedir

Yetim etiket: Dosyası olmayan etiket. İstenen etiket: Dosya var fakat etiket yok. Dosya ismi veritabanı,onetimi.txt olduğunda istenen etiket : veritabanıyönetimi olmalıdır.

II. YÖNTEM

Bu bölümde projenin kodları açıklanacaktır.

A. Main

Kullanıcıdan etiket ve kelime araması yapabilmesi için aranmak istenen kelime string adında char tipinde bir diziye atanmıştır Ayrıca etiket isimlerinin güncellenmesi için de widget isminde char tipinde bir diziye yeni etiket ismi atanmıştır

ayrıca da outputfile,search,filereading isimli fonksiyonlara bu diziler verilerek istenen işlemlerin yapılması sağlanmıştır

B. Dosyalar arası gezinme

ilk olarak Üniversite isimli klasörün içine girilerek altındaki Bölümler,Dersler vb.Klasörlerin içlerine girilerek içlerindeki .txt uzantılı dosyaların okuması gerektiğinden FILE pointerı ile bir dosya açarak işlem yapmanın tek başına yeterli olmayacağına karar verilmiş bu sebepten dolayı bazı araştırmalar sonucunda DIR isimli dirent.h kütüphanesinde bulunan bir pointer ile klasörlerin içlerine girilmesine karar verilmiştir.

DIR *folder=opendir()

şeklinde dosya açılmış ve struct dirent *insidefolder; şeklinde bir struct pointer tanımlanarak readdir() fonksiyonuyla bir while döngüsü içerisinde üniversite isimli klasörün tüm alt klasör isimleri çağırılmıştır. Daha sonra çağrılan klasör adlarının da içine girilerek içlerindeki .txt uzantılı dosyalara erişmek için DIR *hangouttext; isimli yeni bir pointer tanımlanmış ve tekrar yeni bir struct dirent ile readdir() fonksiyonuyla yeni bir while döngüsü kurularak bu alt klasörlerdeki tüm dosya isimleri okunarak strcmp() fonksiyonuyla sonlarında .txt uzantısı olanlar ayrıştırılmıştır. Bu işlemin ardından finishtring isimli bir string içerisine strcat fonksiyonuyla, C://Users//HUAWEI//Desktop//Üniversite adresinin sonun .txt uzantılı dosya isimleri eklenerek dosyaların açılması, kelime araması, güncellenmesi gibi işlemlerde txt dosyalarının içerilerine girmek için gerekli olan dosya yolu finishstring içerisinde oluşturulmuştur

C. Arama

girilen string dizisi search() fonksiyonuna verilir ve verilen fonksiyon içerisinde dosyalar arası gezinme isimli bölümdeki anlatılanlar aynen uygulanır ve dosya adresleri bir FILE pointer ile opendir() fonksiyonu kullanılarak açılır ve fgets() fonksiyonu ile dosya sonuna kadar satır satır okunur ve etiket formatında verilen "[[" ve "]" arasında bulunanları, pointer aritmetiği yöntemi kullanılarak strstr() fonksiyonu ile tespit edilir daha sonra aynı olarak bulunan kelimelerde satırsayac ismi verilen int tipi değişken her seferinde 1 arttırarak etiketin bulunduğu satır bastırılır ve bu toplam sonucu da en sonda summer ismi verilen değişkenle bütün dosyaların toplamında ne kadar geçtiği bastırılır o an enter-çdname isminde struct ında tutulan dosya adı da bastırılarak etiketin hangi satırlarda olduğu hangi dosyalarda bulunduğu ve toplam geçiş sayısı kullanıcıya gösterilir

D. Güncelleme

String adındaki değişkene girilen etiket ismiyle değiştirilmek istenen etiket adı alınır daha sonra widget isimli char tipi dizi ile bu etiket isminin ne ile değiştirilmek istediği alınır ve alınan diziler filereading isimli fonksiyona gönderilir bu fonksiyonda dosyalar arası gezme bölümünde belirtilen işlemler aynen uygulanır ve dosyalar satır satır okunur okunan satırlar tutucudosya isimli char dizisine aktarılır ve updatestring isimli fonksiyona gönderilir bu fonksiyonda malloc fonksiyonu kullanılarak string ve widget isimlidizilerdeki değerlerin uzunluk farkının oluşturduğu sorunu çözülür daha sonra tekrar strstr fonksiyonu ile string dizisinin geçtiği yerler tespit edilir ve alan açıldıktan sonra widget dizisi yerine kopyalanır

E. Output İçerisine Yazdırmak İçin Tüm Etiketlerin Tespiti

Bütün etiketlerin output dosyasın yazdırılabilmesi için bütün etiketlerin tespit edilmesi gerekmektedir bunun için outputfile isimli bir fonksiyonun içinde dosyalar arası gezinme bölümünde söylenenler aynen uygulanır ve yine strstr fonksiyonu ile etiketler [[,]] durumlarına göre tespit edilir bu tespitten sonra tespit edilen etiketler outputstringi isimli ayrı bir dizi içerisine atılır

F. Outputa Bastırma ve Yetim etiket

Bütün etiketler tespit edilip her seferinde outputstringi isimli stringe atıldıktan sonra bu dizi fonksiyon isimli fonksiyona gönderilir.

ve bu fonksiyonda, her etiketin toplamda (tüm dosyalarda geçen) ne kadar geçtiği değerini döner daha sonra etiketlerin yetim olup olmadığının kontrolü için outputstringi isimli dizi yetimefonksiyon isimli fonksiyona gönderilir bu fonksiyonda her etiketin ismi strcmp() fonksiyonuyla sonu .txt uzantılı olan dosyaların isimleri ile kıyaslanır ve eğer fonksiyondan 26 değeri dönerse onun yetim etiket olmadığı anlaşılır ve daha önceden bulunan geçiş sayısı ile beraber yatim olup olmadığı da outputfile isimli fonksiyonun içinde hem consol ekranına hem de üniversite klasörünün içinde oluşturulan output.txt isimli dosyaya fprintf() fonksiyonu ile bastırılır

G. İstenen Etiket

not:yazdırma durumundaki hata sebebiyle alttre sembolü yerine '*' kullanılmıştır.

Outputstringi isimli etiket isimlerini barındıran dizi istenen*etiket isimli fonksiyona gönderilir ve bu fonksiyonda dosyalar arası gezinme bölümündeki adımlar uygulanarak .txt uzantılı dosyaların isimleri enter-çd*name ile bulunur bulunan enter-çd*name (örnek olarak Programlama*I.txt) den strtok() fonksiyonu ile .txt uzantısını gördükten öncesini bir char pointerına atar atanan bu pointarda bir kontrol yapılır burda eğer dosya isminde (Programlama*I) de '*' işareti var ise bunu strchr() fonksiyonu ile indisini bularak bu işaret yerine ' ' 1 adet boşluk değeri koyar ve strcmp() fonksiyonu ile outputstringindeki etiket adı ile bu ayrı bir stringe atılan yeni dosya adı karşılaştırılır eğer bu isimde bir etiket yoksa o istenen etikettir

III. DENEYSEL SONUÇLAR

A. 1

Programın daha başlarındayken main fonksiyonu içerisinde scanf fonksiyonu ile seçim değerini alırken altındaki if bloklarında gets fonksiyonu kullanıldığını için programı çalıştırınca if bloklarına doğru giriş yapmadı ve doğru değerleri almadı bazı araştırmaların ardından bu sorunu çözmek için fflush(stdin) fonksiyonu gets ve scanf fonksiyonları arasına yerleştirildi ve bu sorun çözüldü.

B. 2

Güncelleme fonksiyonu yazılmaya çalışırken widget dizisi ve string dizisi direkt strcpy ile birbirini üzerine kopyalanmaya çalışıldı ancak bu program çalıştığında string dizisinin uzunluğu widget dizisinin uzunluğundan farklı olduğu için hem çalışma sırasında problemler oluşturdu hem de istenmeyen karakterlerin basılmasına sebep oldu bu sebepten yapılan araştırmalar sonucunda malloc() fonksiyonu kullanılarak program hafızası dinamikleştirildi ve alan değişikliklerinde sorun çıkartmaması sağlandı.

C. 3

outputstring değişkenine etiket atılması sırasında program çalıştırıldığında her ne kadar doğrudan bir sorun olmasada doğrudan dizi içine string geçişinin doğru olmadığı fark edildi bu sorunun sebebinin önceki dizilerden kalan fazladan karakterlerin dizi içerisinde kaldığı fark edildi ve memset() fonksiyonuyla dizi içeriği sıfırlanarak sorun çözüldü

IV. ÖZET

Projede bizden dokuwiki adında veritabanı yönetimi kullanmadan klasörlerin içerisindeki dosyalarda gezinip etiket ve kelimeleri bulan ve değiştirilmesi istenen bir program istende benim yazdığım programda arama,güncelleme,output dosyasına yazma ve diğer birçok özelliği de barındıran bir program oluşturuldu

V. SONUÇ

dirent.h kütüphanesi ve fonksiyonları ,string kütüphane fonksiyonlar,dosya yazma,açma,güncelleme gibi birçok konuda bilgi sahibi olundu ve de dosyaları değiştirmeyi ve istenenleri bulmayı yarayan çok kullanışlı bir program yazıldı.

VI. KAYNAKÇA

Dirent.h in öğrenilmesi
<https://stackoverflow.com/questions/11736060/how-to-read-all-files-in-a-folder-using-c/1173750611737506>

not:adresi açmak isterseniz '*' yerine alt tre koy-
malısınız

bazı hazır string fonksiyonlarının öğrenilmesi
https://www.bilgigunlugum.net/prog/cprog/c*stdkut/string/strc









