



T.C.

ONDOKUZ MAYIS ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BIL204

NESNE YÖNELİMLİ TASARIM VE PROGRAMLAMA

ÖDEV 1

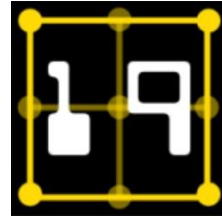
METİN BİÇİMLEME

ÖĞRENCİNİN ADI

Mesut Muhammet Şahin

ÖĞRENCİ NUMARASI

08060318



1.PROBLEMİN TANIMI

Bu ödevde, belirli bir metin dosyası içerisinde yer alan metni, bazı özel karakterler (<, >) kullanılarak normal metinden izole edilmiş etiketlerden hareketle biçimleyerek ekrana yazdıran bir C# programı gerçekleştirilmesi beklenmektedir.

2.YÖNTEM

2.1. Çözüm İçin Önerilen Yaklaşım

Programda öncelikle metinde bulunan özel karakterlerin doğru kullanılıp kullanılmadığı yığıt veri yapısı kullanılarak denetlendi. Hata oluşması durumunda hiçbir işlem yapmadan ekrana uyarı mesajı yazdırılması sağlandı, şayet hata yoksa metin biçimleme işlemi gerçekleştirilerek biçimlenmiş metin ekrana yazdırıldı.

2.2. Gerçekleştirim Detayları

using System : Bu isim uzayının kapsamındaki tüm sınıflardan, arayüzlerden, struct veri tiplerinden yararlanılabileğini ifade eder.

using System.IO : Bu isim uzayı okuma, yazma işlemlerini gerçekleştirmeyi sağlamaktadır.

namespace Uzay2 : Bu isim uzayının içerisinde Stack ve Sınıf1 sınıfları ve bu sınıfların içinde yer alan değişken ve fonksiyonlar bulunmaktadır.

class Stack : String tipinden verileri saklamakta kullanılabilecek bir yığıt veri yapısı gerçeklemek için Stack sınıfı oluşturuldu. Bu sınıfın içerisinde string dizi değişkeni, sayaç ve boyut değişkenleri ve aşağıda belirtilen fonksiyonlar bulunmaktadır. Sınıfın içerisinde bulunan fonksiyonlar başka sınıf içerisinde nesnesi üretilerek çağrılmaktadır. Bu yüzden erişim belirleyicisi olarak public seçilmiştir.

public string[] dizi : Stack sınıfının gerçeklemelerinin yapıldığı dizidir.

public int sayac : Yığıtta bulunan elemanların sayısını tutmaktadır.

public int boyut : Yığıtın kaç eleman aldığını belirtir.

public Stack(int uzunluk) : Aldığı int tipindeki argüman ile, yığıt elemanlarını tutmaya yarar ve string[] tipinden olan dizinin uzunluğunu belirler.

public bool IsEmpty() : Dönüş değeri bool tipinden olup argüman almamaktadır. Eğer yığıtta eleman yoksa true, varsa false döndürür.

public int Push(string eklenen) : String tipinden bir argüman olarak int tipinden bir değer döndürür. Eğer yığıtta boş yer varsa, aldığı elemanı yığıtın tepesine koyar ve elemanın koyulduğu indeksi döndürür. Eğer yığıt dolu ise ve yeni eleman eklenemiyorsa, -1 döndürür.

public string Peek() : Dönüş tipi string olup argüman almamaktadır. Yığtın tepesinde bulunan elemanı döndürür. Yığt boş ise null döndürür.

public string Pop() : Dönüş tipi string olup argüman almamaktadır. Yığtın tepesinde bulunan elemanı döndürür ve bu elemanı yığttan siler. Yığt boşsa null döndürür.

class Sınıf1 : İçerisinde dosya işlemlerini gerçekleyen fonksiyonlar bulunup, görevlerin sorgulanıp, göreve göre işlemlerin yerine getirildiği sınıftır. Bu sınıfın içerisindeki fonksiyonlar birbirlerini çağırılmaktadır ve nesne üretmeye gerek olmadığından, public static erişim belirleyicisi seçilmiştir.

public static string Oku(string DosyaAdresi) : Dosyanın içeriğini string olarak döndürür .

public static void Yaz(string DosyaAdresi, string metin) : Belli bir karakter dizisini dosyaya yazar.

public static bool Tag_mi(string girdi) : Programda okuma yapıldıktan sonra etiketlemelerin doğru olup olmadığına bakılmak için çağrılan fonksiyondur. Eğer burada sonuç false çıkarsa; hatalı mesajı iletilir, true çıkarsa; çözümleme fonksiyonuna geçilir. Bu fonksiyon sonucun hatalı olması durumunda gereksiz yere çözümleme fonksiyonunu çağırılmayı engellemektedir.

public static string Tekrarla(string al) : Kendisine gelen dizgenin iki kere yazdırılmasını sağlayan fonksiyondur.

public static string Koseli(string al) : Kendisine gelen dizgenin köşeli parantezler içinde yazılmasını sağlayan fonksiyondur.

public static string Gizle(string al) : Kendisine gelen dizgenin ekrana bastırılmamasını sağlayan fonksiyondur.

public static string Buyut(string al) : Kendisine gelen dizgenin büyük harfler şeklinde yazılmasını sağlayan fonksiyondur. Burada System.String sınıfının ToUpper fonksiyonunu kullanılmaktadır.

public static string Hangislem(char gorev, string kelime) : Gelen dizgelere hangi işlemin yapılacağı, görevlere göre tespit edilerek işlemlerin yapılacağı fonksiyonlara yönlendirildiği fonksiyondur. Görev olarak etiketlerin içindeki harfi, kelime olarak da işlenecek olan dizgiyi alır.

public static string Cozumleme(string al) : Tag_mi fonksiyonunda sonuç hatalı değil ise buraya gelir. Aldığı dizgeler etiket içermiyorsa normal olarak alır. Eğer etiketleme içeriyorsa açma-kapama olup olmadığına bakar. Açma etiketi geldiğinde, kapama etiketi gelene kadar yığt içinde tutar ve daha sonra yığtta bulunan dizgeye, görevlere göre işlemler yaptırılır. Her işleme uğrayan dizge birleştirilerek tutulur ve bu son dizge döndürülür.

public static void Main() : Programın çalıştırıldığı kısımdır. Dosyadan okuma yapılır. Tag_mi fonksiyonundan olumlu sonuç çıkması durumunda, çözümleme fonksiyonuna geçilir. Aksi takdirde ekrana hata mesajı bastırılır.

3.SONUÇLAR VE YORUMLAR

Öncelikle dosyadan okunan metnin, etiketlemesinin doğru olup olmadığına bakılmaktadır. Doğru olması halinde, açma etiketinden kapama etiketine kadar olan kısım yığıta itilmektedir. Kapama etiketi geldiğinde ise; etiketin türüne göre tanımlanmış olan görevler yapılmaktadır. İşlenmiş dizgeleri tutan ve asıl dizgeyi tutan dizge olmak üzere iki tane dizge tanımlanır. Asıl dizge, işlenmiş dizgeyle birleştirilerek tutulur. İşlenen kısımların tutulduğu dizge, işlemlerin sonrasında sıfırlanır. Etiketlendirme yoksa normal olarak asıl dizgeye ekleme yapılır. Sonuç olarak, toplam dizgenin bulunduğu dizge döndürttürülür.