

GENEL HATIRLATMA:

Aşağıdaki yazılanları baştan sona okumanız ve eksiksiz uymanız zorunludur. Ödevle ilişkin kodlamayı tam ve hatasız yapıp, zamanında iletip, hatasız çalıştırsanız bile, aşağıdaki yazılı kural, talimat, vb. bir tanesine bile uyulmazsa, ödevinizden puan / puanlar kırılabacaktır.

Ödevin Teslim Şekli:

DEÜ Sakai sistemindeki ders sayfasında açılacak olan ödev yükleme (assignment) alanına; tüm dosyalar (tüm kaynak kodları, bağlantılı kütüphane, dizinler, proje dosyaları, vb.), **zip / rar şeklinde sıkıştırılmış tek bir dosya olarak yüklenecektir.**

Ödevinizde **C# veya Java** programlama dillerinden herhangi bir tanesini kullanabilirsiniz. Bu diller dışında bir programlama dili bu ödevde kullanılamaz.

Ödev Konusu, Kapsamı, Gereksinimleri:

Programınızda, ilgili tüm hesaplar, işlemler, değişkenler, vb. sizin ayrıca tanımlayacağınız farklı farklı sınıflar (class) içerisinde yer almalıdır. Bu işlemler ve değişkenler, ana program içerisinde olmayacaktır.

Sınıf, nesne, değişken, metod, vb tüm adlandırmalarda, kodu yazdığınız programlama diline ait evrensel "naming convention" ilkelerine uyulmak zorundadır.

Ekrandan okuma, ekrana yazdırma, vb. I/O işlemleri başta olmak üzere, programınızın gerekli tüm yerlerinde istisna yönetimi (exception handling) olmalıdır.

Kullanıcının, ekrandan 3 ayrı metin girmesi istenecektir. Kullanıcının ekrandan enter tuşuna basıncaya kadar yazdığı her şey, o metin içinde kapsanacaktır. Kullanıcı ilk kez enter tuşuna basınca, 1. metin bitmiş olup 2. metni ekrandan okuyacak, 2. kez enter' a basılınca, 2. metin bitecek, vb...

Ekrandan girilen sözcüklerde büyük / küçük harf ayrımı yapılmayacaktır. Bu nedenle ekrandan girilen tüm sözcüklerin ToLower(), vb. ile küçük harfe çevrilmesi önerilir.

Programınızda, ekrandan girilen her metin sözcüklere bölünecek ve bu her sözcük alınıp string olarak sizin kendi kodlayacağınız ve tanımlayacağınız liste sınıfına eklenecektir. **Sizin bu kendi liste sınıfınızı, List<T> 'ten kalıtımla oluşturacaksınız.**

Kullanıcı ekrandan 3 metni de girmeyi bitirince, ekrana toplamda kaç sözcük girildiği yazılacaktır.

HashSet<T> kullanarak ayrı bir string tipinde hashset oluşturacaksınız. Liste sınıfınızdaki sözcüklerin tekil olarak bu hashset'e eklenmesini sağlayıp kaç adet farklı kelime olduğunu ekrana yazdıracaksınız.

Ayrıca, sizin kendi kodlayacağınız ve tanımlayacağınız generic bir tablo sınıfı olacaktır (örn; MyTable <string,integer>, vb.). Bu tabloda her bir sözcük ve toplam kaç kez geçtiği bilgisi yer alacaktır. Bu bilgiler, yukarıda tanımlanan liste ve hashset' inindeki bilgilerden elde edilerek oluşturulacaktır. Tablonuz, kullandığınız programlama dilindeki (C# ya da Java) hazır generic collections' dan uygun herhangi birisinden (Dictionary, vb) kalıtımla oluşturabilirsiniz.

Bu tablonun oluşturulması sonrasında da, bu tabloyu kullanarak her bir sözcüğü ve bu sözcüğün kaç kere kullanıldığını, ayrıca en çok kullanılan sözcüğü ve bunun adedini ekrana yazdıracaksınız (en sık kullanılan birden fazla sözcük olursa, rasgele bir tanesi seçilip yazdırılması yeterlidir).

Örnek:

> Lütfen 1. metini giriniz:

Merhaba. Bugün ne okusam? Ne yapsam?

> Lütfen 2. metini giriniz:

Ne kadar basit bir kod

> Lütfen 3. metini giriniz:

Bir sorun yok.

Toplam girilen sözcük: 14

Farklı sözcük adedi: 11

Sözcük frekansları:

merhaba: 1

bugün: 1

ne: 3

okusam: 1

yapsam: 1

kadar: 1

basit: 1

bir: 2

sorun: 1

yok: 1

En sık kullanılan sözcük: "ne".

En sık kullanılan sözcüğün frekansı: 3

Ödevde teslim edilecekler listesi:

1-Programın tüm kaynak kodları, bağlantılı kütüphane, dizinler, vb.

2-Kullanılan algoritmalar, vb. ile ilgili kısa bilgiler / notlar (istenirse kaynak kod içine açıklamalar olarak eklenebilir).