1. Kullanıcıdan iki dizgi alan ve bu dizgilerden ilkinin karakterlerinin ilk yarısını, ikinci dizginin karakterlerinin ikinci yarısıyla art arda bağlayarak ekrana yazdıran bir C# programı yazın. Örneğin, ilk dizgi “zonguldak”, ikinci dizgi “bartin” olduğunda ekrana yazdırılacak dizgi “zongutin” olur. Başka bir örnek olarak, ilk dizgi “guzelhisar”, ikinci dizgi “karadeniz” olduğunda ekrana yazdırılacak dizgi “guzeldeniz” olur. “sari” ve “teflon” dizgileri için yazdırılan “salon”dur. Not: Dizgileri kullanıcıdan almanın yolu str bir dizgi başvurusu iken,

str = System.Console.Readline(); kodunu kullanmaktır.

1. dortgen adında, dört farklı bilgi parçası içeren bir sınıf yaratın 🡪 kenar1 (int tipinde), kenar2 (int tipinde) , kenar3 (int tipinde) ve kenar4 (int tipinde). Bilgiler özel (private) erişimli olmalı. Sınıfın, bu dört veri üyesine ilk değerler atayan ve verilen değerlerin doğru olduğunu varsayan (geometrik kuralların kontrollerini yapmayan) üstüne yüklenmiş yapıcıları (constructor) olmalı. Parametresi olmayan yapıcı bütün kenar uzunluklarına 0 atamalı. Bir parametresi olan yapıcı, bütün kenar uzunluklarına o parametre değerini atamalı (eşkenar dörtgen yaratmalı). İki parametresi olan yapıcı, kenar1 ve kenar3’e ilk parametre değerini, kenar2 ve kenar4’e ikinci parametre değerini atamalı (paralelkenar yaratmalı). Üç parametreli yapıcı, 3. parametrenin değerini kullanmayarak ilk iki parametrenin değeriyle iki parametreli yapıcının davrandığı gibi davranmalı. Dört parametreli yapıcı, bu parametrelerin değerlerini sırasıyla kenar1, kenar2, kenar3 ve kenar4’e atamalı (çeşitkenar dörtgen yaratmalı). Ayrıca, sınıf cevreHesapla adında ve dörtgenin çevresini hesaplayıp dönen bir metot içermeli.
2. Değişken sayıda int tipinde parametre alabilen ve parametrelerinin en büyüğü ile en küçüğünün ortalamasını double olarak dönen minmaxort adında bir fonksiyon yazın. Fonksiyon bir nesne fonksiyonu değil, sınıf fonksiyonu olmalıdır.
3. Ders projemizi hatırlayalım. Arttakı (postfix) aritmetik ifadeleri içtakı (infix) ifadelere dönüştürme probleminde, tek harfli işlenenler ve işleçlerden oluşan bir arttakı ifadeyi parantezlerin de işlem önceliğini belirtmek için kullanıldığı bir içtakı ifadeye dönüştürüyorduk. Problemi çözerken yığın (stack) veri yapısı gerektiği gibi kullanılıyordu. ab+cd-\*e/ arttakı aritmetik ifadesi için, başta boş olan yığının içinden geçtiği durumları, aşama aşama, resim çizerek gösterin.