



SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

Nesneye Dayalı Programlama
Ödev II

*Bu raporda ;yapılacak ödevin adım adım yapılışı anlatılmıştır. Öncelikle Console ekranında test edilmiş daha sonra uygun form uygulaması yapılmıştır. Yapılan form uygulamasının ekran çıktısı da son olarak rapora eklenmiştir.

Soru

BenimString isimli bir nesne bazlı sınıf oluşturmanız istenmektedir. Bu sınıfta yer alacak üye metodlar.

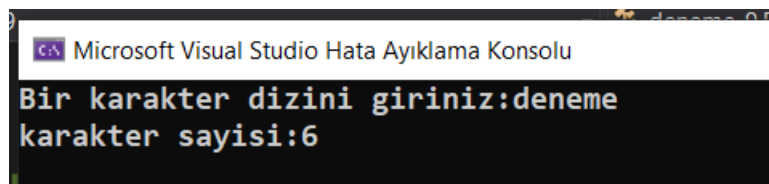
ElemanSayısı : nensedeki eleman sayısını verecek(string sınıfı size metodu gibi)

Fonksiyon:

```
public static int elemanSayisi(string veri) {
    int size=0;
    foreach (char c in veri)
    {
        //if (c == '\n') { break; }
        size++;
    }
    return size;
}
```

Main Yapısı:

```
string kelime;
int karakterSayisi = 0;
Console.Write("Bir karakter dizini giriniz:");
kelime = Console.ReadLine();
karakterSayisi = elemanSayisi(kelime);
Console.WriteLine("karakter sayisi:{1}", kelime, karakterSayisi);
```



Birleştir : İki string nesne değeri birleştirilecek(string sınıfı concat metodu gibi)

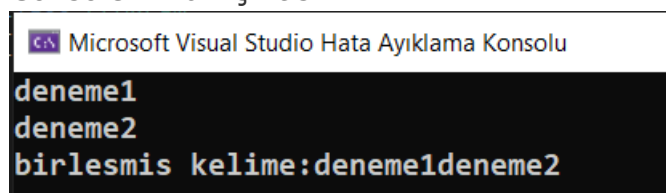
Fonksiyon:

```
public static string birlestir(string kelime1, string kelime2) {
    string birlesmis;
    birlesmis = kelime1 + kelime2;
    //Console.WriteLine(birlesmis);
    return birlesmis;
}
```

Main Yapısı:

```
string k1,k2,bkelime="";
k1 = Console.ReadLine();
k2 = Console.ReadLine();
bkelime = birlestir(k1, k2);
Console.WriteLine("birlesmis kelime:{0}", bkelime);
```

Console Ekran Çıktısı:



ArayaGir : verilen string sabit veya nesne değeri, kaynak nesnenin değerlerinin arasına yerleştirilecek yerleştirme parametre olarak verilen bir indisten itibaren olacaktır.(string sınıfı insert metodu gibi)

Fonksiyon:

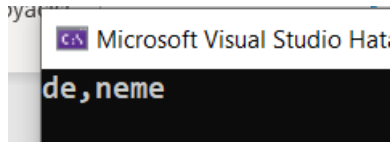
```
public static string arayaGir(string kelime1,int s, string b) {
    string word = "";
    for (int i = 0; i < elemensayisi(kelime1); i++) {
        word = word + kelime1[i];
        if (i == s) {
            word = word + b;
        }
    }
    return word;
}
```

Main Yapısı:

```
string kelime = "deneme";
string gelen;

gelen= arayaGir(kelime, 1, ",");
Console.WriteLine(gelen);
```

Ekran Çıktısı:



DeğerAl : kaynak nesne değerini belirtilen indisten başlayarak belirtilen karakter kadar geri döndüren bir fonksiyon ((string sınıfı insert metodu gibi)

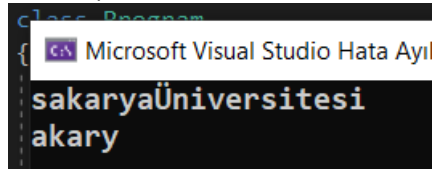
Fonksiyon:

```
static string degerAl(string kelime1,int s,int m) {
    string word = "";
    for (int i = s; i <= m; i++) {
        word = word + kelime1[i];
    }
    return word;
}
```

Main Yapısı:

```
string kelime = Console.ReadLine();
string gelen=degerAl(kelime, 1,5);
Console.WriteLine(gelen);
```

Ekran Çıktısı:



DiziyeAyır : Kaynak nesne değerini verilen karaktere göre diziye ayıracak((string sınıfı split metodu gibi)

Fonksiyon:

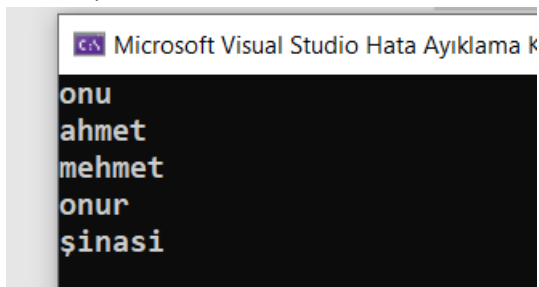
```
public static string[] diziyeAyir(string kelime1,char c) {
    int kelimesayisi = 0,a=0;
    for (int k = 0; k < kelime1.Length; k++)
        if (kelime1[k] == c)
            kelimesayisi++;
    string[] word=new string[10];
    string kelime = "";
    for (int i = 0; i < kelimesayisi; i++) {

        if (kelime1[i] != c) {
            kelime = kelime + kelime1[i];
        }
        else {
            word[a] = kelime;
            kelime = "";
            a++;
        }
    }
    word[a] = kelime;
    return word;
}
```

Main Yapısı:

```
string k = "onu,ahmet,mehmet,onur,şinasi";
string []dizi=diziyeAyir(k, ',');
for (int i = 0; i < dizi.Length; i++) {
    Console.WriteLine(dizi[i]);
}
```

Ekran Çıktısı:



CharDiziyeDonustur : kaynak nesne değerini char diziye dönüştürecek.)

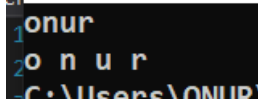
Fonksiyon:

```
public static char[] charDizi(string kelime1) {
    char[] word = new char[kelime1.Length];
    for (int i = 0; i < kelime1.Length; i++)
    {
        word[i] = kelime1[i];
    }
    return word;
}
```

Main yapısı:

```
string kelime = "onur";
char [] dizi=charDizi(kelime);
Console.WriteLine(kelime);
foreach (var item in dizi)
{
    Console.Write(item + " ");
}
```

Ekran Çıktısı:



DegerIndis : kaynak nesne içerisinde aranan nesnenin indisini döndürür.

```
static string degerAl(string kelime1, int s, int m)
{
    string word = "";
    for (int i = s; i < m+s; i++)
    {
        word = word + kelime1[i];
    }
    return word;
}

public static int degerIndis(string kelime1, string aranan) {

    int indis = -1;

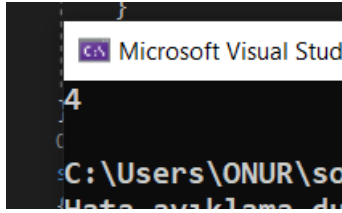
    for (int i = 0; i <= elemensayisi(kelime1)] - (elemensayisi(aranan));)
i++)
    {
        string bulunan = degerAl(kelime1, i, elemensayisi(aranan));
        //string bulunan = kelime1.Substring(i, aranan.Length); //aranan
kelime boyutunda teker teker arama yapar
        if (bulunan.ToUpper() == aranan.ToUpper()) //aralınılan kelime ile
bulunan eşit ise ekrana indis yazar
        {
            indis = i;
            break;
        }
    }

    return indis;
}
```

Main yapısı:

```
int indis;
string cumle = "deneme";
string aranan = "me";
indis = degerIndis(cumle, aranan);
if (indis == -1)
    Console.WriteLine("Aranan karakter cumlede yok");
else
    Console.WriteLine(indis);
```

Ekran Çıktısı:



SıralaAZ : kaynak nesnenin değerlerini a dan z ye sıralı döndürür

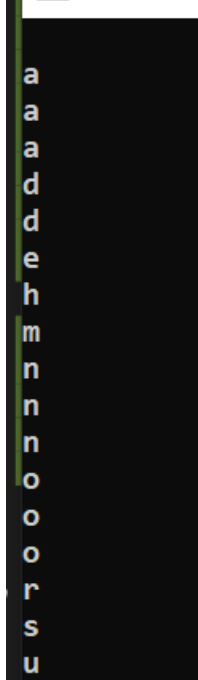
Fonksiyon

```
public static char[] siralaAz(char [] dizi) {
    char temp;
    for (int i = 0; i <= dizi.Length - 1; i++)
    {
        for (int j = i + 1; j < dizi.Length; j++)
        {
            if (dizi[i] > dizi[j])
            {
                temp = dizi[i];
                dizi[i] = dizi[j];
                dizi[j] = temp;
            }
        }
    }
    return dizi;
}

public static char[] charDizi(string kelime1)
{
    char[] word = new char[kelime1.Length];
    for (int i = 0; i < kelime1.Length; i++)
    {
        word[i] = kelime1[i];
    }
    return word;
}

static void Main(string[] args)
{
    string kelime = "onurahmennoosda ad";
    char[] dizi = charDizi(kelime);
    char[] siraliDizi = siralaAz(dizi);
    foreach (var item in siraliDizi)
        Console.WriteLine(item);
}
}
```

Ekran Çıktısı

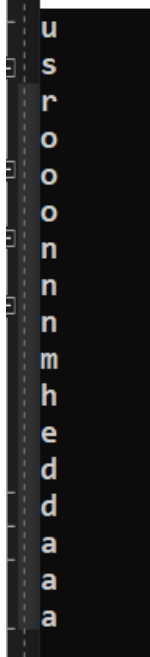


SıralaZA : kaynak nesnenin değerini z den a ya sırala.

Fonksiyon

```
public static char[] siralaZa(char[] dizi)
{
    char temp;
    for (int i = 0; i <= dizi.Length - 1; i++)
    {
        for (int j = i + 1; j < dizi.Length; j++)
        {
            if (dizi[i] < dizi[j])
            {
                temp = dizi[i];
                dizi[i] = dizi[j];
                dizi[j] = temp;
            }
        }
    }
    return dizi;
}
```

Ekran Çıktısı:

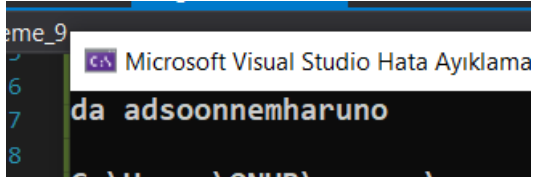
A screenshot of a terminal window with a black background and white text. The text is the string "u s r o o n n n m h e d d a a a" displayed vertically, one character per line, representing the reversed input string.

TersCevir : kaynak nesnenin değerini ters çevirir

Fonksiyon

```
public static string TersCevir(string gelen) {  
    string tersString="";  
    for (int i = elemanSayisi(gelen) - 1; i >= 0; i--)  
    {  
        tersString= tersString+ gelen[i];  
    }  
    return tersString;  
}
```

Ekran Çıktısı:

A screenshot of a Visual Studio console window. The title bar shows "me_9" and "Microsoft Visual Studio Hata Ayıklama". The console output shows the string "da adsoonnemharuno" in white text on a black background. Line numbers 6, 7, and 8 are visible on the left margin.

Bu fonksiyonların kullanıldığı test sınıfı birer adet uygulama yeterlidir.

Uygulamanın Ekran Çıktısı

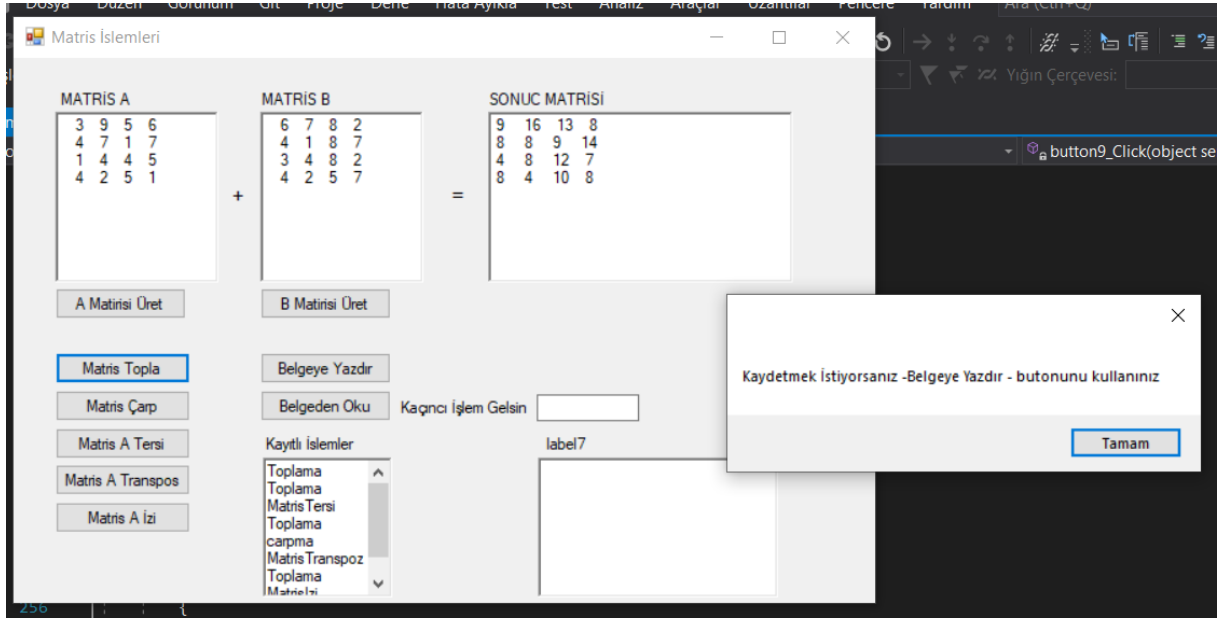
The screenshot shows a Windows application window titled "Form1". The interface is organized into a grid of controls. On the left, there is a vertical list of function names in parentheses: (elemansayisi), (birlestir), (arayaGir), (degerAl), (diziyeAyir), (charDizi), (degerIndis), (siralaz), (siralaz), (tersCevir). Each function name is followed by a "Calistir" button. To the right of these buttons are input fields and labels for each function. For example, for (elemansayisi), there is an empty text box followed by the label "KarakterSayisi:". For (birlestir), there are two "Deger gir" text boxes followed by "Birlestirilmis Deger:". For (arayaGir), there are "Kaynak Nesne", "Ara Deger", and "Indis" text boxes followed by "Son Durum:". For (degerAl), there are "Kaynak Nesne", "Indis", and "Karakter Sayisi" text boxes followed by "Alinana Parça:". For (diziyeAyir), there are "Kaynak Nesne", "Karakter Gir", and "Yeni Dizi:" text boxes. For (charDizi), there are "Kaynak Nesne", "Char Dizisi:", and "A-Z:" text boxes. For (degerIndis), there are "Kaynak Nesne", "Aranan", and "Indis:" text boxes. For (siralaz), there are "Kaynak Nesne", "A-Z:", and "Z-A:" text boxes. For (tersCevir), there are "Kaynak Nesne" and "Ters String:" text boxes.

Çalışan Şekli

This screenshot shows the same application window as the previous one, but with the "Calistir" button for the first function, (elemansayisi), highlighted with a blue border. The rest of the interface remains the same, with the same grid of function names, buttons, and input fields.

Soru 2

Ne yaptık



Matris Üretme

Fonksiyon:

```
public static int[,] matrisUret()
{
    Random rnd = new Random();
    string m = "";
    int[,] matris = new int[4, 4];
    int sayi;
    for (int i = 0; i < 4; i++)
    {
        for (int k = 0; k < 4; k++)
        {
            sayi = rnd.Next(1,10);
            matris[i, k] = sayi;
            m = m + string.Format("{0,5}", sayi.ToString());
        }
    }

    return matris;
}
```

Matris yazdırma metodu

Fonksiyon

```
private void button8_Click(object sender, EventArgs e)
{
    string m = "";
    FileStream fs = new FileStream("deneme.txt", FileMode.Append,
    FileAccess.Write, FileShare.Write);
    StreamWriter sw = new StreamWriter(fs);
    for (int i = 0; i < 4; i++)
    {
        for (int k = 0; k < 4; k++)
        {
            if (islem == "MatrisTersi")
            {
                m = m + string.Format("{0:0.0}", matrisC[i, k]) + " ";
            }
            else
            {
                m = m + string.Format("{0,5}", matris[i, k]) + " ";
            }
        }
        sw.WriteLine(m);
        m = "";
    }
}
```

```

        {
            m = m + string.Format("{0}", matrisC[i, k]) + " ";
        }

    }
}
//islem = "toplam";
m = islem + " "+m;
MessageBox.Show(islem+" islemi kaydedildi");
sw.WriteLine(m);
sw.Close();
richTextBox2.Text = islemOku();
}

```

Matris toplama metodu

Fonksiyon:

```

static void matrisToplama()
{
    int satir = 4;
    int sutun = 4;
    // int[,] matrisC = new int[satir, sutun];
    for (int i = 0; i < 4; i++)
    {
        for (int j = 0; j < 4; j++)
        {
            matrisC[i, j] = matrisA[i, j] + matrisB[i, j];
        }
    }
}

```

Matris çarpma metodu

Fonksiyon:

```

static void matrisCarpma()
{
    int satir = 4;
    int sutun = 4;
    // int[,] matrisC = new int[satir, sutun];
    for (int i = 0; i < 4; i++)
    {
        for (int j = 0; j < 4; j++)
        {
            for (int k = 0; k < 4; k++)
            {
                matrisC[i, j] += matrisA[i, k] * matrisB[k, j];
            }
        }
    }
}

```

Matris okuma metodu

Fonksiyon:

```

static string[] dosyaOku(int islem)
{
    string[] dizi = new string[17];
    string m = "";
    string textFile = "deneme.txt";
    using (StreamReader file = new StreamReader(textFile))
    {
        int counter = 1;
        string ln;

        while ((ln = file.ReadLine()) != null)
        {

```

```

        dizi = ln.Split(' ');
        if (islem == counter) break;
        counter++;
    }

    file.Close();

}

return dizi;

}

```

Matris tersi alma metodu:

Komutları

```

label4.Text = "";
bRichTxt.Text = "";
for (int i = 0; i < 4; i++)
{
    for (int j = 0; j < 4; j++)
    {
        if (i == j)
            matrisC[i, j] = 1;
        else
            matrisC[i, j] = 0;
    }
}

int d, k;
for (int i = 0; i < 4; i++)
{
    d = matrisA[i, i];
    for (int j = 0; j < 4; j++)
    {
        matrisA[i, j] = matrisA[i, j] / d;
        matrisC[i, j] = matrisC[i, j] / d;
    }
    for (int x = 0; x < 4; x++)
    {
        if (x != i)
        {
            k = matrisA[x, i];
            for (int j = 0; j < 4; j++)
            {
                matrisA[x, j] = matrisA[x, j] - (matrisA[i, j] * k);
                matrisC[x, j] = matrisC[x, j] - (matrisC[i, j] * k);
            }
        }
    }
}

```

Matris izi bulma metodu

Matris transpoze metodu

Fonksiyon:

```

static void matrisTanspoz()
{
    for (int j = 0; j < 4; j++)
    {
        for (int i = 0; i < 4; i++)
        {
            matrisC[j,i]=matrisA[i,j];
        }
    }
}

```

}

Uygulamanın Çözümü ve Açıklaması

Giriş kısmında da bahsedildiği gibi öncelikle console uygulaması fonksiyon fonksiyon yapılmıştır .Daha sonrsında ise yapılan form uygulamasına bazı değişiklikler yapılarak uygulanmıştır.

Uygulamanın Çalışma Adımları:

- 1- Matris üret butonlarıyla 2 adet matris üretilir
- 2-İşlem butonları ile işlemler yapılır ve sonuç matrisi elde edilir.
- 3-Kullanıcı isterse bunları belgeye "belgeye yazdır " butonu ile kaydedebilir.
- 4-Kullanıcı isterse kayıtlı olan işlem ve sonuçlarını istediği işlemin numrasını girererek ekranda gösterebilir.

Uygulama çalıştığında elde edilen ekran çıktılarını görelim:

Ödevin Bana Kazandıkları:

Bu ödev ile string sınıfı fonksiyonlarını yaparak arka planda String sınıfının az da olsa işlevlerini kavramış oldum. Tabi yapılan örnekte yine string sınıfı tip olarak kullanılmıştır. Yapılan işlemler fonksiyonlar düzeyindeydi. Ama bu bile programlama bilgime katkıda bulunmuştur.

2.soruda ise dosya okuma ve yazma işlemleri ile birlikte matris ile ilgili işlemleri , static yapısını ve form uygulamasının az da olsa inceliklerini öğrenmiş oldum.