

C#

Ara sınav	- 30%
Ödev (Haftalık)	- 15%
Katılım	- 10%
Final	- 45%

e-posta : akcayol@gazi.edu.tr
web adresi : <http://w3.gazi.edu.tr/~akcayol>

Yrd.Doç.Dr. M. Ali Akcayol G. Ü. Bilgisayar Mühendisliği Bölümü

C#

- giriş
- program yapısı
- değişkenler
- metotlar
- class
- kontrol deyimleri
- döngü deyimleri
- diziler
- struct
- try catch
- files
- GUI

Yrd.Doç.Dr. M. Ali Akcayol G. Ü. Bilgisayar Mühendisliği Bölümü

C#

// Açıklamalar (opsiyonel)
using namespace-adi;

class class-adi

{

static void Main()

{

Değişken tanımları,
Aritmetik işlemler,
Değişken atamaları,
Metot çağırma,
Nesne oluşturma, ...

}

}

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

using System;

namespace Program1

{

class Merhaba

{

public static int Toplam(int a, int b)

{

return a + b;

}

public static void Main(string[] args)

{

Console.WriteLine("Merhaba");
Console.WriteLine(Convert.ToString(Toplam(5, 6)));
Console.Read();

}

}

}

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
// Bu program bir sayıyı dışarıdan alır
// Karesini ekrana yazar.
using System;

namespace Program1
{
    class KareAl
    {
        // Main programın çalışmaya başlama noktasıdır.
        static void Main()
        {
            // Giriş -- kullanıcıdan bir sayı al
            Console.Write("Bir sayı gir: ");
            double number = double.Parse(Console.ReadLine());

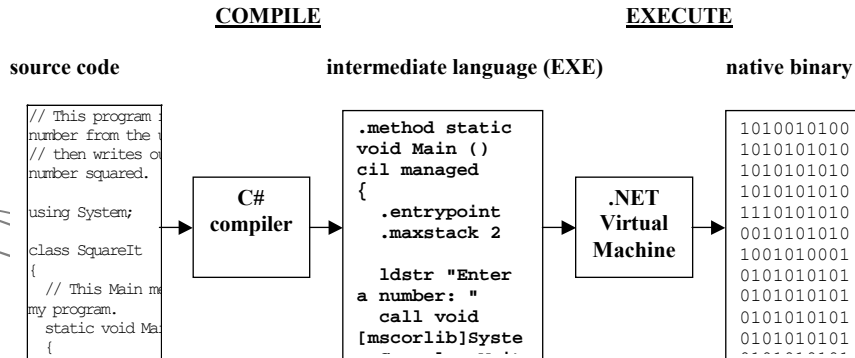
            // İşlem -- giriş değerini kullanarak hesap yap
            double result = number * number;

            // Çıkış -- ekrana sonucu yaz
            Console.WriteLine("sayı = {0} \nkare = {1}", number, result);
        }
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Bir programın derlenmesi ve çalıştırılması



G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Değişken tanımı

Değişken tipleri

sbyte, byte, short, ushort, int, uint, long, ulong, float, double, decimal, char, bool, string

Tanımlama kuralları

İlk karakter harf, alt çizgi veya @ sembolü ile başlamalıdır.

Devam eden karakterler harf, alt çizgi veya rakam olabilir.

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Tip değerleri

Type	Size (bits)	Değer Aralığı
Integer		
sbyte	8	-128 to 127
byte	8	0 to 255
short	16	-32768 to 32767
ushort	16	0 to 65535
int	32	-2,147,483,648 to -2,147,483,647
uint	32	0 to 4,294,967,295
long	64	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
ulong	64	0 to 18,446,744,073,709,551,615
Real		
float	32	1.5×10^{-45} to 3.4×10^{38}
double	64	5.0×10^{-324} to 1.7×10^{308}
decimal	128	1.0×10^{-28} to 7.9×10^{28}
Diğer		
char	16	Karakter kümesi
bool	1	true veya false

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Anahtar kelimeler

abstract	delegate	internal	readonly	try
as	do	is	ref	typeof
base	double	lock	return	uint
bool	else	long	sbyte	ulong
break	enum	namespace	sealed	unchecked
byte	event	new	short	unsafe
case	explicit	null	sizeof	ushort
catch	extern	object	stackalloc	using
char	false	operator	static	virtual
checked	finally	out	string	void
class	float	override	struct	volatile
const	for	params	switch	while
continue	if	private	this	
decimal	implicit	protected	throw	
default	int	public	true	

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

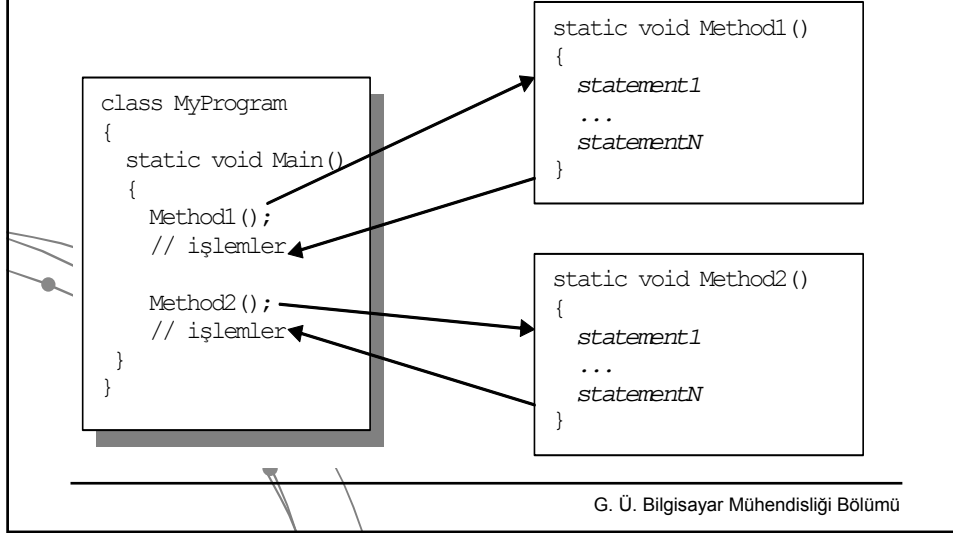
Literal karakterler

Literal karakterleri	İşlem
\'	Tek tırnak
\"	Çift tırnak
\\	Backslash
\0	Null
\a	Alarm (beep!)
\b	Backspace
\f	Form feed
\n	Yeni satır
\t	Yatay tab
\v	Dikey tab

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Metot çağırımı



C#

Metot çağırımı

1- Metot isimleri tip tanımlayıcı olarak alınır

2- Değer döndürme

void (Değer döndürmez)

type (Tanımlanan türde değer döndürür)

3- void olmayan tüm metotlar kapatılmadan önce mutlaka return deyimini kullanmalıdır.

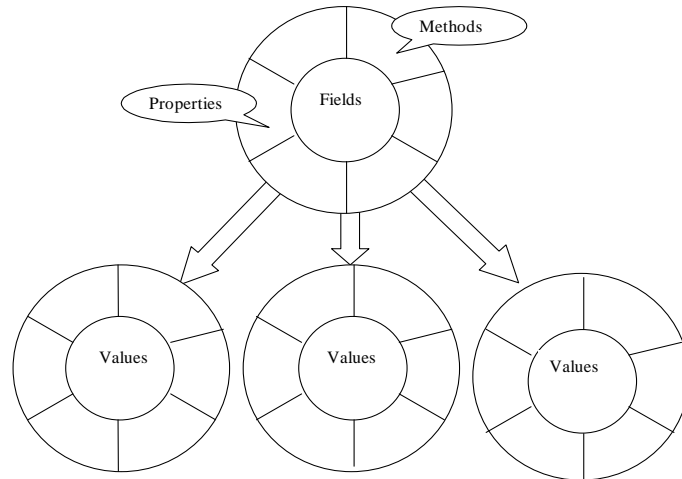
Örnek:

```
public void uyarı();
public int toplam(int a, int b);
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Class



G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
using System;
namespace Program1
{
    public class BankaHesap
    {
        // Fields - Her BankaHesap nesnesi kendi balance,
        // ID, ve PIN değerlerine sahiptir
        public double balance;
        private String id;
        private String pin;

        // Constructor
        // Initializes the fields
        public BankaHesap(String init_id, double init_balance)
        {
            id = init_id;
            balance = init_balance;
            pin = null;
        }
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
// Properties
// Bir hesabın balance değerine erişim (get = okuma, set = yazma)
// Burada sadece okuma yapılabilmektedir
public double Balance
{
    get { return Math.Round(balance, 2); }
}

// Bir hesabın ID değerine erişim
// Burada sadece okuma yapılabilmektedir
public String ID
{
    get { return id; }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
// Bir hesabın PIN değerine erişim
// Burada hem okuma hemde yazma yapılabilmektedir
public String PIN
{
    get { return pin; } // değer okuma
    set { pin = value; } // değer değiştirme
}

// Methods
// Toplama değer ekleme
public void Deposit(double amount)
{
    balance = balance + amount;
}

// Toplamdan değer düşme
public void Withdraw(double amount)
{
    balance = balance - amount;
}

} // BankaHesap class sonu
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
class Islem
{
    static double miktar;
    static string smiktar;

    // Main programın çalışmaya başlama noktasıdır.
    static void Main()
    {
        BankaHesap ba = new BankaHesap("hesap", 1000);
        ba.PIN = "1234567890";

        // Giriş -- kullanıcıdan bir değer al
        Console.Write("miktar gir: ");
        smiktar = Console.ReadLine();
        miktar = Convert.ToDouble(smiktar);
        ba.Deposit(miktar);
        Console.WriteLine("Hesaptaki para = {0}", ba.balance);
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
// Giriş -- kullanıcıdan bir değer al
Console.Write("miktar gir: ");
smiktar= Console.ReadLine();
miktar = Convert.ToDouble(smiktar);
ba.Withdraw(miktar);
Console.WriteLine("Hesaptaki para = {0}", ba.balance);
Console.WriteLine("PIN = {0}", ba.PIN);
Console.ReadLine();
} // Main sonu
// Islem class sonu
} // Namespace sonu
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

if deyimi

```
if (Boolean-expression)
    statement;
if (Boolean-expression)
{
    statement(s);
}
```

Örnek:

```
if (isi > 30)
    Console.WriteLine("Hava sıcak");
if (isi < 10)
{
    Console.WriteLine("Hava soğuk");
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

if - else deyimi

```
if (Boolean-expression)
    statement(s);      // true
else
    statement(s);      // false
```

Örnek:

```
if (satis <= 20000)
    Console.WriteLine("Prim yok");
else
    Console.WriteLine("Prim var");
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

switch deyimi

```
switch (expression)
{
    case constant-expression :
        statement ;
        jump-statement ;
    case constant-expression :
        statement ;
        jump-statement ;
    ...
    default:           // opsiyonel
        statement
        jump-statement ;
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

switch deyimi

Örnek:

```
Console.WriteLine("Seçiminiz 1, 2, or 3: ");
int secim = int.Parse(Console.ReadLine());
switch (secim)
{
    case 1:
        Console.WriteLine("1. seçenek seçildi");
        break;
    case 2:
        Console.WriteLine("2. seçenek seçildi ");
        break;
    case 3:
        Console.WriteLine("3. seçenek seçildi ");
        break;
    default:
        Console.WriteLine(" 1, 2 veya 3. seçeneği seçiniz...");
        break;
} // switch sonu
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

while deyimi

```
while (Boolean expression)
    statement to execute;
while (Boolean expression)
{
    block of statements to execute;
}
```

Örnekler:

```
int i = 0;
int max = 10;
while (i < max)
{
    Console.WriteLine(i);
    i++;
}
```

```
string input = null;
while (input != "OK")
    input = Console.ReadLine(); // tek deyim
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

do/while deyimi

```
do
{
    statement(s) to execute;
}
while(Boolean expression);
```

Örnekler:

```
int counter = 1;
int n = 4;
Console.WriteLine("Döngüden önce...");
do
{
    Console.WriteLine("Döngü #{0}", counter);
    counter++;
}
while (counter <= n);
Console.WriteLine("...Döngüden sonra");
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

for deyimi

```
for (initialization-statement; Boolean-test; update-statement(s))  
    statement to execute;  
for (initialization-statement; Boolean-test; update-statement(s))  
{  
    block of statement(s) to execute;  
}
```

Örnekler:

```
for (int i = 0; i < 10; i++)  
{  
    Console.WriteLine(i);  
}
```

```
for (int pow = 1, result = 2; pow < 10; pow++, result *= 2)  
{  
    Console.WriteLine("2 to the {0} = {1}", pow, result);  
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

foreach deyimi

```
foreach (variable-type variable-name in collection-name)  
    statement to execute;  
foreach (variable-type variable-name in collection-name)  
{  
    statement(s) to execute;  
}
```

Örnekler:

```
string message = "Merhaba!";  
foreach (char ch in message)  
{  
    Console.WriteLine(ch);  
}
```

Ekran çıktısı

M
e
r
h
a
b
a
!

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

break ve continue deyimleri

Örnekler:

```
string message = "Merhaba!";  
foreach (char ch in message)  
{  
    if (ch == 'h') continue;  
    Console.WriteLine(ch);  
    if (ch == 'b') break;  
}
```

Ekran çıktısı:

M
e
r
a
b

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

array

```
type[] array-name; // Array tanımı  
type[] array-name = new type [capacity]; // Tanım ve yapılandırma
```

Array tanımı örnekleri:

```
int[] arrayOfInts; // Array tanımlandı, yapılandırılmadı
```

```
BankAccount[] accountHolder;
```

Array yapılandırma örnekleri:

```
int[] test = new int[100]; // 100 tane değeri 0 olan integer sayı atar
```

```
double[] number = new double[10000]; // 10000 değeri 0.0 olan sayı
```

```
String[] name = new String[500];
```

```
BankAccount[] acct = new BankAccount[1000];
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

array

Örnekler:

```
test[0] = 91; // 91 sayısını 0. elemana atar (ilk eleman)
number[9999] = 123.45;
name[1] = "Aylin"; // String değeri ikinci elemana atar
acct[23] = new BankAccount("Selim", 100.00);
```

```
Console.WriteLine("{0} | {1} | {2} | {3}",
    test[0], name[1], number[9999], acct[23]);
```

Ekran çıktısı

```
91 | Aylin | 123.45 | Selim $100.00
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

array (iki boyutlu – 2D)

```
typeOrClass [ ][ ] identifier = new typeOrClass[satir][sutun];
```

2D - array tanımı örnekleri:

```
double[ ][ ] matris = new double[4][8];
String[ ][ ] ad = new String[5][10];
```

Değer atama ve okuma:

```
String[ ][ ] ad = new String[5][10];
ad[0][0] = "sol ust";
ad[4][9] = "sag alt";
System.out.println("Satır 0, sütun 0: " + ad[0][0]);
System.out.println("Satır 4, sütun 9: " + ad[4][9]);
System.out.println(ad[0][0].toUpperCase() + " " + ad[4][9].toUpperCase());
```

Ekran çıktısı:

```
Satır 0, sütun 0: sol ust
Satır 4, sütun 9: sag alt
SOL UST SAG ALT
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Struct deyimi

```
struct odemeTuru
{
    int taksitSayisi;
    double taksitMiktari;
}
Örnek:
using System;
struct Point
{
    // x ve y static olamaz
    public int x, y;
    public Point(int x, int y)
    {
        this.x = x;
        this.y = y;
    }
}
public class Tester
{
    public static void Main()
    {
        Point a = new Point(10, 10);
        Console.WriteLine(a.x);
        Console.ReadLine();
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Örnek:

```
using System;
struct KeyValuePair
{
    public string key;
    public string deger;
    public KeyValuePair(string key, string deger)
    {
        if (key == null || deger == null) throw new ArgumentException();
        this.key = key;
        this.deger = deger;
    }
}
public class Tester
{
    public static void Main()
    {
        KeyValuePair A = new KeyValuePair("AA11", "12345");
        Console.WriteLine(A.key, A.deger);
        Console.ReadLine();
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
try
{
    çalışacak kod
}
catch (exception-type)
{
    umulan hatanın olması durumunda işlenecek kod
}
```

Örnek:

```
class HandleException
{
    static void Main()
    {
        double number;
        Console.Write("Bir sayı gir: ");
        String numString = Console.ReadLine( );
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
try
{
    number = double.Parse(numString);
    Console.WriteLine("Sayı hatasız çevrildi.");
}
catch (FormatException)
{
    Console.WriteLine("Hata: {0} . Sayı hatalı !", numString);
    Console.WriteLine("Sayıya -1.0 değeri atandı.");
    number = -1.0;
}
Console.WriteLine("{0} sayısı {1} olarak kaydedildi", numString, number);
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Örnek:

```
using System;
class MultipleExceptions
{
    static void Main()
    {
        int[] numberArray = {1, 2, 3, 0, 4, 5};
        String line = null;

        try
        {
            Console.WriteLine("1. Sıra değerini gir : ");
            line = Console.ReadLine();
            int index1 = int.Parse(line);
            Console.WriteLine("2. Sıra değerini gir : ");
            line = Console.ReadLine();
            int index2 = int.Parse(line);
            int result = numberArray[index1] / numberArray[index2];
            Console.WriteLine("Bölüm sonucu: {0}", result);
        }
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
        catch (FormatException fe)
        {
            Console.WriteLine("Hata: Geçersiz sayı girildi: {0}", line);
            Console.WriteLine(fe.Message);
        }
        catch (IndexOutOfRangeException ioore)
        {
            Console.WriteLine("Hata: Geçersiz sıra no.");
            Console.WriteLine(ioore.Message);
        }
        catch (DivideByZeroException dbze)
        {
            Console.WriteLine("Hata: Sıfıra bölüm!");
            Console.WriteLine(dbze.Message);
        }
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

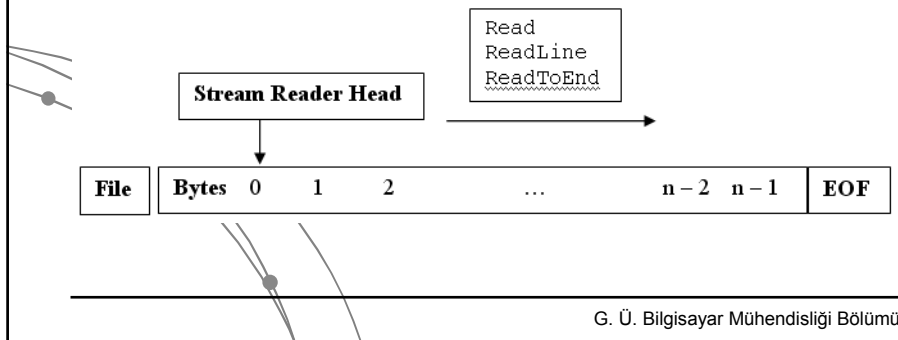
Files

```
StreamReader variable-name = new StreamReader( file-name );
```

Örnekler:

```
StreamReader reader = new StreamReader("input.txt");
```

```
StreamReader reader2 = new StreamReader("C:\\data\\taxes.dat");
```



G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Örnek:

```
using System;
using System.IO;
class ReadEntireFile
{
    static void Main()
    {
        StreamReader reader = new StreamReader("c:\\input.txt"); // Stream aç

        int lineNum = 1; // Satır nolarını saklamak için
        while (reader.Peek() != -1) // Herbir satırı oku
        {
            String line = reader.ReadLine();
            Console.WriteLine("Line {0}: {1}", lineNum, line);
            lineNum++;
        }
        reader.Close(); // Dosyayı kapat
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Files

```
StreamWriter variable-name = new StreamWriter(file-name);
```

Örnekler:

```
StreamWriter writer = new StreamWriter("output.txt");
```

```
using System;
using System.IO;
class WriteToFile
{
    static void Main()
    {
        StreamWriter writer = new StreamWriter("c:\\output.txt");
        writer.Write("Bir satır yazılın");
        writer.WriteLine();
        writer.WriteLine("Başka bir satır");
        writer.WriteLine(String.Format("Formatlı text: {0:C} {1:F2}\n", 2.0, 1.234));
        writer.WriteLine("Dosya sonu.");
        writer.Close();
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Grafik Kullanıcı Arayüzü (Graphical User Interface – GUI)

1- TextBox
2- CheckBox
3- ComboBox
4- RadioButton
5- ListBox
6- TextBox
7- Button
8- Label

•System.Windows.Forms
•System.Drawing

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Örnek:

```
using System;
using System.Windows.Forms;
using System.Drawing;

class BasitGUI
{
    static void Main()
    {
        Form window = new Form(); // Bir Form nesnesi yapılandırır
        window.Text = "Bilgisayar Mühendisliği"; // Özellikler düzenlenir
        window.Width = 500;
        window.Height = 300;
        window.BackColor = Color.SkyBlue;
        Application.Run(window); // Ekranda gösterilir
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Örnek:

```
using System;
using System.Drawing;
using System.Windows.Forms;

class MultipleEventGUI
{
    private Form window;
    private TextBox metinKutu;
    private Button tamam, iptal;

    private void tamam_Click(Object sender, EventArgs e)
    {
        MessageBox.Show("Göndermek istediğiniz adres!\n\n" +
            metinKutu.Text, "Dikkat !");
    }

    private void iptal_Click(Object sender, EventArgs e)
    {
        Application.Exit();
    }
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
static void Main( )
{
    MultipleEventGUI gui = new MultipleEventGUI();
    Application.Run(gui.window);
}

public MultipleEventGUI()
{
    // kontrolleri yapılandır
    metinKutu = new TextBox();
    metinKutu.Text = "Buraya Adres Girin";
    metinKutu.Size = new Size(180, 30);
    metinKutu.Location = new Point(10, 10);

    tamam = new Button();
    tamam.Text = "TAMAM";
    tamam.Size = new Size(80, 30);
    tamam.Location = new Point(10, 50);

    iptal = new Button();
    iptal.Text = "IPTAL";
    iptal.Size = new Size(80, 30);
    iptal.Location = new Point(110, 50);
}
```

G. Ü. Bilgisayar Mühendisliği Bölümü

C#

```
// event'ları düzenle
tamam.Click += new EventHandler(tamam_Click);
iptal.Click += new EventHandler(iptal_Click);

// formu yapılandır
window = new Form();
window.Text = "e-posta gönder";
window.Width = 220;
window.Height = 120;

// kontrolleri forma aktar
window.Controls.Add(metinKutu);
window.Controls.Add(tamam);
window.Controls.Add(iptal);
}
```

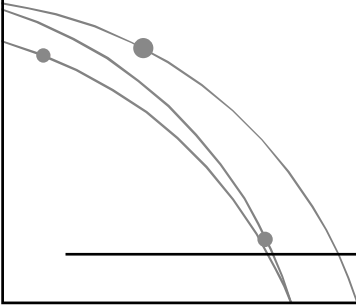


G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Web adresleri

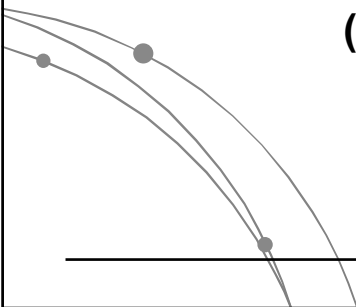
- <http://www.csharp-help.com/what.html>
- <http://msdn.microsoft.com/netframework>
- <http://www.gotdotnet.com>
- <http://www.msdn.net>
- <http://msdn.microsoft.com/vcsharp>



G. Ü. Bilgisayar Mühendisliği Bölümü

C#

Gelecek Hafta Bağlı Listeler (Linked Lists)



G. Ü. Bilgisayar Mühendisliği Bölümü