

# Objektorienterad programmering

## Föreläsning 22

© Copyright  
Mahmud Al Hakim  
[mahmud@dynamicos.se](mailto:mahmud@dynamicos.se)  
[www.webacademy.se](http://www.webacademy.se)

## Agenda

- Rekursion
- Samlingar
  - Listor
  - Mängder
  - Avbildningstabeller

# Rekursion

För att förstå rekursion så måste man förstå rekursion.

- Rekursion handlar om saker som gör något mot sig själva.
- Några exempel
  - En webbsida som via en länk refererar till sig själv.
  - En matematisk funktion som är definierad genom en referens till sig själv.
  - En metod i ett program som anropar sig själv.

Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

## Rekursiv algoritm

- En rekursiv algoritm använder sig själv.
- Exempel på problem som kan lösas med rekursion:
  - Vissa matematiska beräkningar  
T.ex. fakultet, potens, fibonaccital, faktorisering.
  - Komplexa sökalgoritmer (se övning 14.8).
  - Tower of Hanoi
  - Att söka efter en fil i en mapp som innehåller undermappar.  
(tips: <http://support.microsoft.com/kb/303974>)

- Tips

<http://www.codeproject.com/Articles/142292/Recursive-methods-in-Csharp>

Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

## Rekursiva metoder

- En metod kan anropa andra metoder.
- En **rekursiv metod** anropar sig själv.
- Vid exekvering finns det lika många upplagor av metoden som antalet gjorda, oavslutade anrop.
- Varje upplaga av metoden har egna unika värden på parametrar och lokala variabler.

Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

## Rekursion – Grundläggande idé

- Rekursiva metodanrop fungerar på precis samma sätt som andra metod anrop.
- När en metod anropas beräknas först argumenten.
- Sedan genereras minneutrymme för den anropade metodens parametrar och argumenten kopieras dit.
- Vid varje nytt anrop genereras nytt minnesutrymme för parametrarna och för de lokala variablerna.
- Om en viss metod är anropad flera gånger betyder detta att varje upplaga av metoden har sin egen upplaga av parametrarna och de lokala variablerna.
- Därefter utförs satserna i den anropade metoden.
- När dessa satser är klara återvänder man till den punkt varifrån metoden anropades.
- Om en metod anropar sig själv återvänder man till en punkt inne i metoden själv.

Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

## Beräkning av fakultet – del 1

- Ett exempel på rekursion som brukar förekomma i de flesta böcker om programmering är en metod som beräknar den s.k. **fakulteten  $n!$**  av ett heltal  $n$ .
- Fakulteten av ett tal  $n$  definieras på följande sätt:

$$n! = \begin{cases} 1 & \text{om } n = 0 \\ 1 \times 2 \times 3 \times \dots \times n & \text{om } n > 0 \end{cases}$$

- Några exempel

• Om	$n = 0$	$n! = 1$	
• Om	$n = 1$	$n! = 1 \times 1$	$= 1$
• Om	$n = 2$	$n! = 1 \times 2$	$= 2$
• Om	$n = 3$	$n! = 1 \times 2 \times 3$	$= 6$
• Om	$n = 4$	$n! = 1 \times 2 \times 3 \times 4$	$= 24$

Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

## Fakultet – del 2

- Ett annat sätt att skriva definitionen av ett tals fakultet är:

$$n! = \begin{cases} 1 & \text{om } n = 0 \\ n(n-1) & \text{om } n > 0 \end{cases}$$

- Det finns ett fall för vilket värdet är givet ( $n=0$ ).
- Och ett fall där man uttrycker lösningen med hjälp av redan definierade värden.
- Detta sätt att skriva definitionen leder naturligt till en metod som anropar sig själ, en rekursiv metod.

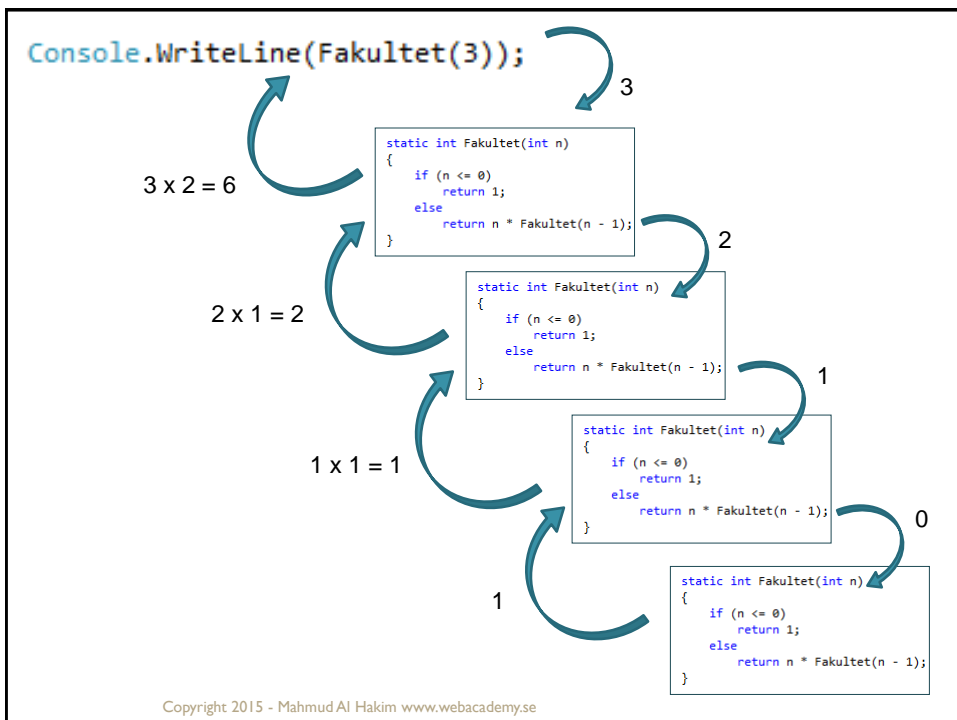
Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

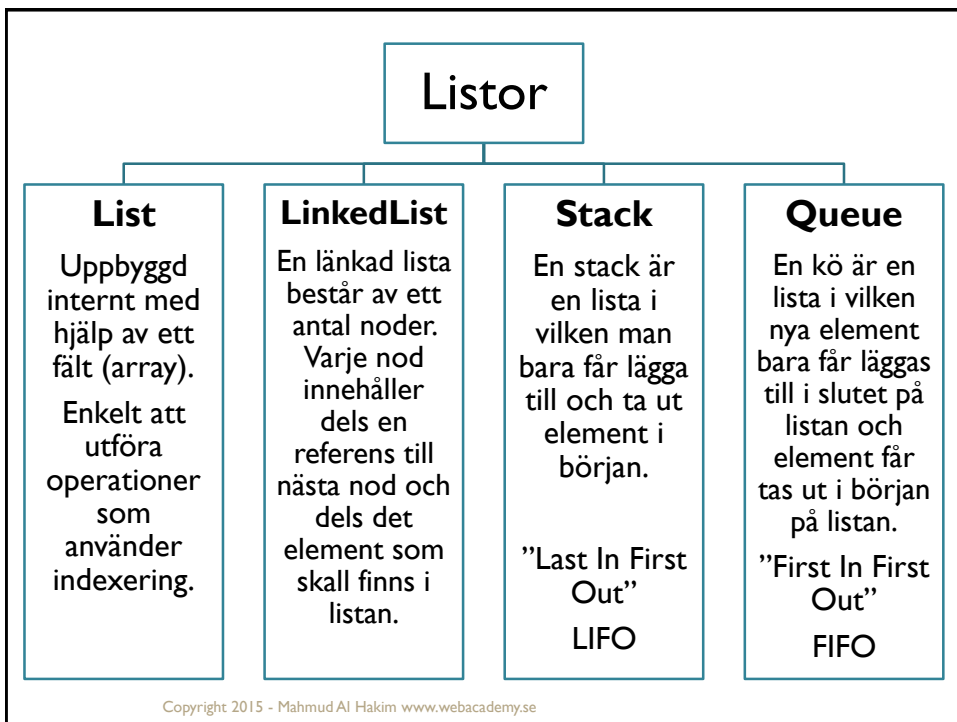
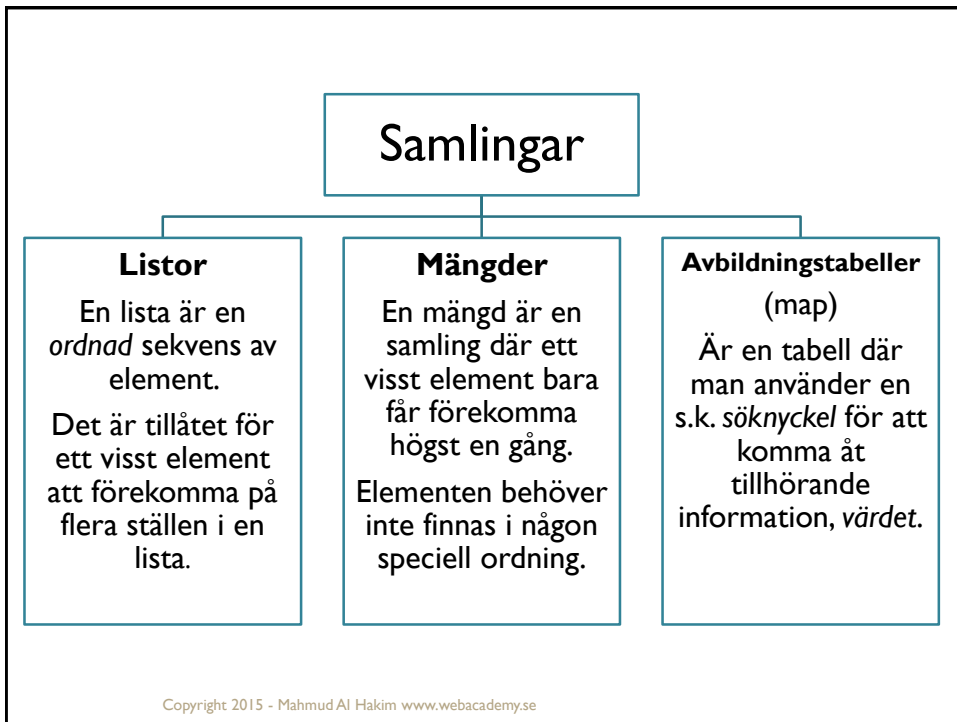
```

static int Faktet(int n)
{
    if (n <= 0)
        return 1;
    else
        return n * Faktet(n - 1);
}

static void Main(string[] args)
{
    Console.WriteLine(Faktet(4));
}

```

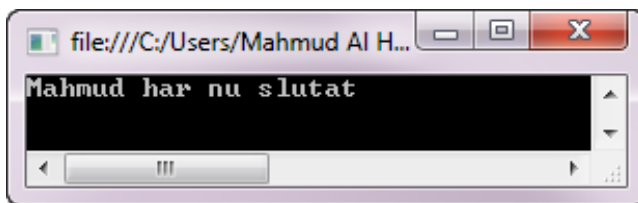




## Stack – Exempel ”Last In First Out” LIFO

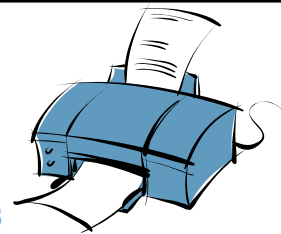


```
Stack<string> personal = new Stack<string>();  
  
personal.Push("Thomas");  
personal.Push("Patricio");  
personal.Push("Mahmud");  
  
Console.WriteLine(personal.Pop() + " har nu slutat");
```

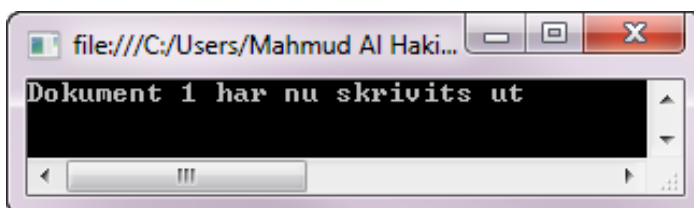


Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

## Queue – Exempel ”Fast In First Out” FIFO



```
Queue<string> dokument = new Queue<string>();  
  
dokument.Enqueue("Dokument 1");  
dokument.Enqueue("Dokument 2");  
dokument.Enqueue("Dokument 3");  
  
Console.WriteLine(dokument.Dequeue() + " har nu skrivits ut");
```



Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

## Mängder – klassen HashSet

- Det är två ting som skiljer en mängd från en lista
  1. Element i en mängd ligger inte i någon speciell ordning.
  2. Ett visst element kan bara förekomma en gång.
- I C# kan man använda den generiska klassen HashSet för att skapa mängder.

```
HashSet<int> m = new HashSet<int>();
```

Man kan initiera en mängd så att den från början innehåller element som kopieras från en annan samling t.ex. från en lista av typen List<int>

```
HashSet<int> m = new HashSet<int>(list);
```

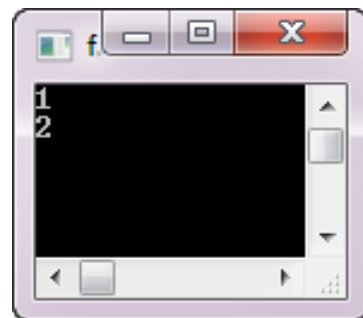
Copyright 2015 - Mahmud Al Hakim www.webacademy.se

## HashSet – Exempel I

```
static void Main(string[] args)
{
    HashSet<int> m = new HashSet<int>();
    m.Add(1);
    m.Add(1);
    m.Add(2);
    m.Add(1);

    foreach (var item in m)
    {
        Console.WriteLine(item);
    }

    Console.ReadKey();
}
```



Copyright 2015 - Mahmud Al Hakim www.webacademy.se

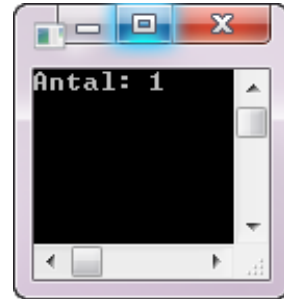


## HashSet – Exempel 2

```
static void Main(string[] args)
{
    HashSet<int> m = new HashSet<int>();
    m.Add(1);
    m.Add(1);
    m.Add(2);
    m.Add(1);

    m.Remove(2);
    Console.WriteLine("Antal: " + m.Count);

    Console.ReadKey();
}
```

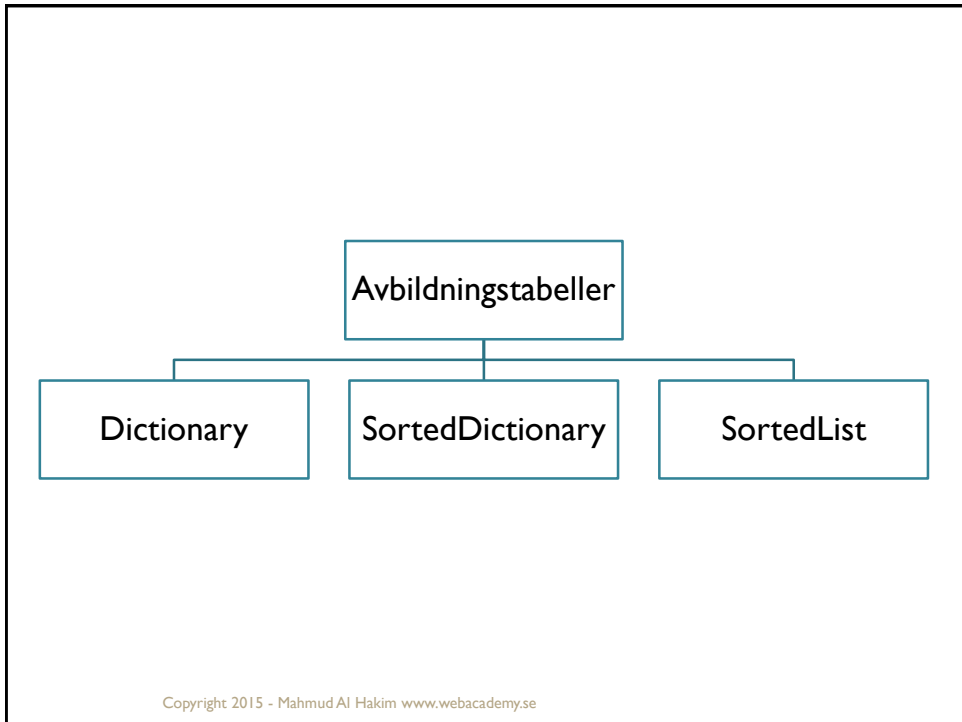


Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)

## Avbildningstabeller

- En avbildningstabell (map på engelska) är en tabell där man använder en s.k. *söknyckel* för att komma åt tillhörande information, *värdet*.
- Ett exempel är ett bilregister, där utgör registreringsnumret *söknyckeln*. Med hjälp av reg. nummer kan man ta fram information om bilen t.ex. bilmärke och årsmodell. *Söknyckeln avbildas på ett värde*.
- En *söknyckel* och tillhörande värde bildar ett par, en s.k. *avbildning*. På engelska används termen *key-value pair*.
- Varje *söknyckel* kan bara avbildas på ett värde (samma bil kan inte ha flera registreringsnummer).


Copyright 2015 - Mahmud Al Hakim [www.webacademy.se](http://www.webacademy.se)



## Dictionary – Exempel

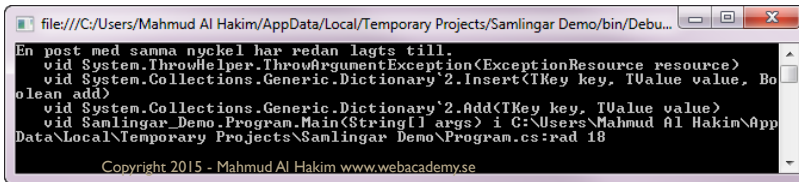
```
Dictionary<string, string> tabell;  
tabell = new Dictionary<string, string>();
```

```
tabell.Add("730202", "Mahmud");  
tabell.Add("730202", "Kalle");
```

 **ArgumentException was unhandled**  
En post med samma nyckel har redan lagts till.

## Dictionary med Try-catch

```
Dictionary<string, string> tabell;  
tabell = new Dictionary<string, string>();  
  
try  
{  
    tabell.Add("730202", "Mahmud");  
    tabell.Add("730202", "Kalle");  
}  
catch (ArgumentException e)  
{  
    Console.WriteLine(e.Message);  
    Console.WriteLine(e.StackTrace);  
}
```

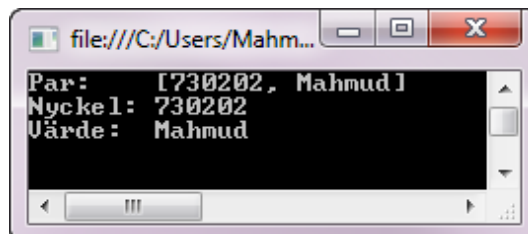


The screenshot shows a console window with the following text:  
file:///C:/Users/Mahmud AI Hakim/AppData/Local/Temporary Projects/Samlingar Demo/bin/Debu...  
En post med samma nyckel har redan lagts till.  
vid System.ThrowHelper.ThrowArgumentException(ExceptionResource resource)  
vid System.Collections.Generic.Dictionary`2.Insert(TKey key, TValue value, Boolean add)  
vid System.Collections.Generic.Dictionary`2.Add(TKey key, TValue value)  
vid Samlingar Demo Program.Main(String[] args) i C:\Users\Mahmud AI Hakim\AppData\Local\Temporary Projects\Samlingar Demo\Program.cs:rad 10  
Copyright 2015 - Mahmud AI Hakim www.webacademy.se

## Att löpa igenom en avbildningstabell

KeyValuePair<K,V> är en generisk structtyp som beskriver avbildningspar. K är nyckel och V är värdet

```
foreach (KeyValuePair<string, string> par in tabell)  
{  
    Console.WriteLine("Par:    " + par);  
    Console.WriteLine("Nyckel: " + par.Key);  
    Console.WriteLine("Värde:  " + par.Value);  
}
```



The screenshot shows a console window with the following text:  
file:///C:/Users/Mahm...  
Par: [730202, Mahmud]  
Nyckel: 730202  
Värde: Mahmud

Copyright 2015 - Mahmud AI Hakim www.webacademy.se