



**Linnéuniversitetet**

Kalmar Vaxjö

## Övningsuppgift

---

# En array baklänges

Steg 1



*Författare:* Mats Looch

*Kurs:* Inledande programmering med C#

*Kurskod:* 1DV402

## Upphovsrätt för detta verk

Detta verk är framtaget i anslutning till kursen Inledande programmering med C# vid Linnéuniversitetet.

### Du får använda detta verk så här:

Allt innehåll i verket En array baklänges av Mats Look, förutom Linnéuniversitetets logotyp, symbol och kopparstick, är licensierad under:



Creative Commons Erkännande-IckeKommersiell-DelaLika 2.5 Sverige licens.  
<http://creativecommons.org/licenses/by-nc-sa/2.5/se/>

### Det betyder att du i icke-kommersiella syften får:

- kopiera hela eller delar av innehållet
- sprida hela eller delar av innehållet
- visa hela eller delar av innehållet offentligt och digitalt
- konvertera innehållet till annat format
- du får även göra om innehållet

Om du förändrar innehållet så ta inte med Linnéuniversitetets logotyp, symbol och/eller kopparstick i din nya version!

Vid all användning måste du ange källan: "Linnéuniversitetet – Inledande programmering med C#" och en länk till <https://coursepress.lnu.se/kurs/inledande-programmering-med-csharp> och till Creative Common-licensen här ovan.

## Innehåll

Uppgift	5
Problem	5
Testa programmet	5
Mål	5
Tips	5
Lösning	6

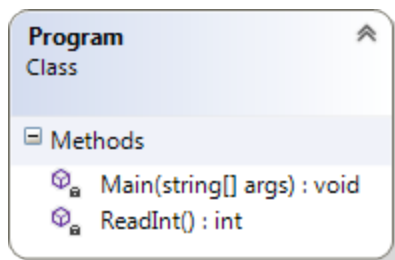


## Uppgift

### Problem

Skriv ett program som ber användaren att ange hur många heltal som ska lagras i en array. Användaren ska sedan mata in heltalen som ska lagras i arrayen. Talen ska skrivas ut baklänges, d.v.s. i omvänd ordning jämfört med hur de matades in.

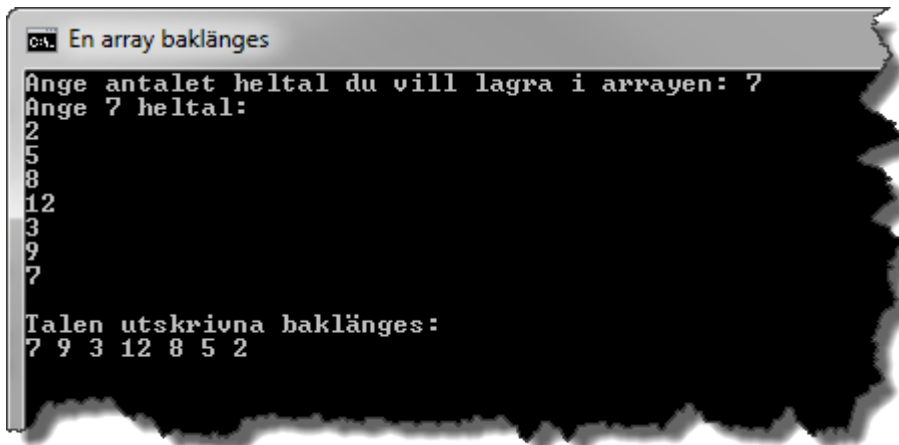
Eftersom programmet kommer att läsa in heltal flera gånger ska du skapa en statisk metod som läser in och returnerar ett heltal. Metoden anropar du sedan då du vill läsa in ett heltal. Självklart ska metoden kunna hantera om användaren råkar mata in något som inte kan tolkas som ett heltal.



Figur 1. Klassdiagram över klassen Program med metoderna Main och ReadInt.

### Testa programmet

Kontrollera att programmet uppfyller de krav som ställs.



```
En array baklänges
Ange antalet heltal du vill lagra i arrayen: 7
Ange 7 heltal:
2
5
8
12
3
9
7

Talen utskrivna baklänges:
7 9 3 12 8 5 2
```

Figur 2.

### Mål

Efter att ha gjort uppgiften ska du känna till hur du:

- Deklarerar en array.
- Kommer åt enskilda element i en array.
- Stegar igenom en array.
- Skriver och anropar en metod som returnerar ett värde.

### Tips

Läs om:

- Arrayer i kurslitteraturen, kapitel 1, under rubriken "Arrays".
- Metoder i kurslitteraturen, kapitel 4, under fram för allt rubriken "Declaring a Method".

Skapa arrayen först efter att användaren matat in hur stor den ska vara.

## Lösning

```
Program.cs  X
ArrayBackwards.Program

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace ArrayBackwards
8 {
9     class Program
10    {
11        static void Main(string[] args)
12        {
13            Console.Title = "En array baklänges";
14
15            // Läser in antalet heltal som ska lagras i arrayen och...
16            Console.Write("Ange antalet heltal du vill lagra i arrayen: ");
17            int count = ReadInt();
18            // ...skapar arrayen.
19            int[] numbers = new int[count];
20
21            // Uppmanar användaren att mata in ett antal heltal...
22            Console.WriteLine("Ange {0} heltal:", numbers.Length);
23            for (int i = 0; i < numbers.Length; ++i)
24            {
25                // ...och metoden ReadInt returnerar ett heltal som
26                // som lagras i det element som variabeln i "pekar ut".
27                numbers[i] = ReadInt();
28            }
29
30            // Arrayens innehåll presenteras från det sista värdet till
31            // det första - "baklänges".
32            Console.WriteLine("\nTalen utskrivna baklänges:");
33            for (int i = numbers.Length - 1; i >= 0; --i)
34            {
35                Console.Write("{0} ", numbers[i]);
36            }
37        }
38
39        // Metoden läser in ett heltal och returnerar det.
40        private static int ReadInt()
41        {
42            // Variablen behöver inte initieras till ett värde eftersom
43            // TryParse tilldelar resultat ett värde om TryParse returnerar
44            // true.
45            int result;
46
47            // Läser in en sträng och försöker tolka strängen till ett heltal
48            // som lagras i result. Misslyckas TryParse med att tolka strängen
49            // till ett heltal returnerar den false och programmet "fastnar" i
50            // while-loopen. OBS! Måste ange out före variabeln result!!
51            while (!int.TryParse(Console.ReadLine(), out result))
52            {
53                Console.WriteLine("\nDu måste ange ett heltal!\n");
54            }
55
56            return result;
57        }
58    }
59 }
```

Figur 3.

Det första som sker i metoden `Main` är att användaren får mata in antalet heltal som arrayen ska lagra. Antalet tal lagras i den lokala variabeln `count` som används då arrayen skapas.

Då arrayen är skapad ”vet” den hur många element den innehåller. Du kan alltid använda egenskapen `Length` för att få reda på hur många element en array innehåller.

Loopar igenom arrayen element för element och lagrar de heltal som metoden `ReadInt` returnerar. Använder egenskapen `Length` i villkorsuttrycket för att kunna bestämma antalet element i arrayen. OBS! Avbrottsvillkoret måste vara ett mindre än värdet `Length` har! Sista elementets index är alltid ett mindre än antalet element i arrayen eftersom det första elementet har index 0.

`Main`-metoden avslutas med att arrayen loopas igenom från det sista elementet till och med det första och respektive elements värde skrivs ut i konsolfönstret.

Metoden `ReadInt` använder metoden `TryParse` för att försöka tolka en sträng som läses in till ett heltal vars värde lagras i out-parametern `result` som skickas med. Detta sätt att tolka en sträng till ett heltal är att föredra framför att använda en ”try-catch”-sats tillsammans med metoden `Parse`. En del upplever den som mer intuitiv och enklare att använda under förutsättning att man vet hur en ”out”-parameter fungerar (se kurslitteraturen, kapitel 4, under underrubriken ”*Output Parameters (out)*”).