

# Hantera fel vid inmatning

# 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 detta verk av Mats Loock, förutom fotografi samt Linnéuniversitetets logotyp och symbol, 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 fotografi samt Linnéuniversitetets logotyp och symbol 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.

# Du har ett problem!

- ✓ Du ska i ett konsolprogram se till att användaren av programmet verkligen matar in ett heltal via tangentbordet.
- ✓ Matar användaren in något annat än ett heltal kraschar programmet!!!

```

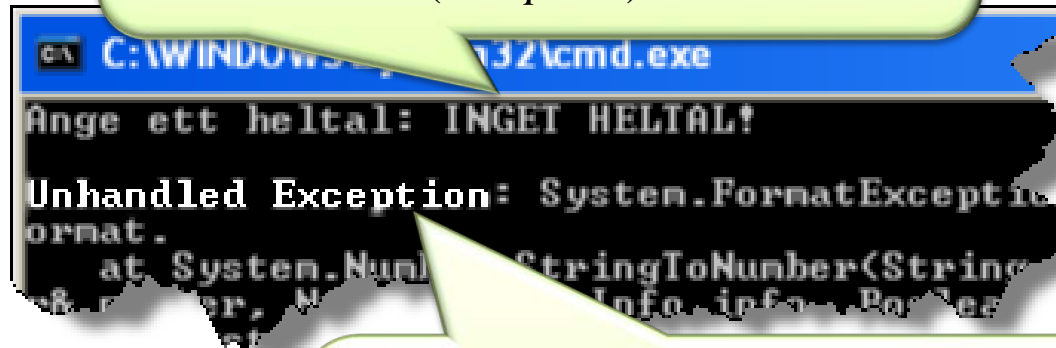
C:\WINDOWS\system32\cmd.exe
ange ett heltal: INGET HELTAL!
Unhandled Exception: System.FormatException: Input string was not in a correct format.
   at System.Number.StringToNumber(String str, NumberStyles options, NumberBuffer& number, NumberFormatInfo info, Boolean parseDecimal)
   at System.Number.ParseInt32(String s, NumberStyles style, NumberFormatInfo info)
   at System.Int32.Parse(String s)
   at Exercise5_2.Program.Main(String[] args) in P:\Kurser\DT2021 - C# för systemadministratörer\Handledningar\Visual Studio\Step1\Exercise5_2\Program.cs:line 14
Press any key to continue . . .
    
```

**Hur kontrollerar jag att  
det är ett heltal?**



# Vad händer då det inte är ett heltal?

Då programmet förväntar sig ett heltal men användaren matar in bokstäver, t.ex. "INGET HELTAL!", kastas ett **undantag** (*exception*).



```
C:\WINDOWS\system32\cmd.exe
Ange ett heltal: INGET HELTAL!
Unhandled Exception: System.FormatException:
    at System.Number.StringToNumber(String s, NumberStyles style, Info info, Pars...
```

**Tar du som programmerar inte hand om undantaget** (felet om du så vill) **avbryts programmet** och dotnetramverket (*.NET Framework*) presenterar ett felmeddelande.

# Varför kastas undantaget?

1. Strängen "INGET HELTAL!" läses in av `Console.ReadLine()`.
2. Metoden `int.Parse()` kan inte tolka "INGET HELTAL!" till ett heltal, och kastar då ett undantag (*throws an exception*).

```
int value = 0;

Console.WriteLine("Ange ett heltal: ");
value = int.Parse(Console.ReadLine());
```

```
Console.WriteLine("Ange ett heltal: ");
value = int.Parse(Console.ReadLine());
```

`int int.Parse(string s, System.Globalization.NumberStyles style, IFormatProvider provider)`  
Converts the string representation of a number in the specified style to an integer.

## Exceptions:

- `System.ArgumentException`
- `System.ArgumentNullException`
- `System.OverflowException`
- `System.FormatException`

TIPS! Placerar du muspekaren över metoden `Parse()` visas ett fönster som bl.a. innehåller de undantag metoden kan komma att kasta.

# Hur tar jag hand om ett undantag?



```
int value = 0;

Console.Write("Ange ett heltal: ");
value = int.Parse(Console.ReadLine());
```

Placera kod som hör ihop och som kan kasta ett undantag i ett "try"-block.

```
int value = 0;

try
{
    Console.Write("Ange ett heltal: ");
    value = int.Parse(Console.ReadLine());
}
catch
{
    Console.WriteLine("FEL! Du måste ange ett heltal.");
}
```

Kod du vill ska exekveras om ett undantag kastas placerar du i ett "catch"-block.

# Så fungerar "try-catch"

Kod som tillhör det **normala programflödet** placerar du i **"try"-blocket**.  
Koden här vill du alltid **försöka exekvera**.

```
try
{
    Console.Write("Ange ett heltal: ");
    value = int.Parse(Console.ReadLine());
}
catch
{
    Console.WriteLine("FEL! Du måste ange ett heltal.");
}
```

Undantag som kastas av kod i **"try"-blocket** fångar du i **"catch"-blocket**. Denna kod **exekveras bara om det inträffar ett fel i "try"-blocket**.

# ...och om jag vill att användaren ska få en ny chans?

Placerar du "try-catch"-blocket i en oändlig "while"-loop tvingar du användaren att mata in ett heltal och inget annat.

```
int value = 0;

// Läser in heltalet som ska analyseras.
while (true)
{
    try
    {
        Console.WriteLine("Ange ett heltal: ");
        value = int.Parse(Console.ReadLine());
        break;
    }
    catch
    {
        Console.WriteLine("FEL! Du måste ange ett heltal.");
    }
}

// Talet för stort?
if (value <= 100)
{
    // t talet?
}
```

Kastas inget undantag exekveras "break"-satsen och "while"-loopen bryts.

```
C:\WINDOWS\system32\cmd.exe
Ange ett heltal: INGET HELTAL!
FEL! Du måste ange ett heltal.
Ange ett heltal: 17
17 > 100, så ett nytt adda tal
```