



**Linnéuniversitetet**

Kalmar Vaxjö

## Laborationsanvisning

# Rita med asterisker

Steg 1, laborationsuppgift 2



*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 Rita med asterisker av Mats Looock, 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

A. Uppgift	5
Problem	5
A-krav	5
Läsvärt	5
B. Uppgift	7
Problem	7
Main	7
ReadOddByte	7
RenderTriangle	8
B-krav	8
Läsvärt	8
C. Uppgift	11
Problem	11
Main	12
IsContinuing	12
ReadOddByte	12
RenderDiamond	12
RenderRow	13
C-krav	13
Läsvärt	13



## A. Uppgift

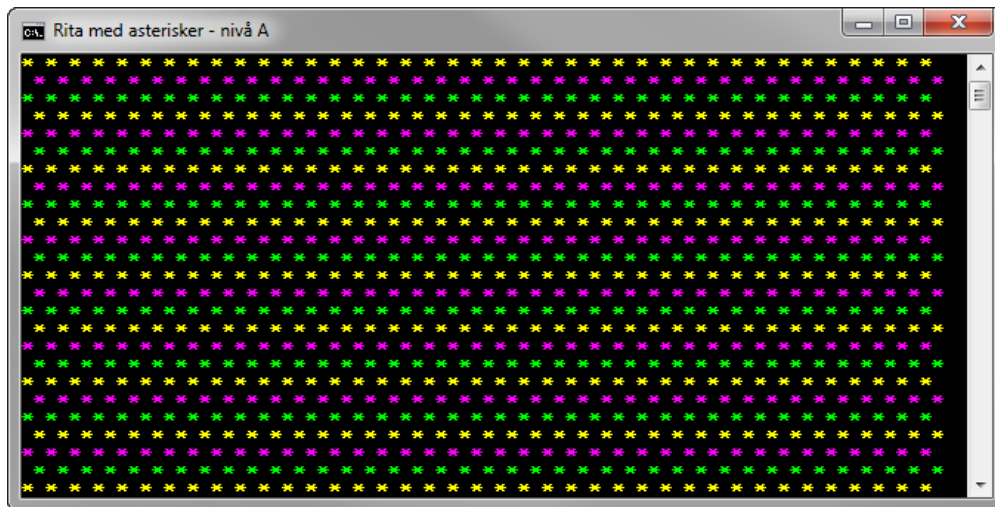
### Problem

Skriv ett program som endast använder följande satser, en gång vardera, för utskrift

```
Console.Write(" ");  
Console.Write("* ");  
Console.WriteLine();
```

Tabell A-1.

för att skapa ett mönster med 39 kolumner och 25 rader enligt figur A.1.



Figur A.1.

Nästlade ”for”-satser ska användas för att åstadkomma rader bestående av kolumner med asteriskecken.

Varannan rad ska vara inskjuten med ett mellanslag. Radernas färger ska variera växelvis mellan gul, magenta och grön. Inskjutningen och variationen av färger löses lämpligen genom att använda modulusoperatörn (%).

### A-krav

1. Satserna i Tabell A-1 ska användas för att skapa figuren. De får dock bara användas en gång vardera.
2. Två ”for”-satser måste användas där den ena ”for”-satsen är nästlad i en den andra ”for”-satsen.
3. Figuren ska bestå av 39 kolumner och 25 rader.
4. Varannan rad ska vara inskjuten ett mellanslag. För att bestämma om en rad ska vara inskjuten eller inte ska en ”if”-sats användas.
5. Radernas färger ska alterneras. Var tredje rad ska vara gul (`ConsoleColor.Yellow`), magenta (`ConsoleColor.Magenta`) respektive grön (`ConsoleColor.Green`). För att bestämma vilken färg en rad ska ha ska en ”switch”-sats användas.

### Läsvärt

- variabler
  - Essential C# 5.0, 13-17.
  - [http://msdn.microsoft.com/en-us/library/hh147285\(VS.88\).aspx#Variables](http://msdn.microsoft.com/en-us/library/hh147285(VS.88).aspx#Variables)

- ”for”-satsen
  - Essential C# 5.0, 130-133.
  - <http://msdn.microsoft.com/en-us/library/ch45axte.aspx>
- ”if”-satsen
  - Essential C# 5.0, 107-114.
  - <http://msdn.microsoft.com/en-us/library/5011f09h.aspx>
- ”switch”-satsen
  - Essential C# 5.0, 135-139.
  - <http://msdn.microsoft.com/en-us/library/06tc147t.aspx>
- %-operatorn
  - Essential C# 5.0, 87-89.
  - <http://msdn.microsoft.com/en-us/library/0w4e0fzs.aspx>
- Hantering av färger i ett konsolfönster
  - <http://msdn.microsoft.com/en-us/library/yae1s0f9.aspx>
  - <http://msdn.microsoft.com/en-us/library/s66hf68a.aspx>
  - <http://msdn.microsoft.com/en-us/library/d3zkyxe.aspx>

## B. Uppgift

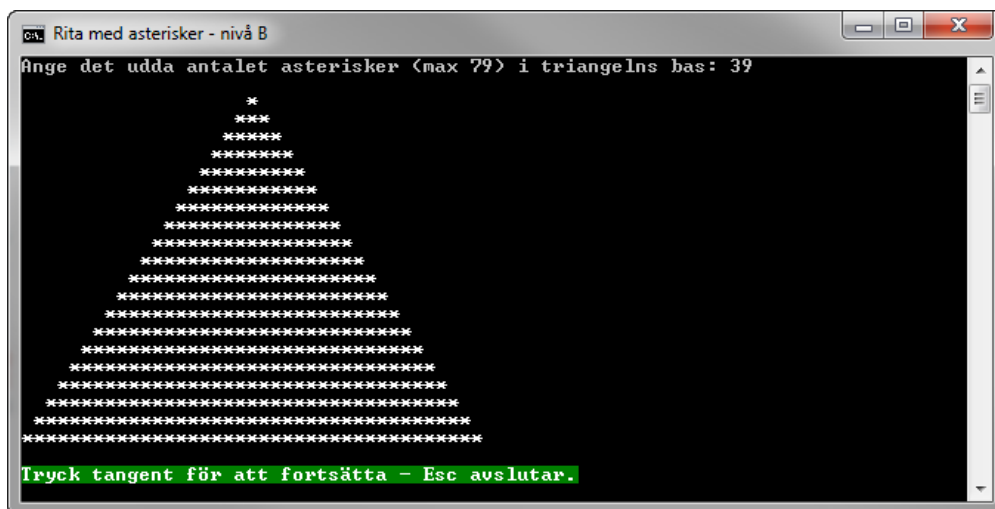
### Problem

Skriv ett program som endast använder följande satser, en gång vardera, för utskrift

```
Console.Write(" ");  
Console.Write("*");  
Console.WriteLine();
```

Tabell B-1.

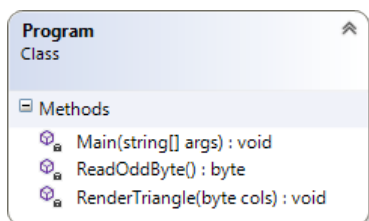
för att skapa en triangel, enligt figur B.1, där användaren ska kunna bestämma det udda antalet asterisker i triangelns bas.



Figur B.1.

Då triangeln ritats ut ska användaren kunna välja att avsluta programmet genom att trycka på Escape-tangenten. Trycker användaren ner någon annan tangent ska användaren på nytt kunna mata in antalet asterisker som triangelns bas ska bestå av.

Programmet ska delas upp de tre privata statiska metoder som återfinns i klassdiagrammet i figur B.2.



Figur B.2.

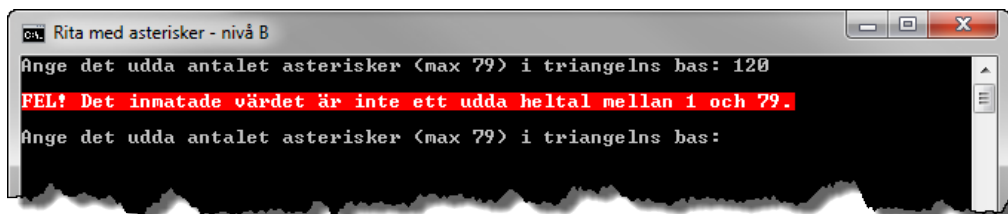
### Main

Denna metod ska anropa metoderna `ReadOddByte` och `RenderTriangle`. Anropen ska placeras i en "do-while"-sats som avslutas då användaren trycker på Escape-tangenten.

### ReadOddByte

Metoden `ReadOddByte` ska returnera ett udda heltal av typen `byte`. Metoden ska säkerställa att inget felaktigt kan matas in, d.v.s. metoden ska kontrollera att det inmatade talet är udda och ligger i det slutna intervallet från 1 till 79 samt ta hand om eventuella undantag som kastas.

Matar användaren in något som inte är ett udda heltal i det slutna intervallet från 1 till 79 ska ett felmeddelande visas, se figur B.3, varefter användaren ska kunna göra en ny inmatning.



Figur B.3.

## RenderTriangle

Metoden `RenderTriangle` har parametern `cols` som ger antalet asterisker triangelns bas ska innehålla. Genom att använda nästlade "for"-satser och satserna i tabell B-1 ska metoden rendera ut en triangel.

## B-krav

1. Programmet ska erbjuda användaren att skapa nya trianglar tills användaren väljer att avsluta programmet genom att trycka på Escape-tangenten.
2. Satserna i tabell b-1 ska användas för att skapa figuren. De får dock bara användas en gång vardera.
3. Användaren ska bestämma hur många asterisker triangelns bas ska ha.
4. Metoden `ReadOddByte` ska returnera antalet asterisker som triangelns bas ska innehålla.
5. Metoden `RenderTriangle` ska rendera triangeln i ett konsolfönster.
6. Antalet asterisker i triangelns bas ska ligga i det slutna intervallet från 1 till 79.
7. Då användaren matar in felaktiga värden ska felmeddelande visas och därefter erbjudas mata in ett nytt värde.
8. Värdet för maximalt antal asterisker och samtliga strängar ska deklarerars som namngivna konstanter.
9. Metoden `ReadOddByte` måste innehålla en "try-catch"-sats för att fånga eventuella undantag som kastas vid inmatning av ett udda heltal.
10. "for"-satser måste användas då triangeln renderas.

## Läsvärt

- "if"-satsen
  - Essential C# 5.0, 107-114.
  - <http://msdn.microsoft.com/en-us/library/5011f09h.aspx>
- "for"-satsen
  - Essential C# 5.0, 130-133.
  - <http://msdn.microsoft.com/en-us/library/ch45axte.aspx>
- "while"-satsen
  - Essential C# 5.0, 127-129.
  - <http://msdn.microsoft.com/en-us/library/2aeyhxcd.aspx>
- "do-while"-satsen
  - Essential C# 5.0, 127-129.
  - <http://msdn.microsoft.com/en-us/library/370s1zax.aspx>



- %-operatorn
  - Essential C# 5.0, 87-89.
  - <http://msdn.microsoft.com/en-us/library/0w4e0fzs.aspx>
- Metoder
  - Essential C# 5.0, 155-167.
  - <http://msdn.microsoft.com/en-us/library/ms173114.aspx>
- Undantag
  - Essential C# 5.0, 194-201.
  - Essential C# 5.0, 423-430.
  - <http://msdn.microsoft.com/en-us/library/0yd65esw.aspx>
  - <http://msdn.microsoft.com/en-us/library/1ah5wsex.aspx>
  - <http://msdn.microsoft.com/en-us/library/ww58ded5.aspx>
- Hantering av färger i ett konsolfönster
  - <http://msdn.microsoft.com/en-us/library/yae1s0f9.aspx>
  - <http://msdn.microsoft.com/en-us/library/s66hf68a.aspx>
  - <http://msdn.microsoft.com/en-us/library/d3zkyxxe.aspx>



## C.Uppgift

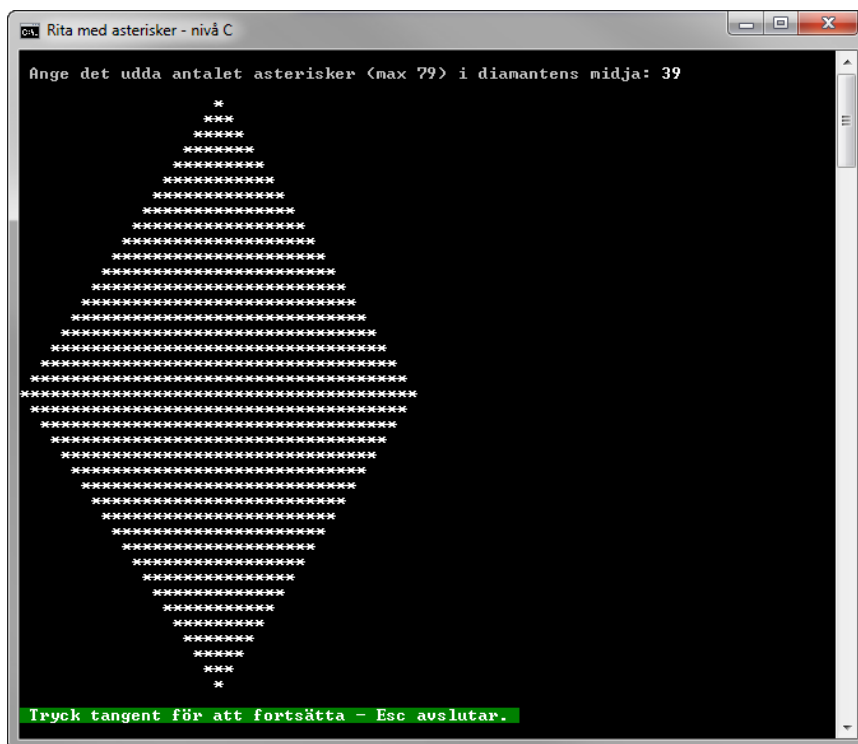
### Problem

Skriv ett program som endast använder följande satser, en gång vardera, för utskrift

```
Console.Write(" ");
Console.Write("*");
Console.WriteLine();
```

Tabell C-1.

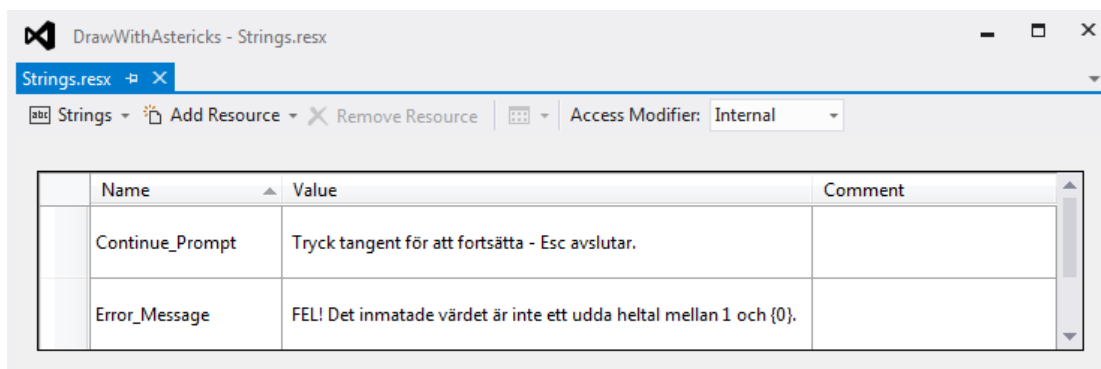
för att skapa en diamant, enligt figur C.1, där användaren ska kunna bestämma det udda antalet asterisker i diamantens midja.



Figur C.1.

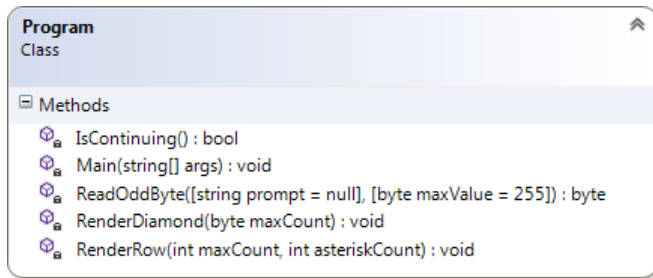
Då diamanten ritats ut ska användaren kunna välja att avsluta programmet genom att trycka på Escape-tangenten. Trycker användaren ner någon annan tangent ska användaren på nytt kunna mata in antalet asterisker som diamantens midja ska innehålla.

Textmeddelanden till användaren ska hanteras med hjälp av en resursfil. Figur C.2 är ett exempel på en resursfil med strängar.



Figur C.2.

Programmet ska delas upp de fyra privata statiska metoder som återfinns i klassdiagrammet i figur C.3.



Figur C.3.

### Main

Denna metod ska anropa metoderna `ReadOddByte`, `RenderDiamond` och `IsContinuing`. Anropen ska placeras i en "do-while"-sats som avslutas då metoden `IsContinuing` returnerar `false` vilket ska ske då användaren trycker på Escape-tangenten.

Då `ReadOddByte` anropas ska det första argumentets värde hämtas från strängresursen. Det andra argumentets värde ska utgöras av en namngiven konstant som bestämmer det maximala antalet asterisker diamantens midja kan innehålla.

Det udda heltal av typen `byte` `ReadOddByte` returnerar ska användas som argument vid anropet av `RenderDiamond`.

### IsContinuing

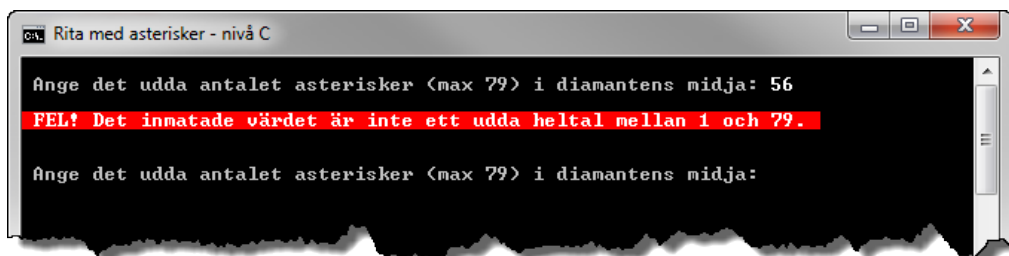
Uppgiften `IsContinuing` har är att presentera ett meddelande som uppmanar användaren att trycka på en tangent för att fortsätta, eller Escape-tangenten för att avsluta programmet. Trycker användaren på Escape-tangenten ska metoden returnera `false`; vilken annan tangent som helst ska leda till att `true` returneras.

### ReadOddByte

Metoden `ReadOddByte` ska returnera ett udda heltal av typen `byte`. Parametrarna ska tilldelas standardvärden som används om metoden anropas och argument saknas för ena, eller båda, parametrarna.

Metoden ska säkerställa att inget felaktigt kan matas in, d.v.s. metoden ska kontrollera att det inmatade talet är udda och ligger i det slutna intervallet från 1 till det värde parametern `maxValue` har samt ta hand om eventuella undantag som kastas.

Matar användaren in något som inte är ett udda heltal i det slutna intervallet från 1 till det värde parametern `maxValue` har ska ett felmeddelande visas, se figur C.4, varefter användaren ska kunna göra en ny inmatning.



Figur C.4.

### RenderDiamond

Metoden `RenderDiamond` har parametern `maxCount` som ger antalet asterisker diamantens midja ska innehålla. Genom att använda "for"-satser ska metoden rendera ut en diamant rad för rad med asterisker genom anrop av metoden `RenderRow`.

## RenderRow

Med hjälp av ”for”-satser och de värden parametrarna `maxCount` och `asteriskCount` har ska metoden rendera ut en rad med mellanslag och asterisker med hjälp av satserna i tabell C-1.

## C-krav

1. Programmet ska erbjuda användaren att skapa nya diamanter tills användaren väljer att avsluta programmet genom att trycka på Escape-tangenten.
2. Satserna i tabell c-1 ska användas för att skapa figuren. De får dock bara användas en gång vardera.
3. Användaren ska bestämma hur många diamanterens midja ska ha.
4. Metoden `ReadOddByte` ska returnera antalet asterisker som diamanterens midja ska innehålla.
5. Metoden `RenderDiamond` ska genom upprepade anrop av metoden `RenderRow` rendera en diamanter bestående av asterisker i ett konsolfönster.
6. Antalet asterisker i diamanterens midja ska ligga i det slutna intervallet från 1 till 79.
7. Då användaren matar in felaktiga värden ska felmeddelande visas och därefter erbjudas mata in ett nytt värde.
8. Värdet för maximalt antal asterisker ska deklarerats som en namngiven konstant.
9. Inga strängar får deklarerats direkt i källkoden utan ska vara placerade i en strängresurs.
10. Metoden `ReadOddByte` måste ha en parameterlista med två parametrar som har standardvärden. Standardvärdena får dock inte användas då metoden anropas.
11. Metoden `ReadOddByte` måste innehålla en ”try-catch”-sats för att fånga eventuella undantag som kastas vid inmatning av ett udda heltal.
12. ”for”-satser måste användas då diamanter renderas.

## Läsvärt

- ”if”-satsen
  - Essential C# 5.0, 107-114.
  - <http://msdn.microsoft.com/en-us/library/5011f09h.aspx>
- ”for”-satsen
  - Essential C# 5.0, 130-133.
  - <http://msdn.microsoft.com/en-us/library/ch45axte.aspx>
- ”while”-satsen
  - Essential C# 5.0, 127-129.
  - <http://msdn.microsoft.com/en-us/library/2aeyhxcd.aspx>
- ”do-while”-satsen
  - Essential C# 5.0, 127-129.
  - <http://msdn.microsoft.com/en-us/library/370s1zax.aspx>
- %-operatören
  - Essential C# 5.0, 87-89.
  - <http://msdn.microsoft.com/en-us/library/0w4e0fzs.aspx>
- Metoder
  - Essential C# 5.0, 155-167.

- <http://msdn.microsoft.com/en-us/library/ms173114.aspx>
- Metoder med frivilliga argument
  - Essential C# 5.0, 189-193.
  - <http://msdn.microsoft.com/en-us/library/dd264739.aspx>
- Undantag
  - Essential C# 5.0, 194-201.
  - Essential C# 5.0, 423-430.
  - <http://msdn.microsoft.com/en-us/library/0yd65esw.aspx>
  - <http://msdn.microsoft.com/en-us/library/1ah5wsex.aspx>
  - <http://msdn.microsoft.com/en-us/library/ww58ded5.aspx>
- Hantering av färger i ett konsolfönster
  - <http://msdn.microsoft.com/en-us/library/yae1s0f9.aspx>
  - <http://msdn.microsoft.com/en-us/library/s66hf68a.aspx>
  - <http://msdn.microsoft.com/en-us/library/d3zkyxxe.aspx>
- Resurser
  - <http://msdn.microsoft.com/en-us/library/9za7fxc7.aspx>