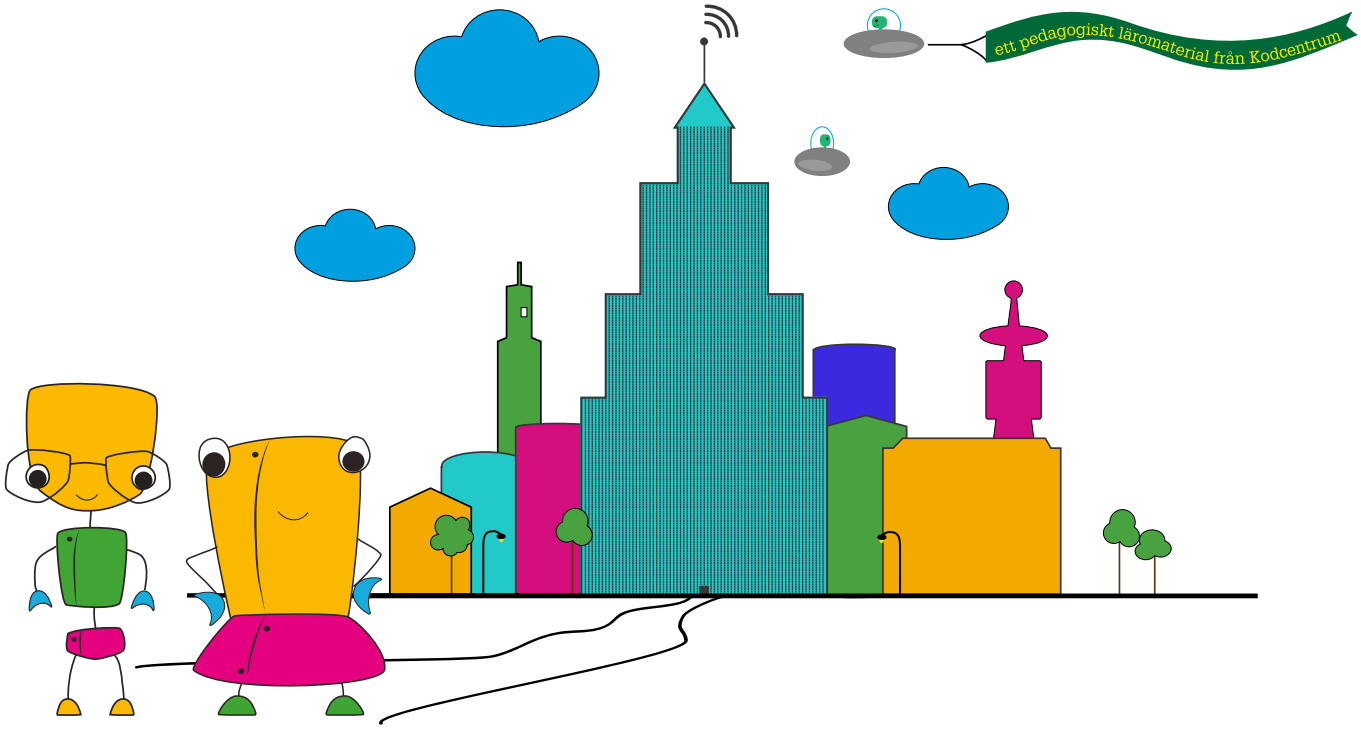
ROBOTDETEKTIVERNA

lärarhandledning

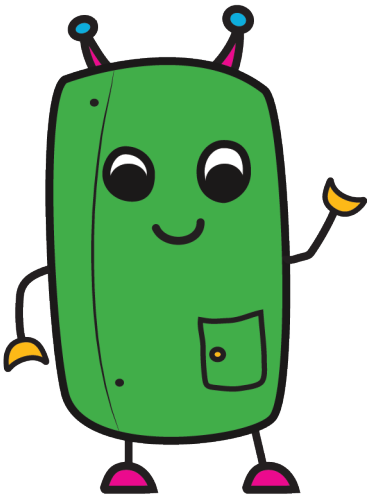


Hej lärare!

Robotdetektiverna är berättelsen om robotarna Else och Random som bor i Nordopolis, en framtida stad där robotar och människor lever tillsammans. Berättelsen innehåller 10 datalogiska utmaningar som på ett lekfullt och klurigt vis övar dig i datalogiskt tänkande - hur du kan titta på problem på ett sätt som liknar hur en dator skulle lösa dem.

Utmaningarna och ramberättelsen är skapad av Marie Gustafsson Friberger, datavetare som driver projekt kring programmering för barn, bland annat Geek Girl Mini och First Lego Legue i Malmö. Illustrationerna är skapade av Elinor Hägg och Erika Granstrand.

Ett par av utmaningarna är inspirerade av Bebras, en tävling i datalogiskt tänkande för barn och unga i åk 2-9 samt gymnasiet. Här finns även många datalogiska uppgifter att lösa även om du inte deltar i tävlingen.
Läs mer på Bebras.se.



Ett pedagogiskt läromaterial
från Kodcentrum.

Text: Marie Gustafsson Friberger
Illustrationer och formgivning:
Elinor Hägg och Erika Granstrand

www.kodcentrum.se
www.kodboken.se

Vad är datalogiskt tänkande?

Datalogiskt tänkande är en förutsättning för att programmera. Det kan beskrivas som en problemlösningprocess för att beskriva, analysera och lösa problem på ett sätt så att datorer kan hjälpa till*.

Datalogiskt tänkande handlar bland annat om att bryta ned problem i mindre delar, skapa algoritmer (instruktioner) för hur dessa delar kan lösas med en dators hjälp, hitta mönster som kan användas för att upprepa sådant som ska hända flera gånger, generalisera lösningar så att de kan användas för olika problem och att kunna identifiera avvikelser som kan orsaka fel i lösningen.

Inom programmering är det viktigt att kunna resonera sig fram och förklara för någon annan hur du har tänkt dig lösa problemet. Det innebär också att vara kreativ och kunna fundera kring olika möjliga lösningar eftersom det ofta finns flera sätt att lösa problem på!

* Fredrik Heintz och Linda Mannila, "Datalogiskt tänkande för svenska grundskolan – Vad, hur och varför?" 2015.

Datalogiska utmaningar i tre nivåer

Utmaningarna är indelade i tre nivåer, markerade med 1-3 pusselbitar. Du kan med fördel dela upp utmaningarna under en längre tid och låta eleverna lösa dem parallellt med att de får en introduktion till och därefter fortsätter utforska programmering i skolan, till exempel med våra lektionsupplägg på **Kodboken.se/kodaiskolan**.

Nivå 1: Det är ingen kod i dessa uppgifter och det räcker med att eleverna har fått en kort introduktion till programmering innan de gör uppgifterna.

Nivå 2: Dessa innehåller kodblock, likt blockprogrammering i exempelvis Scratch. Här behöver eleverna programmerat lite tidigare och känna till vad repetition, villkor och variabler innebär.

Nivå 3: Här är uppgifterna lite klurigare, koden är längre och skriven i så kallad pseudokod, ett sätt att beskriva algoritmer till en dator men utan att skriva det i ett specifikt programmeringsspråk. Eleverna bör vara bekväma med repetition, lite mer avancerad användning av villkor och ha fått en introduktion till listor.

Programmering, kod och pseudokod

När du programmerar använder du ett programmeringsspråk, som sedan kompileras (översätts) till maskinkod, det språk som datorn förstår. Det finns många olika programmeringsspråk som är bra för olika saker.

De flesta programmeringsspråk är textbaserade, det vill säga du skriver koden i text. Men när du ska lära dig att programmera och vill öva ditt datalogiska tänkande, kan du komma igång med hjälp av blockprogrammering eller genom att använda pseudokod.

Blockprogrammering är när du programmerar med visuella kodblock, till exempel i Scratch. Kodblocken är då förprogrammerade.

Pseudokod är ett sätt att beskriva hur du tänker bygga upp din kod, men utan att skriva det i ett specifikt programmeringsspråk. Pseudo betyder att något inte är äkta, pseudokod ser ut som kod men är inget riktigt programmeringsspråk och kan inte köras i en dator.

Om du vill skissa upp ditt program på papper, kanske för att diskutera koden med en kompis, kan du skriva med pseudokod. Då kan du skriva med naturligt språk för att göra koden lättare att förstå. På så sätt är det enklare att resonera kring hur du vill att ditt program ska fungera, innan du sätter igång och kodar det på riktigt. Fungerar det som du tänkt? Finns det några fel, några buggar? Hur kan det göras annorlunda – går det att förenklas?

Vad har detta med datorer och kod att göra?

Till varje utmaning finns en kort faktatext kring hur temat kopplar till datorer och kod. Dessa texter kan du med fördel använda som underlag för en presentation eller en gemensam diskussion i klassrummet. Frågorna som du hittar i faktatexten och i utmaningens uppgifter, kan du använda för att diskutera temat.

Om ni vill fördjupa er inom datalogiskt tänkande, programmering och kod, samt kring teman som ”vad är en dator?” och ”hur pratar datorer?”, kan du och dina elever läsa om det i vår textsamling **Kodboken.se/fattakoden**.

Att lösa utmaningarna

Utmaningarna innehåller både frågor som har ett rätt eller fel svar, samt frågor där eleverna själva ska ge förslag på lösning och där det finns många möjliga svar på frågorna.

Här ger vi exempel på vad sådana svar skulle kunna vara, men det är viktigt att du som lärare betonar att problem ofta har flera olika möjliga lösningar och att det inte finns något givet rätt eller fel svar på dessa frågor.

Uppmuntra gärna dina elever till att försöka hitta lösningar, gärna tillsammans. Programmering handlar om problemlösning och här gäller det att testa sig fram och att felsöka när det inte fungerar som det är tänkt.

Att ta hjälp av andra är viktigt, som programmerare kan du t ex jobba parvis (*parprogrammering*) eller i mindre grupper (*mobbprogrammering*).

Du som lärare får också gärna vara med och diskutera med eleverna om hur de resonerar. Om de fastnar vid någon fråga, kan ni resonera tillsammans för att hitta en lösning?

Diplom efter slutförda utmaningar

I slutet av Robotdetektiverna tilldelas alla robotar en medalj och ett diplom som visar att de tillhör Nordopolis detektivrobotar. Här får även eleven ett slags diplom där du kan fylla i elevens namn efter att denne är klar med utmaningarna - eller gruppens namn om de har arbetat tillsammans.

Tanken är att de får diplomet när de slutfört utmaningarna, oavsett om de har lyckats lösa alla uppgifter eller inte. Det viktigaste är att de har försökt att hitta lösningar och resonerat sig fram tillsammans!

Utmaning 1: Någon har stulit mitt minneskort!

1. Svaret är: Team 2, Dataintrång, Bråttom.

2. Svaret är:
Team-symbol: team 2,
Orsaks-symbol: stöld,
Tids-symbol: bråttom.

3. Här får eleverna hitta på en egen symbol. Vilken kategori tillhör den? Går den att tolka lätt?

4. Här får eleverna hitta på egna meddelanden och låta en kompis tyda det. Gick det att tyda?

5. Ordningen av symbolerna spelar ingen roll i det här sammanhanget. Oavsett i vilken ordning du sätter symbolerna får du fram samma information.

Utmaning 2: Lägg mojängen i verktygsfacket

1. Svaret är: Scannern längst ut, därefter detektorn och längst in mojängen.

2. Här får eleverna hitta på ett eget uppdrag, där verktygen ska användas i följande ordning: detektor, mojäng, scanner.

3. Här får eleverna hitta en egen lösning och det finns inte bara ett rätt sätt att lösa problemet på.

Ett exempel på lösning kan vara att designa robotarnas fack så att de fungerar som en *kö*, det vill säga att det verktyget som stoppas in först, kommer ut först. Det kan till exempel lösas med en öppning mellan de båda facken, som gör att verktygen som sätts in, kommer ut genom det tomma facket. När du använt klart verktyget kan du lägga tillbaka verktyget i det första facket igen. På så sett fungerar det som en loopad verktygslåda!

Utmaning 3: Hemliga meddelanden

1. Svaret är: ÄT INTE UPP ALLT.

2. Det är viktigt att veta hur många tabellkolumner meddelandet använder sig av, detta är chiffrets *nyckel*. Det gör det enklare om du dessutom vet hur många tabellrader meddelandet använder, men det är inte nödvändig information för att lista ut meddelandet.

3. Här får eleverna själva skapa ett meddelande och ge till en kompis för att lösa det. Fortsätt till fråga 4 för att lösa meddelandet.

4. Här får eleverna testa sig fram, men det kan vara svårt utan att ha nyckeln, det vill säga antal tabellkolumner. Kunde eleverna lösa meddelandet utan nyckeln? Med nyckeln? Hur svårt var det att lösa?

5. Detta chiffer är inte särskilt bra på att dölja information, det är rätt enkelt att testa sig fram och hitta lösningen. Som det står i faktatexten används mycket mer komplicerad kryptering nu för tiden för att dölja innehåll i digitala meddelanden, som exempelvis mail.

Utmaning 4: På patrull

1. Svaret är:
Bugg 1: värdet 1000 ska vara 200,
Bugg 2: värdet 3 ska vara 2,
Bugg 3: värdet 80 ska vara 90,
Bugg 4: ”sväng höger” ska vara ”sväng vänster”.

2. Svaret är 400 gånger (2 gånger x 200 våningar).

3. Här får eleverna hitta på sin egen bugg. Har de gjort ett fel som påverkar koden så att robotdetektiven gör fel? Vad händer i deras kod?

Utmaning 5: Loopa ut från lagret

1. Svaret är:

```
repetera tills når grönt område
    om kan gå framåt
        gå 1 steg framåt
    annars
        sväng vänster
```

2. Här får eleverna hitta på sin egen labyrint och låta en kompis lösa den. Fungerar det för kompisen att ta sig genom labyrinten med fyra kodblock?

Utmaning 6: Varför får inte pappa samma reklam som jag?

1. Svaret är:
Inlägg 1: Träning och spa (-3 för bild på heminredning, -2 för långt ord) Inlägg 2: Nya robotleksaker (+3 för bild på robotleksak, +2 för ordet ”Bzzzz”)
Inlägg 3: Nya robotleksaker (+3 för bild på robotleksak, +2 för orden ”BlipBlopp”, -2 för långt ord)
Inlägg 4:Nya robotleksaker (0 för bilden, +2 för ordet ”Tjohej”)

2. Svaret är inlägg 3. Texten är skriven på ett ”vuxet” sätt och klagar på alla nya robotleksaker. Anledningen till att algoritmen tolkar det som ett barn är för att det är en bild på en robotleksak och personen använder sig av ordet ”BlippBlopp” som är populärt bland barn i Nordopolis.

3. Här får eleverna hitta på sin egen bild och text. Tillsammans ska summan bli mellan -1 och +1. Tänk på att varje ord som ger poäng ska räknas, till exempel ger varje ord som innehåller minst 10 bokstäver ger -2 poäng.

Utmaning 7: Kod röd på våning 13

1. Svaret är att roboten först ska kalla på förstärkning, innan den undersöker våningen.

2. Svaret är att roboten ska undersöka våningen själv, eftersom det finns robotdetektiver i närheten som kan hjälpa till om det behövs.

3. Här får eleverna hitta på en egen algoritm utifrån den förenklade regeln. Enklast ser den ut så här:

```
Om kod röd då
    Anropa förstärkning
Annars
    Undersök närmare
```

Utmaning 8: Debugga värmen

1. Svaret är: Hackaren har ändrat ett enhet för Temperatursensor från Celsius (C) till Fahrenheit (F). Därför läser värmesystemet av temperaturen i Fahrenheit, där 12 grader Celcius är lika med 54 grader Fahrenheit.

2. Svaret är: Ändra ”F” till ”C” för Temperatursensor.

Utmaning 9: Samla på spår

1. Svaret är: Krossade minneskort (Våning 18).

2. Svaret är: Slime (Våning 71).

3. Här får eleverna hitta på ett eget spår och skriva in koden för det. Ett exempel kan vara fotavtryck som kan vara spår efter en tjuv:

Om (fotavtryck)
Lägg till i lista (våning + ”Fotavtryck”)

Utmaning 10: Var är robotkatten?

Här behöver eleverna göra en tabell och skriva in sina uträkningar i den. Därefter får de jämföra vilken algoritm som löser problemet snabbast, algoritm 1 eller 2:

	Om hissen alltid tar 1 min	Om hissen alltid tar 10 min
Algoritm 1	8x1 + 7x7 = 57	8x10 + 7x7 = 129
Algoritm 2	39x1 = 39	39X10 = 390
Egen algoritm		

1. Här ska eleverna beskriva med egna ord.
Algoritm 1 säger åt Else att börja på våning 48 och undersöka först städskrubb A och sedan städskrubb B. Därefter ska Else åka ned en våning. Roboten ska sedan upprepa samma procedur tills den är på våning 1. Då ska Else åka till våning 49 och utföra samma procedur, fast istället åka uppåt en våning med hissen ända tills den är på våning 200.

Algoritm 2 säger åt Else att börja på våning 1 och först undersöka städskrubb A och sedan åka upp en våning. Roboten ska sedan upprepa samma procedur tills den är på våning 200. Därefter ska Else, med start på våning 200, undersöka städskrubb B och sedan åka ned en våning och upprepa samma procedur tills den är på våning 1.

2. Svaret för algoritm 1 är: Om hissen alltid tar 1 minut: 57 minuter.
Om hissen alltid tar 10 minuter: 129 minuter.

3. Svaret för algoritm 2 är: Om hissen alltid tar 1 minut: 39 minuter.
Om hissen alltid tar 10 minuter: 390 minuter.
Algoritm 1 är alltså snabbast om hissen alltid tar 10 minuter, medan algoritm 2 är snabbast om hissen alltid tar 1 minut.

4. Här får eleverna själva hitta på en algoritm och räkna ut hur lång tid den tar att utföra om hissen alltid tar 1 minut respektive 10 minuter.