



Programmering

ELEV

Du ska nu få en första introduktion till programmering. Du kommer att arbeta med så kallad blockprogrammering med hjälp av webbsidan Code.org och språket Blockly.

Även om du testat programmering tidigare så kan du få ut något av uppgiften eftersom du kan lära dig/repetera en del programmeringsbegrepp samt få erfarenhet av *parprogrammering*, *felsökning* och *problemlösning*. Kanske hittar du likheter mellan det programspråk du använt och Blockly?

Det är bra om du gjort hela eller delar av programmeringsuppgiften i ”Nivå ETT” först, så du förstår principen för programmeringsspråket.

SYFTE

Syftet med övningen är att du ska

- utveckla datalogiskt tänkande
- träna dig i att lösa problem
- bekanta dig med ett digitalt hjälpmedel
- få en introduktion till blockprogrammering
- repetera begreppen algoritm och loop
- lära dig begreppet funktion
- få erfarenhet av att felsöka/debugga ett program
- testa på parprogrammering
- träna på att ange olika vinklar i grader
- få erfarenhet av att använda digitala verktyg för att rita olika geometriska figurer

REDOVISNING/BEDÖMNING

Din lärare berättar hur du ska redovisa dina uppgifter. Det kan till exempel vara genom diskussioner i helklass eller genom att du lämnar in svar på frågorna.

DEL 1: Funktioner, introduktion

Ibland när man programmerar vill man återanvända ett kodblock flera gånger. Då är funktioner användbara. Man kan säga att en funktion är ett kodblock som man lagrar för att kunna återanvända vid ett senare tillfälle.

Funktioner är användbara eftersom

- Man slipper skriva samma sak flera gånger.
- Det blir enklare att få en överblick i stora program.
- Om man ska göra en förändring någonstans i programmet, behöver man bara göra det på ett ställe.

En refräng på en låt skulle kunna liknas vid en funktion. Istället för att skriva ut hela refrängen igen, kan man skriva ”refräng” mellan verserna. Titta t.ex. på texten till Michael Jacksons låt ”Billy Jean”.

Vi återgår till *algoritmen* för att borsta tänderna från Programmeringsuppgiften i NIVÅ TVÅ.

1. Ta fram tandkrämen
 2. Ta fram tandborsten
 3. Skruva av locket på tandkrämstuben
 4. Smeta tandkräm på tandborsten
 5. Öppna munnen
 6. Borsta tänderna
- ... o.s.v.

Algoritmer är en slags beskrivning av vad som ska göras och i vilken ordning.

Vi skulle kunna göra om hela det här kodblocket till en *funktion* och kalla funktionen för ”Tandborstning”.

Tänk dig att du skulle programmera dig själv, alltså skriva ett program över din hela dag från morgon till kväll. Du har nu kommit fram till tandborstningen på kvällen. Istället för att skriva hela algoritmen för hur man borstar tänderna, använder du istället funktionen ”Tandborstning”.

- A. Kan du komma på någon mer funktion från din dag? Något du gör flera gånger om dagen. Vilken algoritm skulle den funktionen innehålla?
- B. Så här kan ett funktions-block se ut i Blockly:



Vad tror du den här funktionen innehåller för algoritm?

DEL 2: Funktioner, parprogrammering och felsökning/debugging

- A. Gå in på <https://studio.code.org/s/course4/stage/12/puzzle/1>.
- B. Försök göra uppgift 1 själv. Därefter gör ni uppgift 2 tillsammans i klassen.

Uppgiften beskrivs överst:



Kan du rita en fyrkant som är 100 bildpunkter i längd, med en upprepa-loop?

C O
D E

- C. Din lärare visar filmen ”Redigera funktioner” ([Creating functions in code studio by Bill Gates and Saloni](#)) för er. Filmen är på engelska.

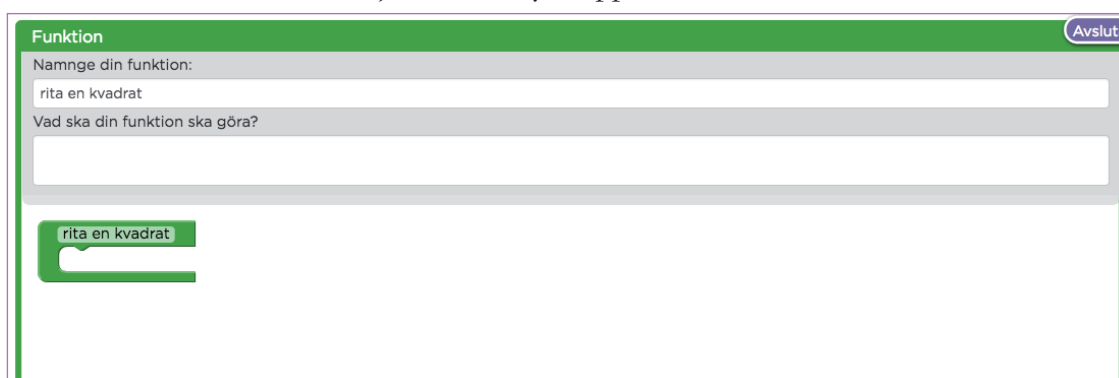
Diskutera tillsammans: Vad är en funktion?

Kort beskrivning på svenska över hur du gör för att skapa/redigera en funktion.

Börja med att beskriva funktionen genom att klicka på den blå rutan (”redigera”).



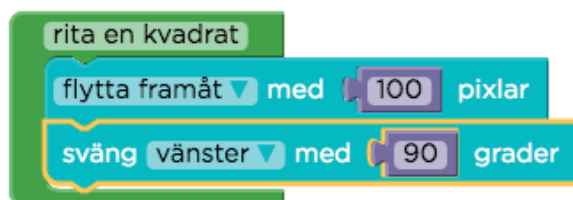
Då kommer följande arbetsyta upp:



Här kan du namnge din funktion. Namnet bör spegla innehållet i funktionen.

I rutan nedanför kan du beskriva lite mer utförligt vad funktionen ska göra.

I den tredje rutan finns funktionen. I den ska du placera en algoritm genom att dra hit block från verktygsfältet. T.ex.:



När du är färdig med din funktion, klickar du på ”Avsluta” uppe till höger.

- D. Fortsätt med uppgifterna 3–6.

Många programmerare använder sig av så kallad parprogrammering (Pair programming).

Då arbetar två programmerare tillsammans vid en gemensam dator. Den ena skriver kod medan den andra granskar varje kodrad när den matas in. De två programmerarna växlar ofta mellan rollerna. Om det blir fel: Felsök koden tillsammans.

Använd tekniken för parprogrammering och felsökning/debugging för kommande uppgifter.

- e. Ibland har man flera olika funktioner i samma program. Gör uppgift 7 som innehåller två olika funktioner.
- f. Man kan även placera en funktion inuti en annan funktion. Det får du se exempel på när du gör uppgifterna 8 och 9. För att öka hastigheten på ”konstnären” kan du dra det här reglaget åt höger:



- g. Sammanfatta: Vad är en funktion och varför använder man dem?
- h. Uppgift 11 testar din förmåga att använda dig av funktioner och loopar som du använt under tidigare uppgifter.

Om du behöver en ledtråd klickar du på glödlampan:



I uppgift 11 är verktygsfältet indelat i olika kategorier, t.ex. loopar och funktioner. Ett tips i den här uppgiften är att ta ett litet steg i taget och sedan köra programmet.

- i. Om du vill arbeta vidare med funktioner kan du fortsätta med Nivå 14 [”Konstnären: Funktioner med parametrar”](#).