

## Kapitel 3: Villkorssatser

I denna andra aktivitet för kapitel 3 så kommer du att lära dig att arbeta med If...Then...EndIf-satser.

Som du kan se i skärmbilden till höger så finns det fyra olika typer av If-satsmallar hos TI-Nspire™ CX. De används för att villkorligt bearbeta programsatser. Detta kallas ibland för *förgrening* i ett program eftersom man i programmet kan följa någon av flera olika vägar genom koden.

Det är tillrådligt att välja dessa If-strukturer från menyn eftersom alla delar som behövs kommer att infogas i koden på rätt plats. Därefter backar du i koden och fyller i "luckorna".

Skärmbilden till höger visar resultatet när du valt **2: If...Then...Endif** från Kontrollmenyn. Därefter fyller man i villkoren mellan **If** och **Then** och åtgärderna mellan **If** och **Endif**.

Vi ska nu skriva ett program som låter användaren mata in värden för variablerna  $x$  och  $y$  och sedan låta programmet bestämma i vilken kvadrant som punkten  $(x, y)$  ligger och därefter också bestämma tecken på koordinaterna i den kvadranten.

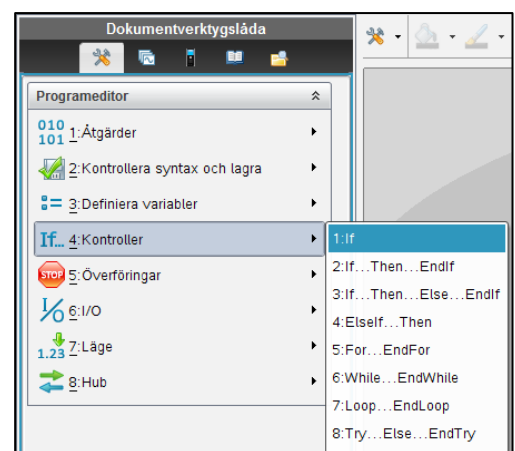
Den första och ofullständiga delen av programmet visas här till höger. Ovanför If-satsen så behövs två **Request**-satser (en för  $x$  och en för  $y$ ). För att spara tid kan du kopiera och klistra in **If...Endif**-strukturen och därefter redigera satserna för de tre andra kvadranterna.

*Glöm inte att separera nyckelordet "and" från omgivande text med blankstegstecken.*

## Övning 2: If...Then-satser

## Syfte:

- Undersöka **If...Then...Endif**-strukturen
- Skapa sammansatta villkor med relationsoperatorerna och de logiska operatorerna
- Skriva ett program som använder **If...Then...Endif**-strukturen



```
* ifthen 6/6
Define ifthen()=
Prgm
{
}
If Then
{
}
EndIf
{
}
EndPrgm
```

```
* ifthen 8/8
Define ifthen()=
Prgm
{
}
If x>0 and y>0 Then
Disp "Kvadrant 1"
Disp "x är positivt"
Disp "y är positivt"
EndIf
{
}
EndPrgm
```

*Kopiera och klistra in kod som du vill återanvända och modifiera den sedan. Testa programmet med olika värden på **x** och **y**. (Finns det några värden som inte ger något resultat?)*

Nedan finns den fullständiga programkoden.

```
Define ifthen(=
Prgm
:Request "x=",x
:Request "y=",y
:If x>0 and y>0 Then
: Disp "kvadrant 1"
: Disp "x är positivt"
: Disp "y är positivt"
: EndIf
: If x<0 and y>0 Then
: Disp "kvadrant 2"
: Disp "x är negativt"
: Disp "y är positivt"
: EndIf
: If x<0 and y<0 Then
: Disp "kvadrant 3"
: Disp "x är negativt"
: Disp "y är negativt"
: EndIf
:If x>0 and y<0 Then
: Disp "kvadrant 4"
: Disp "x är positivt"
: Disp "y är negativt"
: EndIf
:EndPrgm
```

**Lärarkommentar:** Programmet visar inga resultat om x eller y eller båda har värdet noll. Då ligger ju punkten på någon av axlarna eller i origo. Du kan ge eleverna uppgiften att utvidga programmet så att det täcker in alla alternativ.