Kapitel 3: Villkorssatser — Övning 3: If…Then…Else…End Delbarhet

I denna tredje aktivitet för kapitel 3 kommer du att lära dig om en annan form av If-satser i TI Basic samt värdet av att förstå numeriska algoritmer.

Syfte:

- Se strukturen hos If…Then…Else…End-satser.
- Lära sig hur man testar om ett värde är ett heltal.
- Vad är en **algoritm**?

Lärarkommentar: I denna aktivitet introduceras begreppet algoritm. Principerna är viktiga om man ska bli en framgångsrik programmerare. Vi kommer inte att gräva djupare i algoritm-begreppet men idén om att ha en plan innan man börjar skriva ett program är avgörande. Programskrivande kräver ju först förståelse för själva uppgiften och kunskap om programkommandona och vad man vill åstadkomma. Först då kan man konstruera en algoritm för att lösa problemet.
I If…Then…Else…End-satser är syntaxen viktig. Var och ett av de fyra nyckelorden uppträder som separata satser i programmet och det är bara **If** som kan ha ett villkor i satsen.

Om If…Then…Else…End

I den förra aktiviteten lärde du dig att använda **If…Then…End-**satser. Det finns tillfällen då vi behöver ta två olika vägar beroende på ett villkor. Strukturen på denna nya **If…Then…Else…End-**instruktion är liknande:

If <villkor>
Then
<Sant block>
Else
<Falskt block>
End

Obs:

Then, **Else**, *och* **End** *har egna rader.*

<Sant-blocket> *är den uppsättning av satser som exekveras när <villkor> is sant.*

<Falskt-blocket> *är den uppsättning av satser som kommer att exekveras när <villkor>* ***inte*** *är sant.*

Alltså kommer ett av dessa block att exekveras.

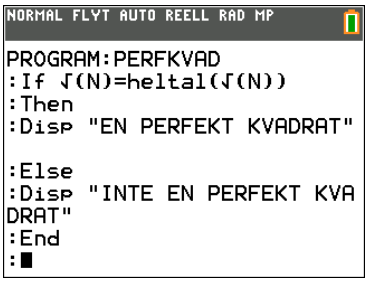
Programmering med If…Then…Else…End

Vi ska nu skriva ett program som talar om ifall ett inmatat tal är en perfekt kvadrat eller inte. En perfekt kvadrat är kvadraten på ett heltal, t.ex. 25 (5^2). Den metod som används här är att beräkna kvadratroten av talet och kontrollera och om det är ett heltal. Du har programlistningen till höger. Den är uppdelad på två skärmar för att du ska kunna se alla rader i koden.

```
NORMAL FLYT AUTO REELL RAD MP
PROGRAM:PERFKVAD
:■lrHome
:Input "MATA IN ETT TAL ",
N
:If √(N)=heltal(√(N))
:Then
:Disp "EN PERFEKT KVADRAT"

:Else
:Disp "INTE EN PERFEKT KVA
```

***Obs**: Tryck* 2nd x² *för att skriva kvadratrotssymbolen*.
***If**, **Then**, **Else**, och **End** finns alla i CTL-menyn i programeditorn. Heltals-funktionen (**Int** på engelska) när du genom att trycka på tangenten* math *och sedan välja menyn NUM. (förkortning för Numerisk). Denna funktion returnerar ett heltal när du matar in ett godtyckligt decimaltal. T.ex. heltal(6.56) → 6 heltal(9.999) → 9 heltal(-2.01) → -3.*



Pröva nu själv med några egna exempel där du använder heltalsfunktionen. Kom ihåg dubbla parentestecken för funktionen **Heltal( )** och kvadratrotsfunktionen. Du kan bara använda VERSALER i dina **Disp**-satser på räknaren. Med gratisprogramvaran **TI-Connect™ CE** kan du använda gemener och å, ä och ö.

Lärarkommentar: Föreslå för eleverna att de kan förbättra utmatningen vid körning av programmet genom att visa det inmatade talet också. Man ska t.ex. kunna få resultatet ”36 ÄR EN PERFEKT KVADRAT” istället för ”EN PERFEKT KVADRAT”. Då behöver man använda instruktionen Output( ) istället för Disp.

Algoritmer

Tekniker som används i denna aktivitet kallas för *algoritmer*. En algoritm är en procedur eller formel för problemlösning. Ett recept för att t.ex. baka en kaka är en slags algoritm. Om du följer receptet kommer du att stå där med en kaka så småningom.
Alla matematiska formler, t.ex. formeln för arean av en triangel, (A=B×H/2) är algoritmer: de ger dig en metod för att bestämma ett nytt värde baserat på existerande värden. Algoritmer, t.ex. som den med ”perfekt kvadrat” i denna aktivitet, är viktiga verktyg för problemlösning. Att lära sig ett antal vanliga ”datoralgoritmer” gör att du får ett bättre utbyte av din programmering.

Här ett kakrecept.
Sätt ugnen på 175°C.

1. Smöra och bröa en ugnssäker form, ca 20x30 cm.
2. Smält smöret. Vispa ägg och socker pösigt med elvisp.
3. Blanda alla torra ingredienser i en bunke.
4. Vispa ner smör och filmjölk i äggvispet. Vänd ner mjölblandningen. Häll smeten i formen.
5. Grädda i nedre delen av ugnen ca 40 minuter. Låt kakan svalna.
6. Vispa färskost och florsocker luftigt. Rör ner de tinade tranbären. Bred krämen över den kalla kakan.
7. Blanda de frysta tranbären med sockret. Strö över kakan. Skär i bitar.

Precis som i program följer bagaren alla steg från början till slut. Till slut så finns det en smakfull kaka att avnjuta. När en dator (eller räknare) följer stegen i ett program (algoritmen) uppnås det önskade resultatet. Det finns

faktiskt en specialitet i datavetenskap där man sysslar med att *bevisa* att en algoritm ger det önskade resultatet. Ungefär som när man bevisar matematiska satser.

Lärarkommentar: Här är ett exempel på en datoralgoritm:

För att avrunda ett tal till *närmaste* heltal kan man använda följande algoritm:

A = heltal(A+0.5)

Hur fungerar detta? När decimaldelen hos A är mindre än 0,5 så betyder addition av 0,5 till talet att talet fortfarande är mindre än nästa större heltal.

Funktionen heltal( ) trunkerar (hugger av) talet vid decimaltecknet och kvar blir heltalsdelen.

Här två exempel:

Anta att A=3.4 A+0.5 = 3.9 heltal(3.9) = 3 (avrundning nedåt)

Anta att A=3.7 A+0.5 = 4.2 heltal(4.2) = 4 (avrundning uppåt)

Nu har förståsTI-84 också en inbyggd funktion, avrund( ). Du hittar den om du trycker på [math] och väljer NUM-menyn.