

Kapitel 4 Loopar

Övning 2: While...End-loopen

I denna andra aktivitet för kapitel 4 ska du lära dig att använda **While...End**-loopen. Vi jämför den med **For...-loop** and och visar varför den är mer kraftfull och mångsidig än **For...-loop**.

While... End-Loopen

While...End-loopen kommer att fortsätta att gå i en slinga (loop=slinga) så länge som dess <villkor> är *sant*. Det ser ut så här:

```
While <villkor>
  <loopkropp>
End
```

Obs

<villkor> är ett logiskt uttryck, t.ex. $X > 0$.

<loopkroppen> är en uppsättning satser, som inkluderar andra loopar och **If**-strukturer. Den processas närhelst <villkor> är sant.

End används för att indikera slutet på <loopkroppen>. Vid **End**-satsen loopar programmet tillbaka till **While**-satsen och undersöker <villkor> igen.

Ge först ett begynnelsevillkor för **While**: ange ett värde så att villkoret är ordentligt fastställt som Sant eller Falskt. Om begynnelsevillkoret är Falskt så hoppar programmet över loopen. Om villkoret är Sant så processas loopkroppen. $0 \rightarrow K$ på första raden i programmet sätter begynnelsevillkoret till Falskt. Utan det så kan man inte veta vad som händer eftersom vilket värde som helst kan ha lagrats i variabeln **K** innan programmet körs.

Någonstans i <loopkroppen> ska det finnas en sats som påverkar <villkor> så att loopen eventuellt slutar och satserna efter loopen processas. Vanligtvis finns denna sats nära slutet av <loopkroppen>. $K+1 \rightarrow K$ ser till att **K** eventuellt blir större än 10.

Lärarkommentar: Det är viktigt att betona att While-loopen kanske inte processas alls. I nästa övning diskuterar vi **Repeat**-loopen som processas åtminstone en gång. Detta är en hårfin men viktig skillnad.

Det behövs tre komponenter för att bygga en lyckosam **While**-loop:

Initialvärden, Testa, och Ändra:

Initialvärden till en variabel.

Testa ett villkor baserat på den variabeln.

Ändra variabeln så att villkoret eventuellt blir falskt och loopen avslutas.

Syfte:

- Lära sig strukturen hos **While...End**-loopen.
- Jämföra den med **For...End** -loopen.
- Se hur den används för att säkerställa giltiga input-värden.

```
NORMAL FLYT AUTO REELL RAD MP
PROGRAM:WHILE
:0→K
:While K≤10
:Disp K
:End
:■
```

En oändlig loop. Varför?

```
NORMAL FLYT AUTO REELL RAD MP
PROGRAM:WHILE
:0→K
:While K≤10
:Disp K
:K+1→K
:End
:■
```

Vad åstadkommer detta program?

