Integralberäkning med programmering

I denna aktivitet som visar vi hur man med ett ganska enkelt program numeriskt kan beräkna areor under kurvor Vi använder här den s.k. mittpunktsmetoden som oftast ger mycket bra resultat.

Vi visar också hur man med några inbyggda verktyg hos TI-Nspire kan göra beräkningar av areor under kurvor.

På sid 2 förklarar vi For .. EndFor-satsen och formeln

$$m := m + f\left(a + d \cdot i + \frac{d}{2}\right) \quad \text{för } m.$$

Därefter kommer sedan en **For...EndFor**–sats som används för att processa en aritmetisk sekvens av värden. Det kallas för iteration. I detta fall ska vi beräkna värdet på *m* (som är arean under kurvan) med ett bestämt värde på ***d*** (rektanglarnas bredd). På sid 4 visar vi med ett enkelt exempel hur man approximativt beräknar arean under kurvan $f(x)=x^2$ från $x=0$ till $x=4$ som arean av fyra rektanglar med bredden (basen) 1. For–satsen **For *i*,0,*n*−1** och uttrycket $m+f\left(a+d\cdot i+\frac{d}{2}\right)\cdot d\to m$ gör att arean först beräknas för en rektangel, sedan för två, därefter för tre och till sist för alla fyra. Se nedan:

rektangel 1: $0+\left(0+1\cdot 0+\frac{1}{2}\right)^2 \blacktriangleright \frac{1}{4}$

rektangel 1+2: $\frac{1}{4}+\left(0+1\cdot 1+\frac{1}{2}\right)^2 \blacktriangleright \frac{5}{2}$ osv.

Man kan läsa mer For … EndFor och loopar i kapitel 4 Övning 1 bland ”**10 minutes of Code**”-övningarna.

Kapitel 4: Loopar **Övning 1: For-loopar**

I denna aktivitet kommer du att lära dig hur begreppet looping fungerar i programmering och närmare undersöka For-loopen.

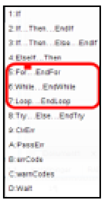
Syfte:
- Beskriva hur looping fungerar i programmering
- Konstruera program där man använder For...EndFor

Lärarkommentar: Det finns tre grundläggande loopar i TI-Nspire TI-Basic: For, While, och Loop. En loopstruktur ger ett program förmåga att processa en uppsättning av satser om och om igen, antingen upprepning över en sekvens av värden (precis som i For-loopen) eller tills ett speciellt villkor är uppfyllt (eller inte) som i While och Loop. Aktiviteterna i kapitel 4 introducerar var och en av dessa strukturer.
Program kan bli komplicerade eftersom det ofta är nödvändigt att blanda in alla kontrollstrukturer (If-satser och loopar) i ett program för att kunna arbeta med mer komplexa algoritmer. Det är ju detta som gör programmering så spännande och intressant … och roligt!

Om Loopar

Programmeringsspråket TI-Basic har förmågan att bearbeta en uppsättning programsatser om och om igen. Denna upprepning av uttalanden kallas **looping**.

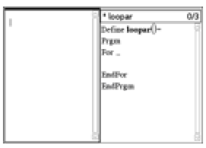
De tre loop-strukturer du lär dig i detta kapitel är nås genom att välja kontrollmenyn i programeditorn. Se skärmbild till höger. While… och Loop… strukturerna kommer att utforskas i senare aktiviteter i detta kapitel.



For...EndFor

For… loopen används för att processa en aritmetisk sekvens av värden. Det kallas för *iteration*.
Genom att välja For...EndFor-satsen från kontrollmenyn får du tillgång till de nödvändiga komponenterna som behövs för att bygga resten av strukturen:
For , , ,

EndFor



Kommatecknen efter ordet **For** indikerar att du behöver lägga till **fyra** poster:

For *i*, 1, *n*, 1

Så här ser programmet ut:

```
Define intberäk()=
Prgm
Local a,b,d,i,m,n,x,f
Request "Skriv in funktionen med avs. på x:",f(x)
Request "Nedre gräns:",a
Request "Övre gräns:",b
Request "Antal rektanglar:",n
m:=0
d:=(b-a)/n
Disp "ger intervallängd=",d
For i,0,n-1
m:=m+f(a+d·i+d/2)·d
EndFor
Disp "Med mittpunktsmetoden blir resultatet:"
Disp approx(m)
DelVar a,b,d,i,m,n,x,f
EndPrgm
```

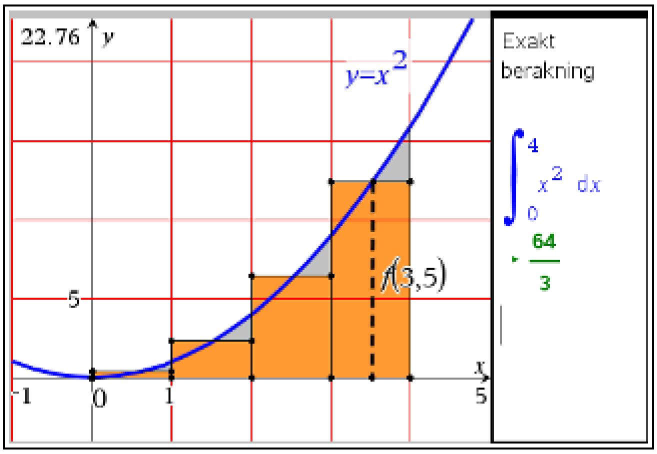
Vid programkörning för funktionen $f(x)=x^2$, nedre gräns 0, övre gräns 4 och 4 delintervall får vi följande resultat

```
intberäk()
Skriv in funktionen med avs. på x: x^2
Nedre gräns: 0
Övre gräns: 4
Antal rektanglar: 4
ger intervallängd= 1
Med mittpunktsmetoden blir resultatet:
21.
                                   Klar
```

Med 8 delintervall får vi följande resultat:

```
intberäk()
Skriv in funktionen med avs. på x: x^2
Nedre gräns: 0
Övre gräns: 4
Antal rektanglar: 8
ger intervallängd= 1/2
Med mittpunktsmetoden blir resultatet:
21.25
                                   Klar
```

Låt eleverna gärna pröva med många delintervall och jämför med det exakta resultatet.



På sid 6 och 7 visas hur man kan göra beräkningarna på annat sätt. Vi använder här funktionen **Seq** som är en förkortning av *Sequence* som här står för *talföljd*. Vi kan sedan summera de 4 termerna i talföljden med funktionen **sum.**

Define funk$(x)=x^2$

Vi beräknar nu den sammanlagda arean av rektanglarna med mittpunktsmetoden:

funk(0.5)·1+funk(1.5)·1+funk(2.5)·1+funk(3.5)·1 ▸ 21.

Vi får 21 areaenheter.

Med funktionen **seq** (förkortning på engelska för talföljd) kan vi få en *lista* med arean av varje rektangel

seq(funk(x),x,0.5,3.5,1)→höjd ▸ {0.25,2.25,6.25,12.25}

Syntaxen för talföljden är: (uttryck,variabel,startvärde,slutvärde,steg).

Vi har sedan sparat våra värden i variabeln *höjd*. Beräkning av den totala arean kan göra så här: sum(höjd) ▸ 21.

Nu gör vi en beräkning med fler intervall:

Med 100 delintervall får vi här ett mycket bra resultat.

Vi gör nu en beräkning med 100 intervall. Bredden på varje rektangel blir då 0,04 (4/100).

Mittpunkten i den första rektangeln blir då 0.02 och mittpunkten i den sista rektangeln blir då 4−0.02=3,98. På samma sätt som tidigare beräknar vi nu en lista med de olika rektangelareorna:

seq(funk(x),x,0.02,3.98,0.04)·0.04→höjd100

sum(höjd100) ▸ 21.3328

Vi får ett riktigt bra värde. Vi får exakt samma värde med programmet på sid 2. Det exakta värdet är $\frac{64}{3}$

På nästa sida har vi listan variabeln **höjd100** i ett kalkylark. Vi summerar sedan alla data i cell b1.

I nästa spalt visas nu hur man kan använda seq-funktionen i några enkla beräkningar.

$\mathrm{seq}(n^2,n,1,6)$ $\{1,4,9,16,25,36\}$

$\mathrm{seq}\left(\frac{1}{n},n,1,10,2\right)$ $\left\{1,\frac{1}{3},\frac{1}{5},\frac{1}{7},\frac{1}{9}\right\}$

$\mathrm{sum}\left(\mathrm{seq}\left(\frac{1}{n^2},n,1,10,1\right)\right)$ $\frac{1968329}{1270080}$

Obs: För att få ett närmevärde,

Handenhet: Tryck på ctrl enter.
Windows®: Tryck på **Ctrl+Enter.**
Macintosh®: Tryck på ⌘**+Enter.**
iPad®: Håll ned **enter** och välj ≈.

$\mathrm{sum}\left(\mathrm{seq}\left(\frac{1}{n^2},n,1,10,1\right)\right)$ 1.54977

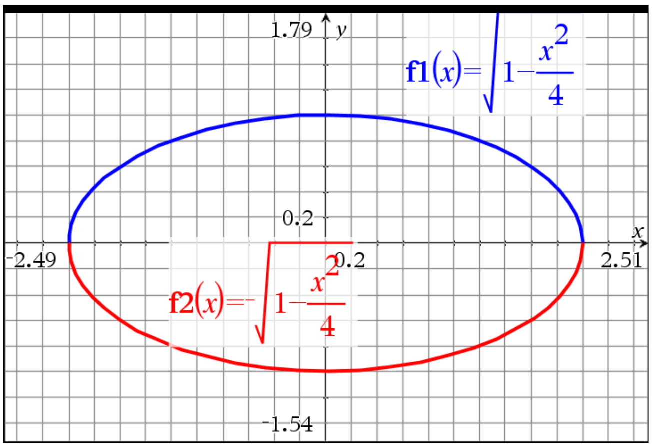
Här är arean av alla rektanglar i ett kalkylark. Vi har sedan summerat i cell b1.

	A höjd100	B	C	D	E
=					
1	0.000016	21.3328			
2	0.000144				
3	0.0004				
4	0.000784				
5	0.001296				
6	0.001936				
7	0.002704				
8	0.0036				

B1 =sum(höjd100)

I problem 2 gör vi sedan en areaberäkning på en halv ellips. Man kan plotta ellipsen med två funktioner

$y=+\sqrt{1-\frac{x^2}{4}}$ resp. $y=-\sqrt{1-\frac{x^2}{4}}$.



$\text{seq}(\mathbf{f1}(x),x,{}^{-}1.98,1.98,0.04)\cdot 0.04 \rightarrow \mathbf{ellips100}$

$\text{sum}(\mathbf{ellips100}) \blacktriangleright \mathbf{3.14257}$

$\text{seq}(\mathbf{f1}(x),x,{}^{-}1.998,1.998,0.004)\cdot 0.004 \rightarrow \mathbf{ellips1000}$

$\text{sum}(\mathbf{ellips1000}) \blacktriangleright \mathbf{3.14162}$

$\text{approx}(\pi) \blacktriangleright \mathbf{3.14159}$ Exakt beräkning: $\displaystyle\int_{-2}^{2} \mathbf{f1}(x)\, dx \blacktriangleright \boldsymbol{\pi}$

Avslutningsvis skulle man kunna säga att numeriska metoder för beräkning av integraler understryker integralens karaktär som summa. Det är lätt att eleverna tappar bort detta när de bara tar fram exakta svar med hjälp av primitiva funktioner.