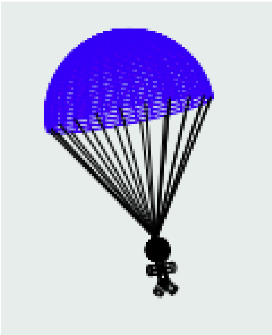
# Differentialekvationer – när man inte kan lösa exakt



Hastigheten hos en fallskärmshoppare som faller kan modelleras med en differentialekvation som inte är linjär. Vi får då ta till en numerisk lösningsmetod.

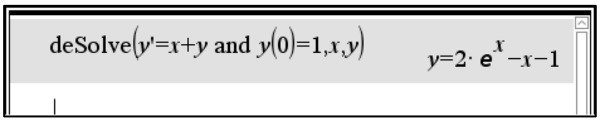
I ämnesplanerna i matematik betonas att man ska få möjlighet att använda digitala verktyg. Ett exempel som passar för kursen Matematik 5 är numeriska lösningar av differentialekvationer. Vi visar här hur enkelt man kan arbeta med de verktyg som finns hos din grafräknare.

Eulers metod är ett grafiskt/numeriskt verktyg hos för att visualisera en *approximativ* lösning till differentialekvationer. Eulers metod motiveras av idén om "lokal linjäritet" — alltså att en deriverbar funktion beter sig som en linjär funktion i små intervall. Med denna idé, om du känner till värdet av derivatan av en funktion i en enda punkt, så kan du approximera en liten del av dess graf med ett linjesegment med den efterfrågade lutningen i den punkten. Låter det krångligt? När du arbetat igenom ett exempel kommer du att förstå bättre.

Om du har en differentialekvation och ett begynnelsevillkor har du den information som behövs för att approximera en liten del av grafen. Om ($x_0$, $y_0$) är begynnelsevillkoret kan man approximera ett *y*-värde $y_1$ på lösningskurvan som mot-svarar $x_1 = x0 + \Delta x$. Nu kan du sedan upprepa denna strategi genom att behandla ($x_1$, $y_1$) som en ny punkt på lösningskurvan. Man upprepar sedan denna process om och om igen.
Vi visar här ett exempel där vi grafiskt, numeriskt löser differentialekvationen

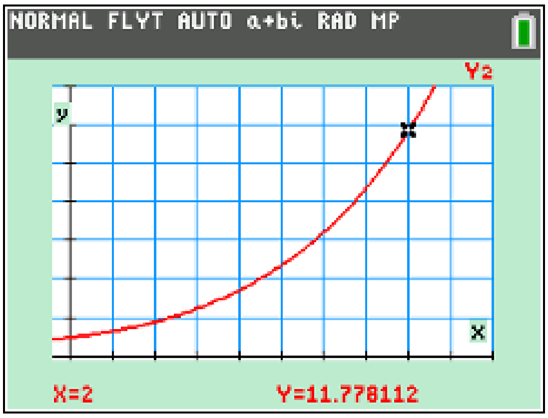
$y' = x + y$

med begynnelsevillkoret $y(0) = 1$. Vi vill beräkna värdet för $x = 2$. Den *exakta* lösningen är $2e^x - x - 1$ och $f(2) \approx 11{,}78$. Vi börjar med steglängden 0,5. Här visas en bild från programmet TI-Nspire-CX.




Men varför ska man nu lösa differentialekvationer numeriskt om det går att göra exakt. Svaret är enkelt. De flesta differentialekvationer som man ställer upp för att modellera verkliga förlopp går ***inte*** att lösa med exakta metoder. Därför är det viktigt att det finns numeriska verktyg och att man kan använda dem. Eulers metod har en enkel algoritm som är lätt att förstå. Det finns andra kraftfullare metoder men där är algoritmerna mer komplicerade.

Så här ser grafen till den exakta lösningen ut:




Vi visar nu här i steg hur metoden fungerar för ekvationen $y' = x + y$.

*Första steget:*
I startpunkten (0, 1) väljer vi den riktning som ges av tangenten till lösningskurvan som går genom (0, 1). Detta ger att riktningen blir

$y' = 0 + 1 = 1.$

Med steglängden 0,5 kan vi då teckna ekvationen för riktningskoefficienten

$\dfrac{y_1 - 1}{0{,}5 - 0} = 1$ som ger att $y_1 = 1{,}5$.

*Andra steget*:

Nu utgår vi från punkten (0,5, 1,5). Vi får nu

$y' = 0{,}5 + 1{,}5 = 2.$ Vi tecknar ekvationen för riktningskoefficienten igen

$\dfrac{y_2 - 1{,}5}{1 - 0{,}5} = 2$ som ger att $y_2 = 2{,}5.$

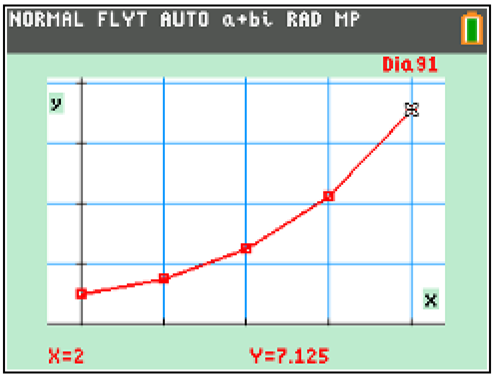
*Tredje steget*:

Nu utgår vi från punkten (1, 2,5). Vi får nu

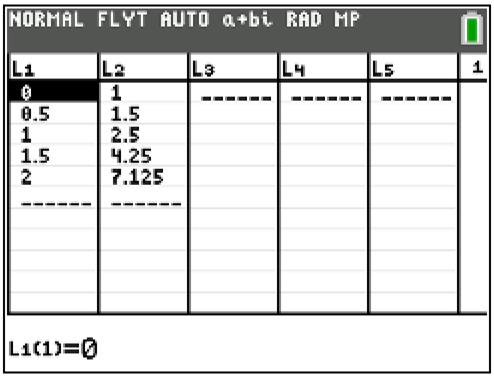
$y' = 01 + 2{,}5 = 3{,}5.$ Vi tecknar ekvationen för riktningskoefficienten igen

$\dfrac{y_3 - 2{,}5}{1{,}5 - 1} = 3{,}5$ som ger att $y_3 = 4{,}25.$

På samma sätt kan vi nu ta ett steg till med steg-längden 0,5 och då får vi värdet 7,125 för x = 2. Vi får en kurva sammansatt av sträckor – en *polygon*. Se figur.




I statistikeditorn ser det ut så här:




Vi behöver nog öka antalet steg för att få ett bättre resultat. I stället för 4 steg så försöker vi nu med 40 steg. Detta vill vi ju inte göra för hand Man kan då skriva ett enkelt program i TI-Basic till exempel som numeriskt löser D.E. med start-värden för *x* och *y*. x-värden och beräknade *y*-värden läggs i listor i räknarens statistikeditor.

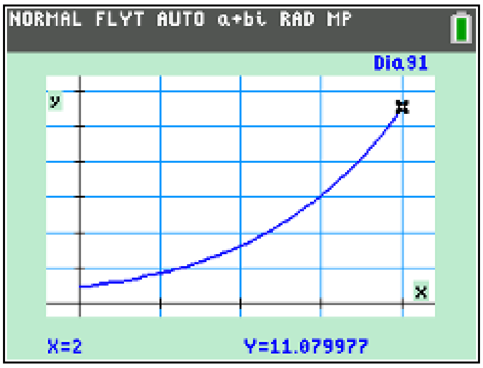
Du kan spåra i lösningskurvan och grafiskt se alla beräknade värden. Du kan ha högst 250 steg och kurvan plottas som ett linjediagram. Innan du kör programmet så matar du in D.E. i funktions-editorn vid Y1. Sedan avmarkera du ekvationen genom att placera markören vid likhetstecknet och trycker på $\boxed{\text{enter}}$.


Här är nu programmet. Det heter EULER1. Du kan nu ner det till din räknare via gratisprogrammet TI Connect CE.

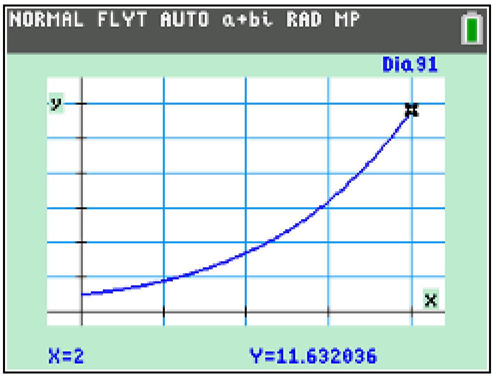
```
RensaLista L₁,L₂
Disp "STARTVÄRDE X":Input X
Disp "STARTVÄRDE Y":Input Y
Disp "SLUTVÄRDE X":Input B
Disp "ANTAL STEG":Input N:
(B-X)/N→H
1→M
While X≤B och M≤250
X→L₁(M):Y→L₂(M):X+H→S
Y+HY₁→T
S→X:T→Y
M+1→M
End
ZoomStat
Diag1(xyLinje,L₁,L₂,•)
```

Så här ser inmatningarna ut när du beräkningarna börjar.


Efter programkörning ser det ut så här på skärmen. Vi har spårat i diagrammet och ser att det beräknade värdet för x=2 blir 11,08. Se nästa sida.




Vi ökar nu antalet steg till 200.!
Nu får vi ett bra mycket bättre värde.



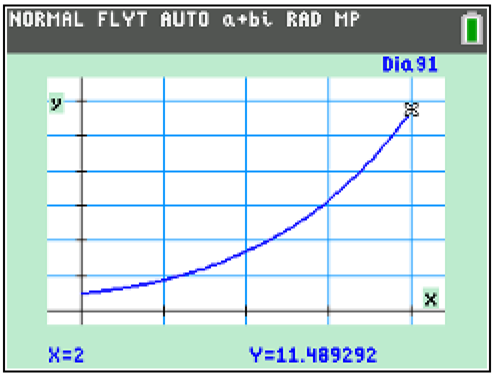

Det korrekt närmevärdemed tre decimaler är 11,778. Detta ger att felet vid 200 steg blir

11,778 - 11,632= 0,146.

Detta kallas för trunkeringsfel. Nu finns en regel som med formelspråk lyder:

$y(\text{sanna värdet}) \approx 2 \cdot y_{2N} - y_n$

*N* står här för antalet steg. Vi beräknade ju värdet för 200 steg till 11,632. Om vi nu gör en beräkning för 100 steg får vi:




Formeln ovan ger nu

$y(2) \approx 2 \cdot 11{,}632 - 11{,}489 = 11{,}775$

Ett utomordentligt bra värde!

Finns det andra sätt än att skriva ett program som fungerar för sådana här upprepade beräkningar? Kalkylprogram passar väldigt bra på att göra sådana här upprepade (iterativa) beräkningar. Vi tar upp detta i aktiviteter som handlar om rekursiva talföljder.

Nedan ser du beräkningarna i ***Excel***. Start-värdena har du i A1 och B1. Derivatavärdena finns i kolumn C.

| | A | B | C |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 2 | 0.5 | 1.5 | 2 |
| 3 | 1 | 2.5 | 3.5 |
| 4 | 1.5 | 4.25 | 5.75 |
| 5 | 2 | 7.125 | 9.125 |

Här ser du formlerna som ligger bakom de beräknade värdena.

| | A | B | C |
|---|---|---|---|
| 1 | 0 | 1 | =A1+B1 |
| 2 | =A1+0.5 | =0.5*C1+B1 | =A2+B2 |
| 3 | =A2+0.5 | =0.5*C2+B2 | =A3+B3 |
| 4 | =A3+0.5 | =0.5*C3+B3 | =A4+B4 |
| 5 | =A4+0.5 | =0.5*C4+B4 | =A5+B5 |

Så här ser det ut i programmet TI-Nspire och dess app Listor och kalkylblad. Samma formler som i Excel-arket.

| | A x | B y | C der_y | D | E |
|---|---|---|---|---|---|
| = | | | | | |
| 1 | 0 | 1 | 1 | | |
| 2 | 0.5 | 1.5 | 2. | | |
| 3 | 1. | 2.5 | 3.5 | | |
| 4 | 1.5 | 4.25 | 5.75 | | |
| 5 | 2. | 7.125 | 9.125 | | |
| 6 | | | | | |

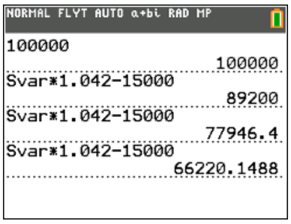
B5 $=0.5\cdot c4+b4$

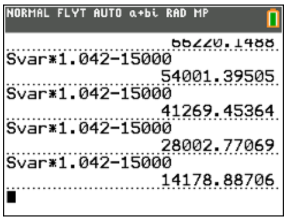
Det finns faktiskt ytterligare ett sätt att göra sådana här beräkningar direkt med räknaren utan att använda ett program. Se nästa sida

Innan vi börjar med detta tredje sätt att numeriskt lösa differentialekvationen så tar vi upp begreppet *rekursion* och speciellt rekursiva talföljder. Du har antagligen redan studerat detta i kurs 5 men här kommer en kort repetition:

Vi börjar med ett enkelt exempel där du utför dina beräkningar i grundfönstret. Tänk dig att någon har ett billån på 100 000 kr som ska avbetalas med 15 000 kr varje år. Räntan är 4,2 %.

Då skriver du först in startbeloppet och trycker på [enter]. Sedan skriver man ∗1.042-15000 och trycker på [enter] igen. Då får vi skulden efter ett år när vi amorterat 15 000 kr.
Vi fortsätter att trycka på [enter] och får då skulden efter 2 år, 3 år osv. Efter 7 år är skulden ca 14 000 kr.

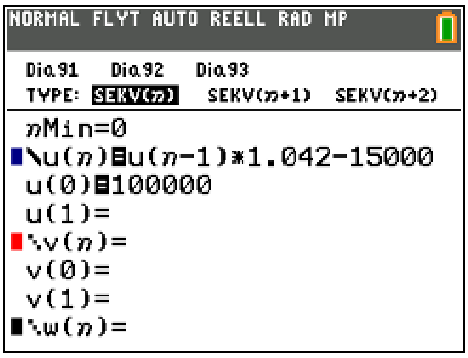



Detta är exempel på en s.k. rekursiv talföljd. En talföljd är rekursiv om nästa tal i talföljden följer från tidigare tal enligt en bestämd regel. Tal som behövs för att sätta i gång följden kallas start-värden. Här har vi startvärdet 100 000 och regeln är "multiplicera med 1,042 och dra sedan av 15 000".

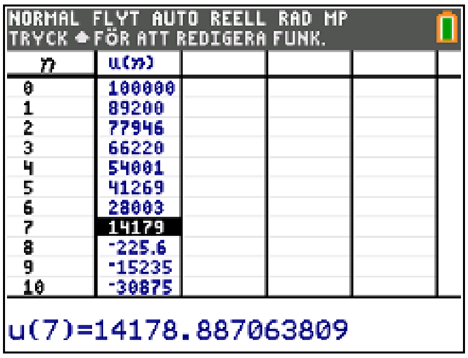
Vi går nu igenom en speciell inställning man kan göra på räknaren, när man vill beräkna termer i talföljder. Vi använder samma exempel som ovan. Tryck på [mode]. På fjärde raden, som handlar om inställningar vid grafritning, ställer du in läget **SEKV**. SEKV står för det engelska ordet Sequence, som betyder sekvens eller i matematiksammanhang talföljd.


Tryck nu på knappen [y=]. Nu ska vi skriva in en formel för vår upprepade beräkning. Vi startar vårt räkneverk vid 0 genom att skriva *n*MIN=0. Gå sedan till nästa rad och skriv in enligt skärmbilden. **u** når du genom att trycka [2nd] [7]. Det står ju ett litet **u** i gul stil ovanför knappen med siffran 7. ***n*** når du genom att trycka på knappen [X,T,θ,n].

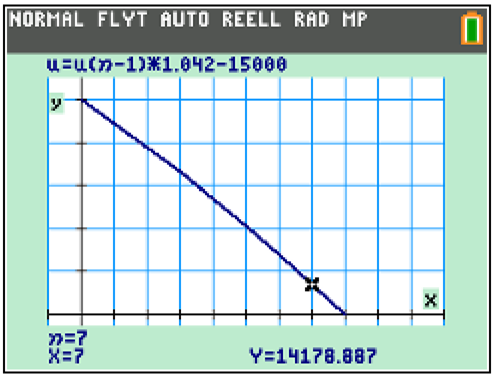



Vi har nu en formel i s.k. *rekursiv* form. Vi får nästa term genom att utgå från den föregående.

Tryck nu på [2nd] [table]. Då du en tabell på skärmen. Den visar de första termerna i talföljden.



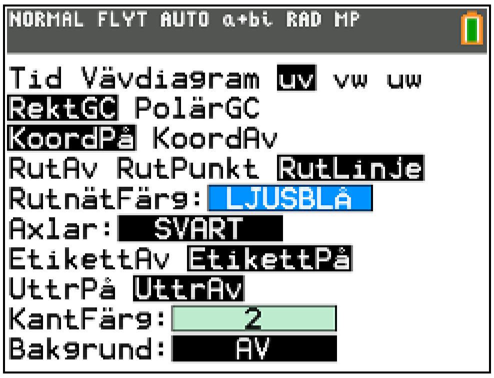

Du kan också plotta. Man får då se till att man har en bra inställning på sitt fönster.



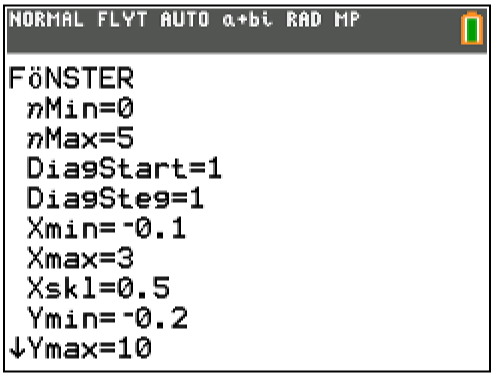

Efter denna genomgång går vi nu direkt över till differentialekvationen

$y' = x + y$

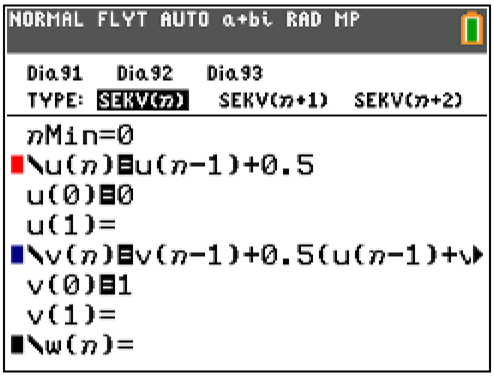
Vi kommer i inmatningsfälten att ha två olika talföljder där den första ger värden på x och den andra värden på y. I stället för x och y har vi nu u och v.
Tryck först på [2nd] [format] för att ställa in rätt format för plottningen. Överst så ställer du in formatet för talföljderna som uv. Det betyder att du ska plotta **u** längs x-axeln och **v** längs y-axeln.




Ett bra fönster ser du här. Vi ska ju bara plotta fram till x= 2 och med steglängden 0,5.




Nu kommer vi till det intressanta, nämligen inmatningen av våra uttryck för talföljden.

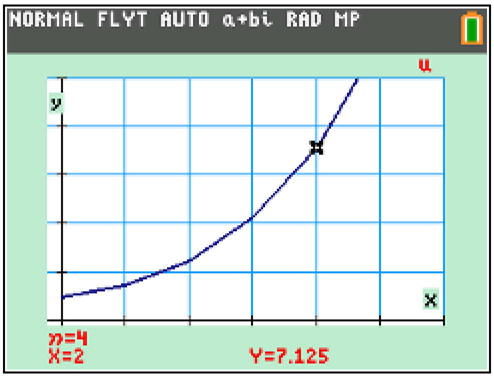



Uttrycket för *v* får inte plats i fönstret men där står nu

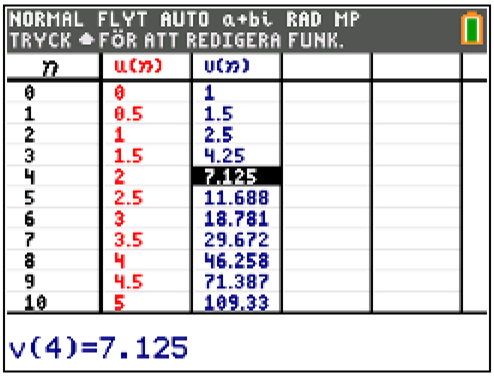
$v(n) = v(n-1) + 0.5(u(n-1) + v(n-1)$

Vi ser att *u*(*n*-1), som visar x-värdena förekommer i uttrycket för *u*(*n*). Man kan säga att talföljden v(*n*) är kopplad till talföljden *u*(n)
Startvärdet (0, 1) ser vi som u(0) och *v*(0).

Nu plottar vi. Tryck på [graph]




[2nd][table] ger en värdetabell:


NORMAL FLYT AUTO a+bi RAD MP
TRYCK ◆ FÖR ATT REDIGERA FUNK.

| n | u(n) | v(n) |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0.5 | 1.5 |
| 2 | 1 | 2.5 |
| 3 | 1.5 | 4.25 |
| 4 | 2 | 7.125 |
| 5 | 2.5 | 11.688 |
| 6 | 3 | 18.781 |
| 7 | 3.5 | 29.672 |
| 8 | 4 | 46.258 |
| 9 | 4.5 | 71.387 |
| 10 | 5 | 109.33 |

v(4)=7.125

Vi avslutar denna aktivitet med ett exempel från verkligheten. Se nästa sida.

För en fallskärmshoppare som faller får man en bra matematisk modell om man antar att luft-motståndet är proportionellt mot *hastigheten i kvadrat*. Modellen kan då skrivas så här

$$\frac{dv}{dt} = g - \frac{k}{m} \cdot v^2$$

*Detta är en ekvation som man inte kan lösa med elementära metoder. Nu måste vi ta till en numerisk metod.*

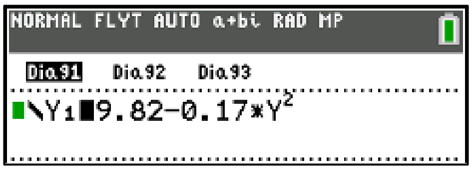
*g* är tyngdaccelerationen 9,82 m/s$^2$, *k* är en proportionalitetskonstant som kan sättas till 15 kg/m och *m* är massan av hopparen + fall-skärmen, som kan sättas till 90 kg. När skärmen utvecklas har hopparen farten 35 m/s. Alltså är *v*(0)=30.

Vi låter nu *v*(*t*) beteckna hopparens hastighet *t* sekunder efter att skärmen utvecklats. Vi vill beräkna hopparens hastighet efter 2 sekunder.

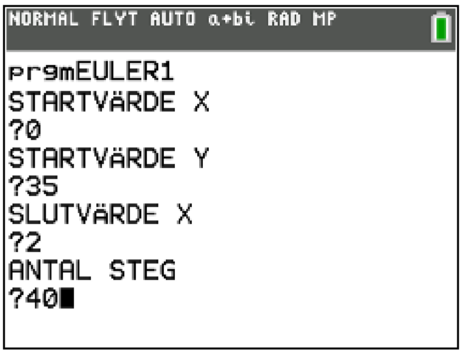
Efter insättning av konstanterna får vi följande uttryck

$$\frac{dv}{dt} = 9{,}82 - 0{,}17 \cdot v^2$$

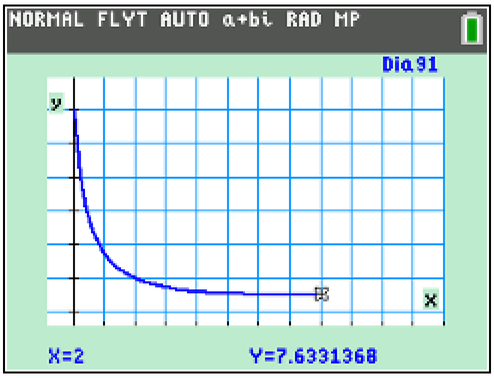
I Y1 skriver du då in enligt skärmbilden nedan. Glöm inte att *avmarkera* Y1 genom att placera markören vid Y1 och trycka på [enter].




Start nu programmet Euler1 och välj värden så här:

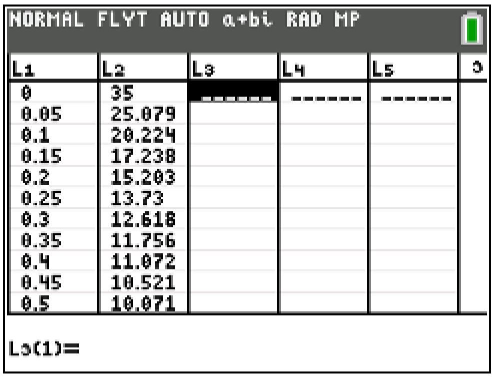



Nu kan vi plotta lösningen. Se nästa spalt.



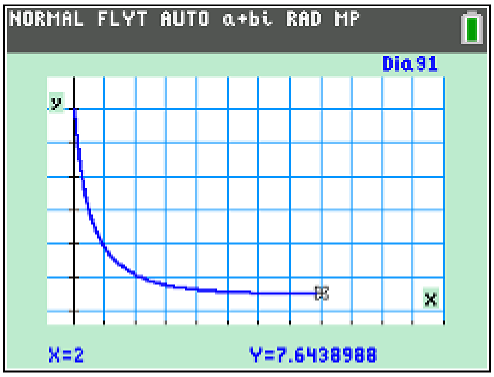

Vi ser att hastigheten blir ca 7,6 m/s.

I statistikeditorn ser det ut så här:




Vi har alltså 0,05 sekunder mellan varje värde. Vi satte ju antalet steg till 40.

Vi prövar nu med 80 steg. Vi får nästan samma värde på v(2) som förut.




Vi ser att kurvan planar ut och för att beräkna den hastighet hopparen får när tiden går mot oändligheten löser man ekvationen

$$0 = 9{,}82 - 0.17v^2$$

($\frac{dv}{dt}$, dvs accelerationen är ju 0 då)

Vi får värdet $v = \sqrt{\frac{9.82}{0.17}} \approx 7{,}60$ m/s.