Contents

1 W1 Gymnasiematte: påminnelse om saker vi behöver 1
1.1 Reella tal 2
1.1.1 Som oändliga sekvenser av siffror i decimal form 2
1.1.2 Varför skall en IT ingenjör veta om struktur och teori av reella tal? 4
1.1.3 Varför skall en IT ingenjör veta om operationer med reella tal? 4
1.1.4 Tallinje och det Kartesiska koordinatsystemet 5
1.1.5 Ekvationer/olikheter med absolut belopp 6
1.1.6 Komplexa tal 7
1.2 Enkla och komplicerade funktioner 14
1.2.1 Allmänt om funktioner 15
1.2.2 Olika sätt att beskriva kombinationer av funktioner 16
1.2.3 Implicita funktioner med kända eller okända namn 19
1.2.4 Trigonometriska funktioner 21
1.2.5 Hyperboliska funktioner 22
1.2.6 Algebraiska likheter mellan trigonometriska och hyperboliska funktioner 23
1.3 Ytterligare likheter mellan trigonometriska och hyperboliska funktioner 24
1.3.1 Derivata av enklare funktioner och deras kombinationer vi behöver 25
1.3.2 Ett typiskt misstag studenter gör med den enklaste kedjeregeln 26
1.3.3 Varifrån kommer den enklaste kedjeregeln? 27
1.3.4 Trigonometriska och hyperboliska funktioner kommer från liknande DE 27
1.3.5 Differentiella ekvationer intro 29
1.3.6 Den primitiva funktionen och obestämd integral 29

1 W1 Gymnasiematte: påminnelse om saker vi behöver

Mängd är ett av de viktigaste koncepten inom matematik och en viktig byggsten för allt som följer i kursen och senare i yrkeslivet. Några exempel: det är ganska svårt att fundera vad funktion betyder om man inte kan idéerna bakom olika tal mängder; det är svårt att skriva ett objektorienterat program om man inte behärskar mängder och operationer med mängder. Vi börjar kursen med att diskutera några typer av mängder: talmängder, mängder som en del av tallinje R (1D[1]), mängder som en del av $R \times R$ (2D[2]). I kontexten av matematisk analys är den viktigaste mängden vi kommer att diskutera omgivningen av en punkt och olika sätt att beskriva den i 1D (intervall). Senare när ni

[1] 1D = en dimension; motsvarar R
[2] 2D = två dimensioner; motsvarar $R \times R$

börjar med flervariabelanalys skall ni möjligtvis vara intresserade av omgivningar i högre dimensioner: 2D (disk) och 3D (en boll). Men i envariabelanalys nöjer vi oss med 1D.

1.1 Reella tal

1.1.1 Som oändliga sekvenser av siffror i decimal form

Vi definierar ett reellt tal r som en oändlig talföljd i form

$$r = p \pm 0.x_1x_2x_3 \cdots x_{n-1}x_nx_{n+1} \cdots$$

där p är ett heltal $Z = \{0, \pm 1, \pm 2, \cdots\}$ och x_1, x_2, osv är siffror (0, 1, 2, …, 9). Alltså representerar vi reella tal i formen

$$r = p \pm q$$

För mera detaljer se P.1."Real Numbers and real Line", alldeles i början.

Siffror i q kan ha en struktur, där siffror upprepas, eller kan upplevas som någonting slumpmässig. Hur som helst, samlar vi alla sådana tal i en mängd R. Alla andra mängder ni är vana vid får man genom att begränsa val av siffror (se P.1, sidan 2, "the set of real numbers has some important subsets").

Kan ni se hur man skall begränsa siffror i r för att få tal mängder ni känner till?

- $N = \{0,1,2,3, \cdots\}$
- $Z = \{0, \pm 1, \pm 2, \pm 3, \cdots\}$
- $Q = \left\{\frac{n}{m} \mid n, m \in N, m \neq 0\right\}$
- $I = R \backslash Q$

Det finns en speciell notation i ADAMS för upprepande siffror (se P.1, sida 2, "the bar indicates the pattern of repeating digits"), tex

$$0.123123123 \cdots \equiv 0.\underline{123}$$

Notera att i vardagsbruk av matematiken skriver vi 0.123 men vi vet att det finns en oändlig följd av nollor efter 3:an,

$$0.123 = 0.123000 \cdots \equiv 0.123\underline{0}$$

Så, teoretiskt sett, finns det sekvenser som är snällare och andra som är mer komplicerade. De enklaste sekvenser är av typ 0.123 och sen en oändlig följd av nollor. Nästa i skalan av svårighetsgrad är sekvenser som inte är nollor, är oändliga, men upprepar sig. Dessa sekvenser är oändliga men har en struktur. Den tredje typen av sekvenser är de som ser ganska oregelbundna ut, utan någon som helst upprepning, men det finns en algoritm (oracle) som kan "spotta ut" siffror. Alltså, r förekommer i två former:

- Upprepande siffror (rationella tal Q)
- Icke upprepande siffror (irrationella tal $\underline{Q}$)

Ett superviktigt budskap från ADAMS är att r med upprepande siffror alltid kan skrivas i formen n/m där n och m är hela tal. Se exempel P.1.EX1 för hur man konstruerar n och m från r. Vad menas med P.1.EX1? Man skall gå till kapitel P.1 i ADAMS

P.1 Real Numbers and the Real Line

och där finns EXEMPEL 1 som refereras till

EXAMPLE 1 Show that each of the numbers (a) $1.323232\cdots = 1.\overline{32}$ and (b) $0.3405405405\ldots = 0.3\overline{405}$ is a rational number by expressing it as a quotient of two integers.

För att träna ytterligare på konceptet se problemen P.1.3 och P.1.4. Här och framöver alla anvisningar till ADAMS kommer att bli i grönt. Vad betyder P.1.3 och P.1.4? Det betyder att man skall gå till kapitel P.1 i ADAMS, och sedan i varje kapitel det finns en lista med tillhörande problem. För kapitlet P.1. övningar finns under

EXERCISES P.1

och där hittar ni problemen som refereras till.

In Exercises 3–4, express the given repeating decimal as a quotient of integers in lowest terms.

3. $0.\overline{12}$ **4.** $3.2\overline{7}$

Vi människor kan inte begripa det oändliga. Vi måste föreställa oss en process där vi närmar oss oändligheten på något sätt. Det reella talet r är det yttersta resultatet av en process där vi behåller mera och mera siffror

$$r = \lim_{n\to\infty} r_n$$

Om ni tror ni fattar, då får jag lov att utmana er med en bra funderare P.1.6.

När vi skriver som ovan, tänker vi oss en maskin som kan räkna ut alla siffror för varje angivet tal r. Maskinen använder en vis algoritm som vi inte behöver känna till. En viktig antagande är att vi kan faktisk förklara med ändlig mäng information vad talet vi är intresserade i är. I datavetenskap sådana maskiner som mirakulöst gör jobbet åt oss ofta brukar heta ”oracle”. Vill vi veta r med större noggrannhet, då väntar vi bara längre tills oraklet ”spottar ut” flera siffror tills vi är nöjda. Och här antar vi att vi har allt tid i världen att vänta att siffrorna levereras.

Fråga: om oraklet gav oss 6 siffror, vad är precision av det reella talet vi just fick? Hur utrycker man noggrannhet av ett tal? Svar: Vi avvaktar med svaret på fråga tills vi har diskuterat olikheter med absoluta belopp, men vad tror ni intuitivt? Frågan är relaterad till problem P.1.6

1.1.2 Varför skall en IT ingenjör veta om struktur och teori av reella tal?

I praktiska tillämningar vi alltid kapar r så att vi behåller ett ändligt antal siffror:

$$r_n = p \pm 0.x_1x_2x_3 \cdots x_{n-1}x_n$$

Vi vet att resterande siffror är 0. Om man använder den speciella notationen som Adams använder för upprepande siffror:

$$r_n = p \pm 0.x_1x_2x_3 \cdots x_{n-1}x_n\underline{0}$$

Faktum att vi jobbar med ändliga sekvenser har visa konsekvenser som man måste vara medveten om.

1.1.3 Varför skall en IT ingenjör veta om operationer med reella tal?

anledning 1: att klara duggor

anledning 2: att skriva säker kod

anledning 3: att tillämpa operationer med reella tal på ätt korrekt sätt.

En funderare: Varför tror ni att koden som har linjer som nedan kommer inte att fungera?

- `if(x^2=11) then stop`
- `a=float(1)/3; if ( 3*a == 1) then stop`

Alternativ sätt att skriva kod ovan är

- `if(x^2=11) then {en viktig kod}`
- `a=float(1)/3;`
  `if ( 3*a == 1) then {en viktig kod}`

eller att göra saken ganska konkret

- ```
  if(x^2=11)then
     {
       version 1: alla i klassen inklusive
       läraren får 1 MSEK
     }
  ```
- ```
  a=float(1)/3;
  if ( 3*a == 1) then
     {
        version 2: alla i klassen inklusive
        läraren får 100 SEK
     }
  ```

Det är kanske inte så svårt att se att version 1 kommer aldrig att hända. Vi kommer aldrig att hitta ett tal sådan att $x^2 = 11$ för vi vet att $\sqrt{11}$ är irrationell. Tyvärr, inga fria pengar från riksministeriet för utbildning.

Men, varför skulle version 2 vara omöjlig? En 100 lapp för kaffe och bulle borde vi väl kunna få från Matilda Ernkrans: Trotts allt som duktiga programmerare har vi definierat a som ett reellt tal. Visa programmeringsspråk skulle tolka 1/3 som ”integer division” och utföra (i maskin kod) a=0 men vi var försiktiga med det. Allt borde vara ok. Men, testa i vilket programmeringsspråk ni vill (utom de symboliska som Mathematica eller Maple) och ni kommer att se att det inte funkar.

Det är också superviktig att förstå när man skall fundera att använda extra parenteser i uttrycket, och ana hur kompilatorn kan tolka koden. Några exempel på att skriva farlig kod och varför är det viktig att använda parenteser när ni skriver kod:

```
program farlig_kod
   a = 2
   b = 3
   c = a/b*c // är det (a/b)*c eller a/(b*c)?
   d = a^b*c // är det (a^b)*c eller a^(b*c)?
   e = a*b/c*d // är det (a*b)/(c*d), eller a*(b/c)*d, eller …?
end program
```

Varje kompilator har egna regler hur man tolkar sådana uttryck och hur olika operationer skall prioriteras. Tex i `a^b*c` de flesta kompilatorer skulle ordna så att potenser räknas först och multiplikation senare som i `(a^b)*c` men det kan man inte ta för givet. Problemet med en kod som ovan som saknar parenteser är att den inte omvandlas till samma maskin kod. Vad programmet gör kan ändras beroende på vilken maskin man kör den, eller ännu värre, på vilken kompilatorn vi har använt!

Ungefär samma sak gäller med matematiska uttryck. Vi vill vara extra tydliga när det väl behövs. I matematiken om det står inga parenteser då menar vi att multiplikationer görs först, före additioner och subtraktioner. Vi också antar att potenser räknas först och allt annat efteråt.

Viktiga algebrakunskaper:

- kvadrerat regel: $(a+b)^2 = a^2 + 2ab + b^2$
- konjugat regel: $(a-b)(a+b) = a^2 - b^2$

1.1.4 Tallinje och det Kartesiska koordinatsystemet

Tallinje: intervallnotation

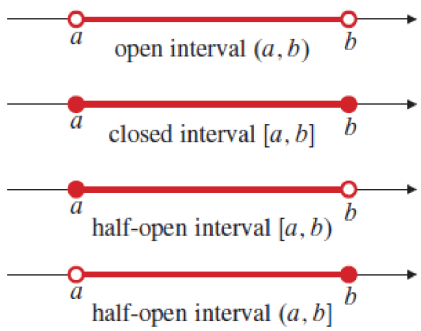


Figure P.2 Finite intervals

Exempel 1: Låt $A = [1,2]$ och $B = [2,3]$. Alltså, A är alla tal mellan 1 och 2 inklusive kanter och B är alla tal mellan 2 och 3 inklusive kanter. Ni känner till de vanliga mäng operationer från gymnasiet, union, tvärsnitt, osv. Vad är $A \cap B$ och $A \cup B$ lika med?

Det Kartesiska koordinatsystemet:

- linje med känd riktningskoefficient som går igenom en punkt; för att fräscha upp se P.2.15-16

- linje som är vinkelrätt mot en annan linje: för att fräscha upp se P.2.31-32
- tvärsnitt av olika objekt i 2D och relation med ekvationssystem: Under gymnasiet tid ni har jobbat med ekvation system osv. I princip, vad man gör när man löser ett sådant ekvationssystem är att man räknar tvärsnitt mellan två mängder. Med varje ekvation vi associerar en mängd med punkter som är lösningar i det Cartesiska koordinatsystemet.

1.1.5 Ekvationer/olikheter med absolut belopp

Absolut belopp definitionen:

The Absolute Value

The **absolute value**, or **magnitude**, of a number x, denoted $|x|$ (read "the absolute value of x"), is defined by the formula

$$|x| = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases}$$

The vertical lines in the symbol $|x|$ are called **absolute value bars**.

EXAMPLE 6 $|3| = 3, \quad |0| = 0, \quad |-5| = 5.$

1D: kanter av en intervall; 2D: cirkel:

Enklast att förklara är med några **exempel**. Alltså, vad föreställer ekvationer?

1) $|x| = 3$ (P.1.27)
2) $|x - 3| = 7$ (P.1.28)
3) $|x - 3| = 0.5$
4) $|x - 3| = 0.1$
5) $|x - 3| = 0.01$

I exemplen ovan är det viktigaste att föreställa sig lösningar som mängder i tallinje. Alla mängder ovan är väldigt speciella för de innehåller två punkter. Tex lösningen till P.1.27 är en mängd i form $\{x_1, x_2\}$ och vi vet från definitionen av absolut belopp funktionen att det måste vara så att $x_1 = 3$ och $x_2 = -3$. I text, vi skulle utrycka det så här: Lösningar till ekvationen $|x| = 3$ är $\{+3, -3\}$ eller så skriver vi ibland $x = \pm 3$, eller $x_1 = +3$, $x_2 = -3$. Just nu favoriserar vi mängdnotation.

Omgivningen av en punkt i 1D, 2D, och 3D

Ytterligare **exempel**: Man kan försöka visualisera ett enkelt intervall som en 1D-circkel med origo som är i mitten av intervallet. Alla lösningar till ekvationer nedanför är sådana 1D-cirklar.

6) $|x| < 2$ (P.1.33); Tips[3]
7) $|s - 1| \leq 2$ (P.1.35); Tips[4]
8) $|s - 1| \leq 0.1$; Tips[5]
9) $|s - 1| \leq 0.01$; Tips[6]

Tillämpningar:

[3] Mängden är 1D-cirkel med centrum i 0 med radie 2.
[4] Mängden är 1D-cirkel med centrum i 1 med radie 2.
[5] Mängden är 1D-cirkel med centrum i 1 med radie 0.1.
[6] Mängden är 1D-cirkel med centrum i 1 med radie 0.01.

- behövs för att beskriva felet när vi jobbar med approximationer
- behövs för att definiera kontinuitet
- behövs för att definiera gränsvärdet

Exempel 2: Låt $A = [1,2]$ och $B = (2,4)$. Ett alternativt sätt att beskriva den första mängden är $A = \left\{x; \left|x - \frac{1}{2}\right| \leq \frac{1}{2}\right\}$. Alltså vi föreställer oss *A* som an omgivning kring ½ med radie ½. Vad är det alternativa sättet att beskriva *B*? Tips: tänk omgivningen av *3* med radie *1*. Notera att kanter inte är med för *B*.

Exempel 3: Alltså om vi vill betrakta $[a, b]$ som en 1D-cirkel. Vad är koordinaten av cirkelns centrum, x_c, och radien, r? Kolla att $x_c + r = b$ och att $x_c - r = a$. Varför hävdar jag att ni skall kolla att det är så? Varför villa jag att det är så? Tips[7]

Nu när man kan absolut belopp, fundera man kan fixa den problematiska koden vi har diskuterat tidigare. Här har ni ett exempel vi vilket absolutbelopp funktionen kan användas för att fixa kod som kraschar:

- `if(`$|x^2 - 11| < 0.0001$`) then {en viktig kod}`
  Strategin att undvika numeriska komplikationer ät att kolla att x^2 är tillräcklig nära 11. Hur nära? Just nu har vi valt 0.0001 men vad man väljer beror på variabel typen man använder (float, real, double, osv). Det flesta programmerings språk har speciell konstruktion för absolut belopp som ofta heter `abs()` eller liknande. I praktiken skulle vi skriva ett styck kod som liknar den här
  ```
  if (abs(x*x-11)<0.0001) then {en viktig kod}
  ```
- `a=float(1)/3; if (`$|3a - 1| < 0.0001$`) then {en viktig kod}`
  Vi kollar att $3a$ är tillräcklig nära 1. I praktiken skulle vi skriva någonting så här
  ```
  if (abs(3*a-1)<0.0001) then {en viktig kod}
  ```
  Varför vi gör så? En dator skulle utvärdera $3a$ till någonting nära *0.99999*.

1.1.6 Komplexa tal

Komplexa tal finns inte i naturen med vi använder de ändå. Hur är detta möjligt? Det är en bra fråga alltså. Det grundläggande begreppet ”komplex tal” dyker upp när vi betraktar alla möjliga lösningar till ekvationen $x^2 + 1 = 0$. Vi antar att det finns ett tal i som beskriver lösningar, $x = \pm i$ där vi tolkar $i = \sqrt{-1}$. Varför skall en blivande IT/Data ingenjör lära sig komplexa tal? Som sagt, de är inte naturliga. Ju det kan man tycka. Men, likväl man kan ställa ett motargument: vi vet att $\sqrt{2}$ kan inte representeras i en dator, men vi använder den ändå. Så varför är det ok att ifrågasätta användning av i och inte användning av $\sqrt{2}$? Ja, samma som med roten av två, det finns argument varför det kan vara fördelaktig att använda komplexa tal.

Oavsett i vilken led man ställer sig angående frågor ovan, det komplexa talet är en användbar bokförings verktyg för att hålla reda på komplicerad algebra. Med hjälp av komplexa tal ganska komplicerade matematiska identiteter kan skrivas mycket kortare. Vitsen med att jobba med komplexa tal är att vi inte kan riktig räkna roten av negativ tall, sant, men vi kan jobba med i som det vore ett tal. Skulle det dyka upp potenser av i vi kan försöka förenkla så gott det går. För trotts allt, vi

[7] Centrum ligger i $x_c = \frac{a+b}{2}$ och radien är $r = \frac{b-a}{2}$. Förhållandet mellan x_c och r är rent geometrisk, förstås. Villkoren $x_c + r = b$ och att $x_c - r = a$ bara utrycker vad ”centrum” betyder.

kan alltid börja från $i^2 = -1$. Då vet vi att $i^3 = i^2 i = -i$, och så vidare. Att kunna räkna i^n med höga potenser är en kunskap **som testas systematisk på tentamen**.

Nästan varje programmeringsspråk tillåter operationer med komplexa tal, och har bibliotek med funktioner att jobba med dessa. Allt det här med komplexa tal är så viktig att komplexa talet är ett moment **som examineras på den skriftliga tentamen**. Det alltid kommer en fråga där man skall förenkla ett enklare algebraiskt uttryck som har med komplexa tal att göra.

Finemang, kan ni säga, du har fortfarande inte visat oss ett enda exempel varför det här är viktig. Ok, det finns en massa exempel som jag kunde ge. Till att börja med, ibland när man diagonaliserar en matris är det svårt att veta om det finns komplexa eller reella eigenvärde. Så vill man skriva en anmäl kod då måste man kunna tillåta komplexa lösningar och vara beredd att arbeta med komplexa tal i kod. Som Yoda skulle säg till Luke när han tränade för att bli en jediridare in en av Star Wars filmer: ”patience my young padwan”. Flera exempel kommer ganska snart. För att uppskatta det jag kommer att visa er, ni måste kunna arbeta med komplexa tal.

Användning av komplexa tal går ungefär så här. Först vi ”kodar” in problemet vi är intresserade i som ett komplex tal $z = x + iy$. Sen gör vi alla möjliga grejer med z, alltså räknar $w = f(z)$ där f en algoritm (funktion) som beskriver vad man skall jag göra och är problem beroende. Notera att resultatet w är också ett komplex tal. Sist, vad man vill göra till slut är att extrahera den reella och imaginära delen av w för den informationen finns vi vill åt (som sagt, några exempel senare). Det finns två speciella funktioner för detta ändamål som finns i de flesta datorprogram. Inom matematisk analys skriver vi

$$x = \mathrm{Re}(z) = \mathrm{Re}(x + iy)$$

$$y = \mathrm{Im}(z) = \mathrm{Im}(x + iy)$$

Alltså man väntar med stort tålamod och tuggar på z tills det är dags att räkna den reella delen eller den imaginära delen. Märklig nog, alla enkla algebraiska uttryck med komplexa tall kan förenklas, efter ganska naturliga algebra steg, till formen $z = x + iy$. Hur ser processen att ”tugga sig igenom”. Ju, vi ofta behöver räkna $z_1 + z_2$, $z_1 z_2$, z_1/z_2, eller z^n där $n \in N$. Tror ni vi kan räkna z^{-n} där $n \in N$? Jajamensan, det kan vi göra också! Kan vi räkna $\sin z$? Yeap, det går också, men man måste tänka ordentligt vad man verkligen menar när man gör så.

En bra grej med komplexa tal är att vi kan presentera de på ett naturligt sätt. Vi använder det komplexa tal planet C, som ni känner från gymnasiet.

Två viktiga funktioner är absolut belopp

$$|z| = \sqrt{x^2 + y^2}$$

och det komplexa konjugatet

$$z^* = (x + iy)^* = x - iy$$

Viktig: blanda inte konjugatet här med konjugatregeln från algebra. Det händer lika ofta att man betecknar det komplexa konjugatet med ett sträck som $\underline{z}$. Det är bra att ni bekantar er med bägge två notationer för de förekommer lika ofta.

Viktig: notera att Re och Im är reella tal, alltså Im är också ett reellt tal! Snälla, skriv inte (som ni brukar göra på tentamen) $\mathrm{Im}(2 + 3i) = 3i$, inte världens ände om ni gör så, men fel är det likväl. En superviktig grej är identiteten

$$zz^* = (x+iy)(x-iy) = xx - ixy + ixy - y^2i^2 = xx + yy = x^2 + y^2 = |z|^2$$

I uträkningen, vi har använt att $i^2 = -1$ för att skriva om $-y^2i^2$ till $-y^2(-1)$ som till sist blir y^2. Jajamensan, visst, vi vet, hör jag er säga, men ändå på tentamen gör folk (inte så sällan) en sådan felaktig uträkning

$$(2+3i)(2-3i) = 2 + (3i)^2 = 2 - 9$$

Alltså formeln ovan för zz^* säger att vi skall ta kvadrater av reella och imaginära delar, eller om jag skall vara övertydlig: $zz^* = \mathrm{Re}(z)^2 + \mathrm{Im}(z)^2$ och då gäller det att komma ihåg det jag har skrivit tidigare att $\mathrm{Im}(2+3i)$ är 3 och inte $3i$!

En väldigt viktig praktisk grej att kunna är att skriva om $z = x + iy$ till polär form

$$z = r(\cos\theta \ + i\sin\theta)$$

Se A.I.Graphical representation of Complex Numbers, texten innan meningen "The expression on the right side is called the polar representation". Jag kommer att kalla detta **polär form 1**.

Nu är det högt aktuellt att parata om en annan form av det komplexa talet, som jag skulle vilja kalla **polär form 2** som bygger på den berömda Euler formeln

$$e^{i\theta} = \cos\theta + i\sin\theta$$

Alltså med hjälp av Euler formeln vi kan skriva

$$z = re^{i\theta}$$

Speciellt fall med $x = \pi$ ger det som brukar kallas den vackraste ekvationen i matematiken

$$e^{i\pi} + 1 = 0$$

Exempel: Omvandla $z = 2 + 3i$ till polär form (den första kvadranten). Under gymnasiet/basåret ni har kanske gjort så här: Om $z = x + yi = re^{i\theta}$ då enligt receptet hur man omvandlar vi vet att (a) $r^2 = x^2 + y^2 = 2^2 + 3^2 = 4 + 9 = 13$, $r = \sqrt{13}$, och (b) $\tan\theta = \frac{y}{x} = \frac{3}{2}$, alltså $\theta = \arctan\frac{3}{2}$.

Exempel: Omvandla $z = -2 + 3i$ till polär form (den andra kvadranten). Samma sak här som i exemplet ovan; att (a) $r = \sqrt{13}$, och (b) $\tan\theta = \frac{3}{-2} = -\frac{3}{2}$, alltså $\theta = \arctan\left(-\frac{3}{2}\right) = -\arctan\frac{3}{2}$. Man måste vara försiktig i det sista steget när man antar att $\theta = \arctan\frac{3}{2}$. När en miniräknare "svarar" vad invers av tangens är då är det programmerad att alltid omvandla svar till den första kvadranten (om argumentet är positivt) eller den fjärde kvadranten (om argumentet är negativt). Det komplexa talet vi är intresserade i ligger i den andra kvadranten, så för det här problemet det korrekta svaret är $\theta = -\arctan\frac{3}{2} + \pi$ så man måste korrigera.

Exempel: Omvandla $z = -2 - 3i$ till polär form (den tredje kvadranten). Samma sak, avstånd ifrån origo är som den är men vinkeln man måste korrigera med extra en faktor: $\tan\theta = \frac{-3}{-2} = \frac{3}{2}$, $\theta = \arctan\frac{3}{2} + \pi$.

Exempel: Omvandla $z = 2 - 3i$ till polär form (den fjärde kvadranten); $\tan\theta = \frac{-3}{2} = -\frac{3}{2}$, $\theta = -\arctan\frac{3}{2}$.

Exempel: Låt $z_1 = e^{\frac{i\pi}{4}} = \frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}$ och $z_2 = e^{-\frac{i\pi}{4}} = \frac{1}{\sqrt{2}} - \frac{i}{\sqrt{2}}$. Vi vet att $z_2 = \frac{1}{z_1}$ därför att $e^{-a} = \frac{1}{e^a}$. Med $a = \frac{i\pi}{4}$ då är det sant att $e^{-\frac{i\pi}{4}} = \frac{1}{e^{\frac{i\pi}{4}}}$. Visa att samma sak gäller om de komplexa talen omvandlas från polär formen till den vanliga formen (rektangulär form). Svar: Vi omvandlar först från den polära formen till den rektangulära formen med hjälp av Euler satsen: $e^{\frac{i\pi}{4}} = \cos\frac{\pi}{4} + i\sin\frac{\pi}{4} = \frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}$ och $z_2 = e^{-\frac{i\pi}{4}} = e^{i\left(-\frac{\pi}{4}\right)} = \cos\left(-\frac{\pi}{4}\right) + i\sin\left(-\frac{\pi}{4}\right) = \cos\left(\frac{\pi}{4}\right) - i\sin\left(\frac{\pi}{4}\right) = \frac{1}{\sqrt{2}} - \frac{i}{\sqrt{2}}$. I det sista steget har vi använt att sinus är en udda funktion, $\sin(-x) = -\sin x$, och att cosinus är en jämn funktion, $\cos(-x) = \cos x$. Resten är vanligt algebra med lite hjälp av komplex konjugat: $\frac{1}{z_2} = \frac{1}{\frac{1}{\sqrt{2}} - \frac{i}{\sqrt{2}}} = \frac{1}{\frac{1}{\sqrt{2}} - \frac{i}{\sqrt{2}}} \cdot$

$$1 = \frac{1}{\frac{1}{\sqrt{2}} - \frac{i}{\sqrt{2}}} \cdot \frac{\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}}{\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}} = \frac{1 \cdot \left(\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}\right)}{\left(\frac{1}{\sqrt{2}} - \frac{i}{\sqrt{2}}\right) \cdot \left(\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}\right)} = \frac{\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}}{\left|\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}\right|^2} = \frac{\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}}{\frac{1}{2} + \frac{1}{2}} = \frac{\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}}}{1} = \frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}} = z_1.$$

Tillämpningar av Euler formel finns inom

- elektroniken
- uträkning av integraler
- lösningar av differentiella ekvationer
- uträkning av integraler

Här är ett exempel hur man kan räkna en ganska komplicerad integral med hjälp av komplexa tal. Ni kan verifiera att resultatet stämmer senare när vi lär oss lite extra om integrations tekniker. Just i det här fallet behöver vi partiell integration. Dock notera att alla steg efter att vi tillåter komplexa tal är ganska enkla. Ganska många av er har gjort sådana typer av integraler under gymnasiet, tex antar att många har räknat sådana integraler $\int e^{ax}dx$. Nytt med uträkningen under är att exponenten a är ett komplex tal.

Bestäm $\int e^x \cos x \, dx$ med hjälp av komplexa tal:

$\mathrm{Re}(z_1+z_2) = \mathrm{Re}(z_1) + \mathrm{Re}(z_2)$

$$\int e^x \cos x \, dx = \int e^x \mathrm{Re}\{\cos x + i \sin x\} \, dx = \int e^x \mathrm{Re}\{e^{ix}\} \, dx$$

$$= \mathrm{Re}\left\{ \int e^x e^{ix} \, dx \right\} = \mathrm{Re}\left\{ \int e^{(i+1)x} \, dx \right\}$$

$$\int e^{ax} dx = \frac{1}{a} e^{ax}$$

$$= \mathrm{Re}\left\{ \frac{e^{(i+1)x}}{i+1} + c \right\} = \mathrm{Re}\left\{ \frac{1-i}{2} e^{(i+1)x} + c \right\}$$

$$\frac{1}{1+i} = \frac{1-i}{(1+i)(1-i)} = \frac{1-i}{i^2+1^2} = \frac{1-i}{2}$$

$$= \mathrm{Re}\left\{ \frac{1-i}{2} e^x (\cos x + i \sin x) + c \right\}$$

$$= \mathrm{Re}\left\{ \frac{1}{2} e^x (\cos x + \sin x) + i(\dots) + c \right\}$$

$$= \frac{1}{2} e^x (\cos x + \sin x) + c$$

För att kunna visa den måste vi först gå igenom hur man kan approximera funktioner med polynomer. Det går att räkna e^x som ett polynom av en ganska hög grad. Notera att nu jobbar vi med reella tal. Ju större noggrannhet vill vi ha dessutom större antal termer i polynomet måste vi jobba med. När man approximerar man måste välja en **referenspunkt** där approximationen fungerar bäst eller är faktisk exakt. Senare vi kommer att vissa att följande polynomer är bättre och bättre approximationer till den exponentiella funktionen kring $x \approx 0$.

$$e^x \approx 1$$

$$e^x \approx 1 + x$$

$$e^x \approx 1 + x + \frac{x^2}{2!}$$

$$e^x \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!}$$

$$e^x \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}$$

$$e^x \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!}$$

Hemläxa: Försök göra grafer av funktioner ovan med ett program ni är bekanta med och jämför den exponentiella funktionen med olika polynom approximationer.

Ok, nu gäller det att komma ihåg att en Jedi riddare är redo att hitta sätt att behärska kraften som styr universum. En Jedi riddare skall inte vara rädd nu att testa vad händer om vi använder polynomer för att räkna $e^{i\theta}$ istället. Om man använder polynomer med $x = i\theta$ vad får vi då? En sista kontroll av ljus sabel och då kör vi…

$$e^{i\theta} \approx 1$$

$$e^{i\theta} \approx 1 + i\theta$$

$$e^{i\theta} \approx 1 + i\theta + \frac{(i\theta)^2}{2!} = 1 - \frac{\theta^2}{2!} + i\theta$$

$$e^{i\theta} \approx 1 + i\theta + \frac{(i\theta)^2}{2!} + \frac{(i\theta)^3}{3!} = 1 - \frac{\theta^2}{2!} + i\left(\theta - \frac{\theta^3}{3!}\right)$$

$$e^{i\theta} \approx 1 + i\theta + \frac{(i\theta)^2}{2!} + \frac{(i\theta)^3}{3!} + \frac{(i\theta)^4}{4!} = 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} + i\left(\theta - \frac{\theta^3}{3!}\right)$$

$$e^{i\theta} \approx 1 + i\theta + \frac{(i\theta)^2}{2!} + \frac{(i\theta)^3}{3!} + \frac{(i\theta)^4}{4!} + \frac{(i\theta)^5}{5!} = 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} + i\left(\theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!}\right)$$

Intressant, inte sant? Man ser ett tydligt mönster i det hela. Vi ser att om vi fortsätter då får vi

$$\mathrm{Re}\left(e^{i\theta}\right) = 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \cdots$$

$$\mathrm{Im}\left(e^{i\theta}\right) = \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \cdots$$

Vi kommer att visa senare att två kända funktioner kan approximeras med polynomer ovan

$$\cos\theta = 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \cdots$$

$$\sin\theta = \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \cdots$$

och då vet vi att

$$\mathrm{Re}\left(e^{i\theta}\right) = \cos\theta$$

$$\mathrm{Im}\left(e^{i\theta}\right) = \sin\theta$$

Nu har vi visat att om man vågar räkna på det sätet då kan vi lita på Euler formeln. Senare vi kommer att visa att de polynomapproximationer vi har använt håller. Alltså, vi kan lita på Euler formeln!

Varför allt det här? När man har omvandlat z till polär form då kan man utföra några operationer med komplexa tal ganska snabbt. Som ett exempel, om man omvandlar z till polär form $re^{i\theta}$ kan

man räkna z upp till en hög potens väldigt fort $(1+i)^{103}$. Vi kan till och med fundera på att räkna ganska tokiga grejer som $(1+i)^{\frac{1}{3}}$. Wow!

Allt bygger på att vi kan snabbt multiplicera två komplexa tal om de är skrivna i polär form, och vi kan räkna väldigt höga potenser av sådana tal. Nämligen med hänsyn till Euler formel är det inte svårt att argumentera att följande är sant:

$$(\cos\theta \ + i\sin\theta\ )^n = \cos n\theta \ + i\sin n\theta$$

$$(\cos\theta_1 \ + i\sin\theta_1\ )(\cos\theta_2 \ + i\sin\theta_2\ ) = \cos\left(\theta_1+\theta_2\right) \ + i\sin\left(\theta_1+\theta_2\right)$$

De två resultat hänger tät ihop. Det är ganska enkelt att se att $z_1z_2 = \left(r_1e^{i\theta_1}\right)\left(r_2e^{i\theta_2}\right) = (r_1r_2)(e^{i\theta_1}e^{i\theta_2})$. Läst texen kring A.I."Complex Arithmetic".TH1 (de Moivre's Theorem) för beviset att $e^{i\theta_1}e^{i\theta_2} = \cos\left(\theta_1+\theta_2\right) \ + i\sin\left(\theta_1+\theta_2\right)$ som kan skrivas om till $e^{i(\theta_1+\theta_2)}$. Det visar att samma vanliga regler som gäller den exponentiella funktionen $e^xe^y = e^{x+y}$ gäller för komplexa exponent så vi kan i lugn och ro räkna $\left(re^{i\theta}\right)^{103} = r^{103}e^{i103\theta}$. Det är bara att omvandla $1+i$ till $re^{i\theta}$ och använda de Moivre satsen eller liknande, och det är i princip gymnasiematte.

Exempel 1: Räkna $(1+i)^{103}$ med hjälp av polär formen av talet.

Först, vi måste omvandla talet till polärformen; $1+i = \sqrt{2}\frac{1+i}{\sqrt{2}} = \sqrt{2}\left(\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}i\right) = \sqrt{2}\left(\cos\frac{\pi}{4}+i\sin\frac{\pi}{4}\right) = \sqrt{2}e^{\frac{i\pi}{4}}$. Då kan vi räkna ganska enkelt höga potenser av $1+i$: $(1+i)^{103} = \left(\sqrt{2}e^{\frac{i\pi}{4}}\right)^{103} = \left(\sqrt{2}\right)^{103}e^{\frac{i\pi 103}{4}}$. Vi kan skriva om $\frac{\pi 103}{4}$ som $\frac{\pi 103}{4} = \frac{\pi 100}{4}+\frac{\pi 3}{4} = 25\pi+\frac{3}{4}\pi$. Det betyder att vi kör 12 hela varv och en halv, plus $\frac{\pi}{2}$ och sen $\frac{\pi}{4}$ till. Då hamnar man i $-\frac{\pi}{4}$. Alltså vi vet att nu att $e^{\frac{i\pi 103}{4}} = \cos\left(-\frac{\pi}{4}\right)+i\sin\left(-\frac{\pi}{4}\right) = \cos\left(\frac{\pi}{4}\right)-i\sin\left(\frac{\pi}{4}\right) = \frac{1}{\sqrt{2}}-i\frac{1}{\sqrt{2}} = \frac{1-i}{\sqrt{2}}$. Pusslar man ihop alla bidrag får vi $(1+i)^{103} = \left(\sqrt{2}\right)^{103}\frac{1-i}{\sqrt{2}} = \left(\sqrt{2}\right)^{102}(1-i) = 2^{51}(1-i)$.

Exempel 2: Räkna $(1+i)^{103}$ med ett alternativt sätt att tänka där man använder $(1+i)^2 = 2i$.

En enkel uträkning visar att $(1+i)^2 = (1+i)(1+i) = 1+2i+i^2 = 1+2i-1 = 2i$. Det är speciellt bara för det här fallet. Annars det behöver inte vara så att kvadrat av en komplex tall blir "rent" imaginärt tal. Tex $(2+i)^2 = 2^2+2\cdot 2\cdot i+i^2 = 4+4i-1 = 3+4i$ skiljer sig från $4i$. Tillbaka till problemet: $(1+i)^{103} = (1+i)^{2\cdot 51+1} = [(1+i)^2]^{51}(1+i) = [2i]^{51}(1+i) = 2^{51}i^{51}(1+i)$. Hur skall man räkna i^{51}. Vi testar: i är bara vad den är, $i^2 = -1$, $i^3 = i^2i = -i$, $i^4 = i^2i^2 = 1\cdot 1 = 1$, osv. Vi ser att vi måste utrycka exponenten i form $a+4b$ alltså räkna division moduls 4; $i^{51} = i^{4\cdot 12+3} = (i^4)^{12}i^3 = (1)^{12}i^3 = i^3 = -i$. Pusslar man ihop alla bidrag får vi $(1+i)^{103} = 2^{51}(-i)(1+i) = 2^{51}(-i-i^2) = 2^{51}(-i-(-1)) = 2^{51}(1-i)$.

Exempel 3: Hitta alla komplexa tal sådana att deras kvadrat blir "rent" imaginärt.

Vi vill att $(x+iy)^2 = wi$ och vi vill hitta alla möjliga kombinationer av x och y som tillåter detta. Det är bara att räkna: $(x+iy)^2 = x^2+2xyi+y^2i^2 = x^2+2xyi-y^2 = x^2-y^2+2xyi$. Vi ser att villkoret är $x^2-y^2 = 0$ och då får vi $w = 2xy$. Alltså alla tal i formen $x\pm ix = x(1\pm i)$ är sådana

att deras kvadrat är ”rent” imaginärt. Det här är intressant. Vi vet att $1+i$ i polär form är $1+i=\sqrt{2}e^{\frac{i\pi}{4}}$. Visst, kvadrat av $e^{\frac{i\pi}{4}}$ blir ”rent” imaginärt för $\left(e^{\frac{i\pi}{4}}\right)^2=e^{\frac{i2\pi}{4}}=e^{\frac{i\pi}{2}}=i$. Hur är det med $1-i$? Allt bli bra. Matematiken sviker oss inte; $1-i=\sqrt{2}e^{-\frac{i\pi}{4}}$ och $\left(e^{-\frac{i\pi}{4}}\right)^2=e^{-\frac{i2\pi}{4}}=e^{-\frac{i\pi}{2}}=-i$.

Jag antar att ni har haft komplexa tal under gymnasiet. För att fräscha upp (eller lära på nytt) följ tabellen nedan

begrepp	att läsa	att träna	
$z=x+iy$	A.I.”Definition of compex numbers”; DEF 1	A.I.1, 13, 18	
$Re(z)$	A.I.”Definition of compex numbers”; DEF 2		
$Im(z)$	A.I.”Definition of compex numbers”; DEF 2		
C	A.I.”Graphical Representation of Complex Numbers”		
$z_1 \pm z_2$	A.I.”Complex Arithmetic”; EX 3	A.I.34	A.I.43
$z_1 z_2$	A.I.”Complex Arithmetic”; EX 4	A.I.36	
z_1/z_2	A.I.”Complex Arithmetic”; EX 6	A.I.40	
$\|z\|$	A.I.”Graphical Representation of Complex Number”; DEF3	A.I.5	
z^* eller $\underline{z}$	A.I.”Graphical Representation of Complex Number”; DEF5	A.I.24,26	
polär form	A.I.”Complex Arithmetic”; efter EX 4, “It is particularly easy to determine the product…”; A.II.”The Complex Derivative”→ “The Exponential Function”	A.I.”Complex Arithmetic”, EX 5 A.I.46 A.II.17 $(i+1)^{103}=?$ $(i+1)^{\frac{1}{3}}=?$	
komplexa rötter	A.I.”Complex Numbers” > Roots of Complex Numbers	A.I.EX7, A.I.EX8, A.I.51-53	

Antar att nu har ni läst allt om komplexa tal och gick igenom angivna delar av ADAMS. Ni tror ni förtjänar sättet i Jedi rådet. Finemang, låt mig utmana med en tentamen fråga, fixar ni det får ni 1/7 del av sättet:

Vad är realdelen och imaginärdelen av z?

$$z=\frac{(3+i^{27})(2-i^{1013})}{(1+i^5)(1+2i)}+\frac{10}{\sqrt{2}e^{\frac{i\pi}{4}}+1}$$

1.2 Enkla och komplicerade funktioner

Hur kan man formalisera operationer med tal? Hur kan man formalisera vad en dator gör? Vilka matematiska abstraktioner lämpar sig att beskriva en dator som löser en optimerings problem? Alla dessa frågor har ett enkelt svar: funktion. Kanske det viktigaste milstolpe är att kunna visualisera hur ett matematiskt uttryck kan omvandlas till linjer i ett datorprogram. Som ett exempel:

matematiskt uttryck	datorprogram[8]

$f(x) = \dfrac{x^2+2x+1}{x+1}$	`real function f(real x)`
	`  real a,b,c`
	`  a = x*x+2*x+1`
	`  b = x+1`
	`  c = a/b`
	`  return c`
	`end function`

Vi diskuterar i detalj hur man kan definiera funktioner och diskuterar tekniker för att arbeta med funktioner. Vi börjar med en viktig grej: notation. Senare, kommer vi att diskutera trigonometriska funktioner (mycket fort för dem har ni tuggat på under gymnasiet ganska mycket) och hyperboliska funktioner. Hyperboliska funktioner är någonting som kommer att vara nytt. Speciellt nytt kommer att vara diskussion kring hur dessa två grupper av funktioner är relaterade till varandra.

1.2.1 Allmänt om funktioner

Funktioner kommer i olika former. Enkla som ni har diskuterat under gymnasiet (sammansatta och invers), och komplicerade som vi kommer att fokusera på (implicita). Sammansatta och inversfunktioner hänger tätt ihop. Vårt fokus kommer att vara på derivata av komplicerade funktioner. I följande texten vi kommer att introducera problemet, men kommer att lösa den senare.

Sammansatta och invers funktioner: Jag antar att ni är bekanta från gymnasiet vad en sammansatt funktion är, tex $h(x) = f(g(x))$, och vad en inversfunktion är $h(x) = f^{-1}(x)$; $f\left(f^{-1}(x)\right) = x$ och $f^{-1}\left(f(x)\right) = x$. Det vanliga receptet som ni säkert känner till för att hitta inversfunktion om $f(x)$ är känd är att skriva $f(y) = x$ och lösa ekvationen för y. I bild, detta betyder följande:

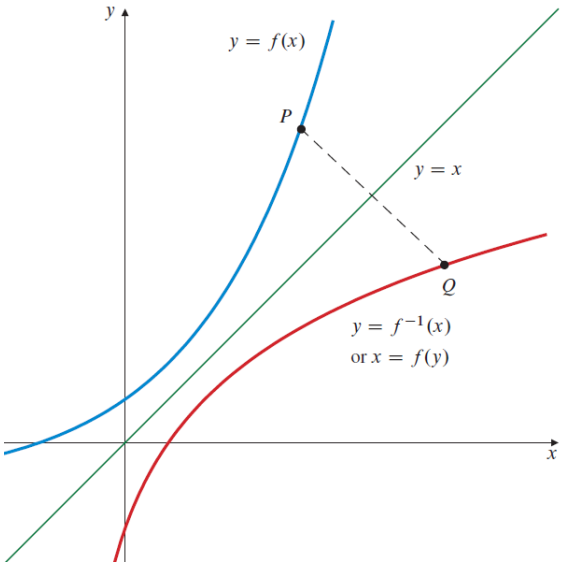


Figure 3.3 The graph of $y = f^{-1}(x)$ (red) is the reflection of the graph of $y = f(x)$ (blue) in the line $y = x$ (green)

Men, för oss, vi kommer att fokusera på formalia. **När har man faktisk lov för att definiera inversfunktion?** Och för att svara på denna fråga, är det viktig att komma ihåg vad en funktion är till att börja med. Detta är speciellt viktig när vi analyserar vilka funktioner har invers! Det viktigaste är att känna till från en angiven graf om den representerar en funktion eller inte. Det är just detta som

[8] Ser ni möjligheter att förbättra koden? Det finns flera faktisk. Tips: Det första man säger till en funktion som man träffar är inte ”god dag, hur mår ni idag” men ”god dag, var har ni för definitionsmängd”.

avgör när vi kan hitta invers eller inte. En bra övning som kommer att hjälpa er förstå vad är en funktion och inte är en funktion är P.4.7. Starkt rekommenderas!

Om någon känner sig osäker på hur man hittar en inversfunktion läs igenom 3.1 Inverse Functions. För hur man gör med funktioner som har flera ”grenar” (är inte ”one-to-one”) se 3.1 Inverse Functions > Inverting Non-One-to-One Functions. Ett typiskt exempel av sådan funktion diskuteras i texten under. Några bra problem att träna på: 3.1.EX1, 3.1.EX3, 3.1.2, 3.1.3, 3.1.6, och 3.1.21.

Ett typiskt exempel när man diskuterar invers är funktionen $f(x) = x^2$. Den har en ganska känd invers, inte sant? Men, om vi försöker hitta invers till den kvadratiska funktionen det går inte faktisk! Problemet är att funktionen är inte bijektiv: den mappar inte ”one-on-one”. Den mappar ”two-to-one” för tex bägge -2 och -2 skickas till 4; kolla själva att $f(2) = 4$ och $f(-2) = 4$. Men vi ändå jobbar med $\sqrt{x}$, inte sant? Hur kommer det sig att jobbar med invers dock inversen borde inte finnas om nu följer de grundläggande principer? Svaret finns i texten kring bilden (3.1 Inverse Functions > Inverting Non-One-to-One Functions):

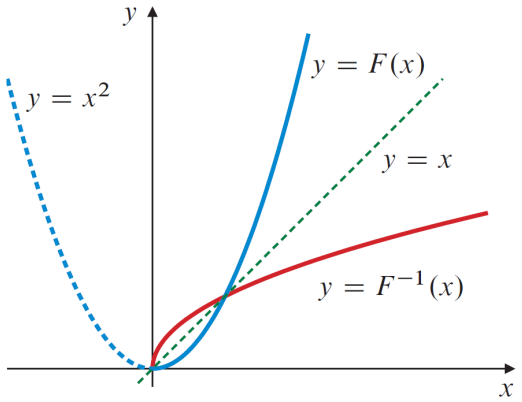


Figure 3.5 The restriction F of x^2 (blue) to $[0, \infty)$ and its inverse F^{-1} (red)

Man kan omdefiniera funktionen genom att göra dess **definitions mängd** smalare och då kan man definiera invers. Ofta vi jobbar med två definitions mänger: $R_+ = \{x | x \in R, x \geq 0\}$ och $R_- = \{x | x \in R, x \leq 0\}$. Då får man två inversfunktioner $\pm\sqrt{x}$ och bägge två är inverser, dock inte till samma funktion. Varför? Ju, det är viktig att komma ihåg att en funktion definieras också med sin definitionsmängd. Vi pratar om två funktioner $f_+: R_+ \to R$ och $f_-: R_- \to R$. Visst det är sant att $f_+(x) = f(x)$ och $f_-(x) = f(x)$ men dessa två funktioner opererar på olika mängder, och vi betraktar de som olika.

1.2.2 Olika sätt att beskriva kombinationer av funktioner

En viktig skillnad från gymnasiet är att vi kommer att bli extra noga med formalia, och framför allt notationen kring hur man arbetar med funktioner! **En bra notation är som grönt te, den renar tankar och skapar struktur i hur vi tänker (och skriver kod)!**

En sak som ni har kanske sett under gymnasiet är hur man kan kombinera olika funktioner.

kombination	vad den gör
$h(x) = f(x) \pm g(x)$	addition och subtraktion

$h(x) = f(x)g(x)$	produkt
$h(x) = \dfrac{f(x)}{g(x)}$	division av två funktioner
$h(x) = kf(x)$	multiplikation med en konstant
$h(x) = f(x)f(x)$	definierar en ny funktion som vi brukar kalla för ”kvadrat av funktionen” $f^2(x)$

Vad är viktig här är att inse att kombinationer av funktioner definierar nya funktioner. Ibland har vi en bra notation för nya kombinationer, för vi anser de viktiga, men ibland inte. Som ett exempel, kan hända att vi behöver räkna ofta kvadrat av en funktion, vi skriver $h_1 = f^2$. Samma sak med summan och division av två funktioner; vi skriver $h_2 = f \pm g$ och $h_3 = \frac{f}{g}$. Det kan dyka upp ett problem här med notationen. Nämligen, om vi vill att en funktion-kombination h tuggar på ett visst tal x, då vill vi beskriva det genom att skriva $h(x)$. Låt oss se vad kan hända om vi lägger till ett (x) bredvid h som $h \cdots (x) \to h(x)$.

- Notation exempel med h_1: I fal av kvadraten det blir inga konstigheter: $h_1 \cdots (x) \to f^2(x)$ ser bra ut.
- Notation exempel med h_2: Men $h_2 \cdots (x) \to f \pm g(x)$ ser inte alls bra ut. I ärlighets namn, uttrycket är en ren dumhet, för $g(x)$ redan har en betydelse och då f blir hängande kvar utan argument. Förstås, vi försöker fixa detta med hjälp av parenteser, och det blir mycket bättre att skriva $h_2 \cdots (x) \to (h_2) \cdots (x) \to (f \pm g)(x)$. Nu blir det korrekt. Och den här notationen uttrycker med oändlig noggrannhet vad vi menar, inte sant?
- Notation exempel med h_3: Samma gäller för $h_3 \cdots (x) \to \frac{f}{g}(x)$ vilket blir kanske en smula konstig, så det vore bättre att skriva $h_3 \cdots (x) \to (h_3) \cdots (x) \to \left(\frac{f}{g}\right)(x)$ alltså $h_3 \equiv \frac{f}{g}$. Vi skriver sällan så, men det är det vi menar när vi skriver $\frac{f(x)}{g(x)}$.

Tror ni att det kan inte blir krångligare? Tyvärr det kan det. Låt oss prata om en kombination av funktioner där vi först räknar $f(x)$ och sedan drar roten av det vi har räknat: $h_4(x) = \sqrt{f(x)}$. Ett ganska precis sätt att skriva är $h_4(x) = \left(\sqrt{f}\right)(x)$ och $h_4 \equiv \left(\sqrt{f}\right) = \sqrt{f}$, men vi gör sällan så, inte sant.

Nu ner vi har diskuterat notationen då kan vi titta på den berömda notationen för inversfunktion f^{-1}. Det finns en fara att tolka f^{-1} som $\frac{1}{f}$ men det är ofta tydligt från kontexten vad vi menar, menar vi invers eller värkligen $1/f$.

Varför är det här snacket om notation viktig för en IT ingenjör? Ju, det kan hända att när man skriver kod då vill man använda en vis kombination av funktioner upprepande gånger, och då är det klokt att reservera ett speciellt namn för kombinationen. Detta gör kod transparent och enkelt att underhålla. För gör man ändringar, då behöver man göra bara ändringar på ett enda ställe. Inom ”software engineering” detta heter ”encapsulation” och paras ihop med ”code normalization”. Som ett exempel, låt oss anta att vi vill räkna väldigt ofta $\sin(x) + \cos(\sqrt{x})$. Då kan man skriva en sådan bit av kod

```
real function SinPlusCosSqrt(real x)
   real sqx, s, c, resultat
```

```
    sqx = sqrt(x)
    s = sin(x)
    c=cos(sqx)
    resultat = s + c
return resultat
```

och det blir ganska enkelt att använda funktionen. I de flesta programmeringsspråk man använder funktionen igenom att ”klistra” argumentet till namnet.

```
BRA_PROGRAM( )
…
a = SinPlusCosSqrt(1.2)
….
b = SinPlusCosSqrt(3.5)
…
for (i=-10,10)
    …
    temp = 10/i
    SinPlusCosSqrt(temp)
    …
endfor
```

Visst vi kunde ha skrivit samma program utan funktionen

```
DÅLIG_PROGRAM( )
…
a = sin(1.2)+cos(sqrt(1.2))
…
b = sin(3.5)+cos(sqrt(3.5))
…
for (i=-10,10)
    …
    temp = 10/i
    sin(temp)+cos(sqrt(temp))
    …
endfor
```

men då kommer vi på att vi inte kan riktig räkna roten av ett negativt tal. Mycket enklare att rätta till på ett ställe, i koden för `SinPlusCosSqrt`, än att leta i hela programmet för sådana ställen där vi räknar $\sin(x) + \cos\sqrt{x}$, inte sant? Som ett exempel, en förbättrad version av koden kunde se ut så här:

```
real function SinPlusCosSqrt(real x)
    real sqx, s, c, resultat
    if(x<0) abort("SinPlusCosSqrt-E: roten av negativt tal")
    sqx = sqrt(x)
    s = sin(x)
    c=cos(sqx)
    resultat = s + c
return resultat
```

Varför är den här versionen bättre? Man kan ju påpeka att vi inte har ändrat någonting, bara dödat körningen (processen). Ju, skillnaden är att vi dödar körningen med ett fel meddelande. Programmeraren eller användaren vet varifrån felet kommer. Då kan man kolla hur det kommer sig att man har kallat funktionen med negativ x.

1.2.3 Implicita funktioner med kända eller okända namn

Senare vi kommer att diskutera derivata av komplicerade funktioner som är definierad på ett implicit sätt. Alla funktioner i tabellen är definierade implicit, alltså lösningar av en ekvation definierar funktionen i två steg. Steg 1: man väljer ett x. Steg 2: Använd x-et vi valde att lösa ekvationen.

funktion	implicit ekvation
$y = f^{-1}(x)$	$f(y) = x$
$y = x^{\frac{1}{n}}$	$y^n = x$
$y = \arcsin x$	$\sin y = x$
$y = \arccos x$	$\cos y = x$
$y = \ln x$	$e^y = x$
$y = \pm\sqrt{x+1} - 1$	$y^2 + 2y = x$
?	$y^{11} + 3y^7 + 5y^2 + y = x$

Exempel: Anta att x är känd. Vilka y löser ekvationen $y^2 + 2y = x$? Just i det här fallet vi kan räkna analytisk: $y^2 + 2y + 1 - 1 = x \rightarrow (y+1)^2 - 1 = x \rightarrow (y+1)^2 = x + 1$. Drar man roten av hela ekvationen får vi $y + 1 = \pm\sqrt{x+1}$ som kan skrivas om till $y = -1 \pm \sqrt{x+1}$. Vi ser att vi alltid har två lösningar (om ingenting tokigt händer med argumentet av roten, alltså $x + 1$ måste vara positivt). Notera att i det här fallet vi kunde faktisk hitta den explicita formen för funktionen $f(x) = -1 \pm \sqrt{x+1}$. Anledning till det är att vi kunde hitta inversen till $g(y) = y^2 + 2y$. Från $x = g(y)$ kan man få $g^{-1}(x) = g^{-1}\big(g(y)\big) = y$ som ger $y = g^{-1}(x)$. Funktionen vi är intresserade i är faktisk $g^{-1}(x)$ det är just denna funktion som definierar den implicita funktionen vi jagar, alltså $f(x) = g^{-1}(x)$.

Exempel: Hur gör man med $y^{11} + 3y^7 + 5y^2 + y = x$? Tyvärr, här kan vi inte hitta inversen till $g(y) = y^{11} + 3y^7 + 5y^2 + y$. Man måste jobba numerisk. Här är Mathematica kod som gör jobbet:

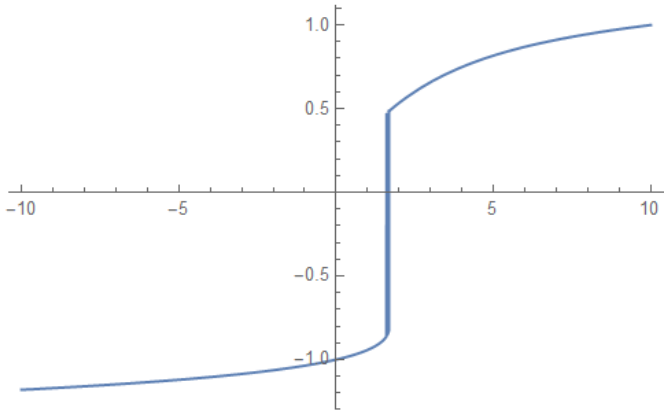
```
In[•]:= eq = y^11 + 3 y^7 + 5 y^2 + y == x
```

Out[•]= $y + 5y^2 + 3y^7 + y^{11} == x$

```
In[•]:= f[xarg_] :=
         (
          eqx = eq /. x → xarg;
          sol = Solve[eqx, y][[1]];
          y /. sol
         )

In[•]:= Plot[Evaluate@f[x], {x, -10, 10}]
```

Out[•]=



Om man översätter till vanlig kod, det skulle se ut så här:

```
real function implict_function(real x)
   local explicit function g(y) = y^11 + 3*y^7 + 5*y^2 + y
   xguess = x;
   repeat
      xnew = förbättra_lösningen(xguess, g)
      diff = abs(xnew-xguess)
      xguess = xnew
      sluta = diff<0.0001 // av typ boolean
   until sluta
end real function
```

En funderare: med hänsyn till diskussionen ovan (se ”Olika sätt att beskriva kombinationer av funktioner”), vad kan vi säga att f^{-1} är för att alla fall i tabellen? För visa det går ganska enkelt att säga vad f är (tex. $arcsin$ eller $arccos$), men för visa är det ganska krångligt. Ser ni vilka är enkla och vilka är svåra? Det finns en till som är enkel, kanske två.[9]

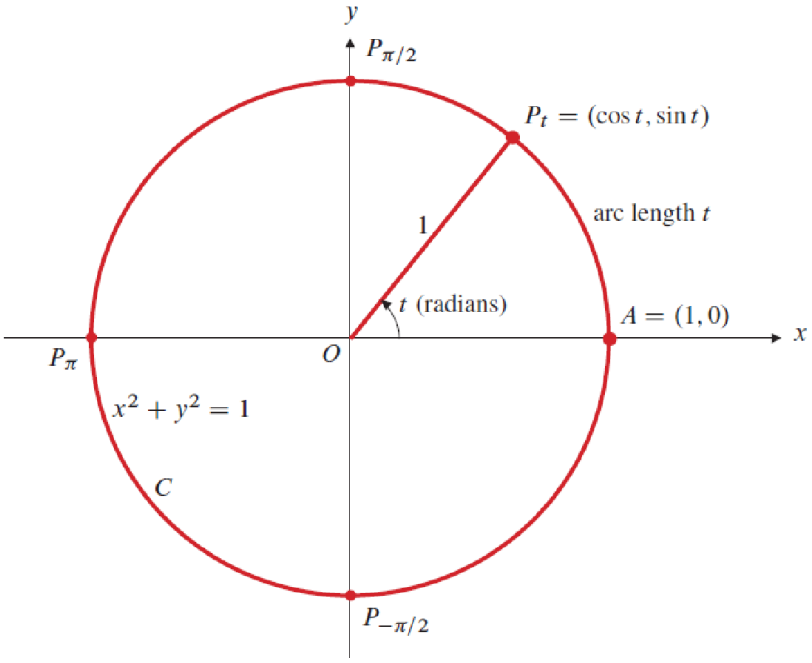
För visa funktioner definierade på det här sättet har vi speciella namn vi använder $\mathbf{arcsin}\, \boldsymbol{x}$ eller $\mathbf{ln}\, \boldsymbol{x}$ men det är inte alla funktioner som är kändisar. Visa gör sitt jobb tyst och diskret utan att

[9] $f^{-1} = \ln$ är självklart fall, för allt blir ok om man vill betona att vi utvärderar funktionen på ett vis tal x: $f^{-1} \cdots (x) \to \ln \quad \cdots (x) \to \ln(x)$. Kanske man kan hävda att $f^{-1} = \sqrt[n]{\ }$ är någorlunda ok, dock när vi klistrar in argumentet som vi brukar göra $f^{-1} \cdots (x) \to \sqrt[n]{\ } \cdots (x) \to \sqrt[n]{\ }(x)$ det blir en smula konstig notation, inte sant? För $= \pm\sqrt{x+1} - 1$ man kunde kanske definiera i kod en funktion som kunde heta `SqrtXplus1minus1` och använda den med `SqrtXplus1minus1(x)` .

många känner till de, kanske bara personen som programmerar in de. Tex den sista raden är sådan funktion som kan dyka upp i praktiska tillämpningar men har inget namn. Senare, vi kommer att visa hur man räknar derivata av sådana implicita funktioner.

1.2.4 Trigonometriska funktioner

Jag kommer inte att orda mycket om trigonometriska funktioner för ni har bekantat er med dessa under gymnasiet. Men jag kommer att betona några viktiga grejer. För en Jedi ridare är enhets cirkeln det bästa vapnet i kampen att behålla ordningen i universum.



Alla förväntas kunna räkna värden för enkla vinklar som tex 0, $\pi/4\ n$, $\pi/2\ n$, πn, $\frac{\pi}{3}n$ där $n = 1,2,3$ osv. Notera att studenter brukar uppleva 0 eller $\pi/2$ som svårast, konstig nog.

Att fräscha upp gymnasiet kunskaper kring trigonometriska funktioner läs: P.7.The Trigonometric Functions.

Om man inspekterar enhetscirkeln ser man hur den berömda trigonometriska ettan dyker upp; det är bara att använda Pythagoras sats

$$\sin^2 x + \cos^2 x = 1$$

Additions formler är viktiga men svårare att härleda.

$$sin\ (x + y) = sin\ x\ \ cos\ y + cos\ x\ \ sin\ y$$

$$cos\ (x + y) = cos\ x\ \ cos\ y - sin\ x\ \ sin\ y$$

För beviset se satsen P.7.TH2. Jag stark rekommenderar att ni lär de där formlerna utantill. Från additions formler man kan härleda alla andra, tex formler för dubbla vinklar, i.e. att omvandla från 2x till x:

$$sin\ (2x) = sin\ (x + x) = sin\ x\ \ cos\ x + cos\ x\ \ sin\ x = 2\ sin\ x\ \ cos\ x$$

$$cos\ (2x) = cos\ (x + x) = cos\ x\ \ cos\ x - sin\ x\ \ sin\ x = \cos^2 x - \sin^2 x$$

Också det är viktig att kunna omvandla från x/2 till x

$$sin\left(\frac{x}{2}\right) = \cdots \cos x \ldots sin\ x\ \ldots ?$$

$$cos\left(\frac{x}{2}\right) = \cdots \cos x \ldots sin\ x\ \ldots ?$$

Varför behöver man det? Ju, om vi kan räkna tex $\sin\frac{\pi}{4}$ då kan vi också räkna $\sin\frac{\pi}{8}$. Hur hittar man sådana formler för halv vinkel? Enklast kanske att använda formel för dubbel vinkel av cos. Notera att vi kan skriva om formel som $cos\ (2x) = \ 1 - 2\sin^2 x$, och byta x mot x/2. Detta kommer att ge oss formel för halv vinkel av sin. På samma sätt kan vi skriva $\cos 2x = 2\cos^2 x - 1$, och igen byta x mot x/2. Detta kommer att oss formel för halv vinkel av cos.

$$sin\frac{x}{2} = \pm\sqrt{\frac{1 - \cos x}{2}}$$

$$cos\frac{x}{2} = \pm\sqrt{\frac{1 + \cos x}{2}}$$

Var försiktig med att välja rätt tecken! Man måste kolla mot rätt kvadrant.

1.2.5 Hyperboliska funktioner

Hyperboliska funktioner är definierade med hjälp av den exponentiella funktionen $\exp(x)$:

$$\sinh x = \frac{1}{2}(e^x - e^{-x})$$

$$\cosh x = \frac{1}{2}(e^x + e^{-x})$$

$$tanh\ x = \frac{sinh\ x}{cosh\ x}$$

$$coth\ x = \frac{cosh\ x}{sinh\ x}$$

Varför vill man jobba med sådana funktioner? Det visar sig att i många praktiska tillämpningar av matematiken sådana kombinationer av exp(x) och exp(-x) dyker upp naturligt.

Dessutom, det finns en cool symmetri, ekvivalens mellan hyperboliska funktioner och trigonometriska funktioner. Tex trigonometrisk etta, additions formler, formler för dubbla vinklar är ganska lika. Tom derivator beter sig ganska likt.

Tabell: matchande identiteter för trig och hyp funktioner	
Identiteter för trigonometriska funktioner	Identiteter för hyperboliska funktioner
$\sin^2 x + \cos^2 x = 1$	$\cosh^2 x - \sinh^2 x = 1$
$\sin 2x = 2\sin x \cos x$	$\sinh 2x = 2\sinh x \cosh x$
$\cos 2x = \cos^2 x - \sin^2 x$	$\cosh 2x = \cosh^2 x + \sinh^2 x$

Varför är det så? Kan vi förklara dem mystiska ändringar i visa tecken? I kort, allt beror på Euler formeln som vi diskuterar lite senare.

För att fräscha upp eller lär på nytt om hyperboliska funktioner se 3.6."Hyperbolic Functions".DEF15 och DEF16 men inte mycket mera just nu (titta inte på Cauchy-Riemann, derivator av komplexa funktionen, etc). Kolla också beviset för $\cosh^2 x - \sinh^2 x = 1$ strax under DEF15, leta efter:

$$\cosh^2 t - \sinh^2 t = 1 \quad \text{for any real } t.$$

och försök göra uträkningen där. Titta också på symmetriegenskaper av hyperboliska funktioner

$$\cosh(-x) = \cosh x, \qquad \sinh(-x) = -\sinh x.$$

En funderare: Använd symmetriegenskap ovan av den hyperboliske sinus, $sinh\ (-x) = -sinh\ x$, för att visa att det måste vara sant att $sinh\ 0 = 0$.

Gör övningen 3.6.2 för att visa additions formler for hyperboliska funktioner

$$\cosh(x+y) = \cosh x \cosh y + \sinh x \sinh y,$$
$$\sinh(x+y) = \sinh x \cosh y + \cosh x \sinh y,$$

1.2.6 Algebraiska likheter mellan trigonometriska och hyperboliska funktioner

Tidigare diskuterade vi att man kan tycka att all det här med Euler är strunt prat för komplexa tal de finns ju inte i naturen, och det är sant. Det finns ett argument att all det här är konstig. Men, Euler formeln är en ganska användbar bokföring system. Man kan relatera olika funktioner på ett elegant sätt. Nu kommer vi att diskutera ett viktigt exempel!

Remark The distinction between trigonometric and hyperbolic functions largely disappears if we allow complex numbers instead of just real numbers as variables. If i is the imaginary unit (so that $i^2 = -1$), then

Nu när vi kan Eulers formeln, kan vi diskutera relationen mellan trigonometriska och hyperboliska funktioner. De är relaterad på två sätt

trig(ix)→hyp(x)	**hyp(ix) →trig(x)**
$\sin ix = i \sinh x$ $\cos ix = \cosh x$	$\sinh ix = i \sin x$ $\cosh ix = \cos x$

Man kan "fusk" bevisa formler ovan från Eulers formel, om vi antar att x kan var komplex tal, $x = iu$

$$e^{i(iu)} = cos\ (iu) + i\ sin\ (iu)$$

$$e^{i(-iu)} = cos\ (-iu) + i\ sin\ (-iu)$$

och då får vi två ekvationer med två okända $cos(iu)$ och $sin(iu)$

$$e^{-u} = cos\ (iu) + i\ sin\ (iu)$$

$$e^{u} = cos\ (iu) - i\ sin\ (iu)$$

Här har jag också "fusk" räknat med symmetriegenskaper av sin och cos

$$sin\ (-x)\ = -\ sin\ x$$

$$cos\ (-x)\ = cos\ x$$

och antog att dessa gäller även för komplexa tal. Löser man ekvation systemet, två ekvationer med två okända $cos(iu)$ och $sin(iu)$, får man den angivna relationen mellan trigonometriska och hyperboliska funktioner. Adderar/subtraherar man ekvationer får vi

$$e^u + e^{-u} = 2\ cos\ (iu)$$

$$e^u - e^{-u} = -2i\ sin\ (iu)$$

som kan skrivas om till (dela den första ekvationen med 2, multiplicera den andra med $\frac{i}{2}$)

$$sin\ (iu)\ = \frac{i}{2}(e^u - e^{-u}) = i\ sh(u)$$

$$cos\ (iu)\ = \frac{1}{2}(e^u + e^{-u}) = ch(u)$$

Likaväl med hjälp av Euler satsen man kan visa att

$$\sinh iu\ = \frac{1}{2}\left[e^{iu} - e^{-iu}\right] = \frac{1}{2}[\cancel{\cos u}\ + i\sin u - (\cancel{\cos u}\ - i\sin u)] = i\sin u$$

$$\cosh iu\ = \frac{1}{2}\left[e^{iu} + e^{-iu}\right] = \frac{1}{2}[\cos u + \cancel{i\sin u} + (\cos u\ - \cancel{i\sin u})] = \cos u$$

En bra övning: Försök relatera formler för trigonometriska och hyperboliska funktioner. Tex, börja från trigonometriska identiteter i Tabell ” matchande identiteter för trig och hyp funktioner” och visa att motsvarande hyperboliska identiteter gäller. Sedan, byt sidan och visa den motsatta riktningen. Sist men inte minst, gör samma för additions formler.

En funderare: Jag ”frågade” Mathematica vad reell och imaginär dela av $\sin(2+3i)$ och $\sinh(2+3i)$ är. Man kan använda ComplexExpand kommando för detta. Den har ”spottat” ut (ganska snabbt) svaren nedan. Kunde ni förklara hur Mathematica har ”tänkt” här?

```
Sin[2 + 3 I] // ComplexExpand

Cosh[3] Sin[2] + i Cos[2] Sinh[3]

Sinh[2 + 3 I] // ComplexExpand

i Cosh[2] Sin[3] + Cos[3] Sinh[2]
```

1.3 Ytterligare likheter mellan trigonometriska och hyperboliska funktioner

Nyckelord: likheter mellan trigonometriska och hyperboliska funktioner med hänsyn till derivata, enkla derivataregler, den primitiva funktionen, obestämd integral, differentiella ekvationer.

Vi kommer att prata ytterligare likheter mellan trigonometriska och hyperboliska funktioner. Framför allt i det här kapitlet fokuserar vi på likheter men hänsyn till derivata. Diskussionen kommer

att leda oss, på ett naturligt sätt, till begrepp som **den primitiva funktionen** som är relaterad med **den obestämda integralen**, och **differentialekvationer**. Det kan hända att några av er har sett en differentialekvation under gymnasiet dock antar jag att ni inte har ”brottats” med sådant tidigare. I ärlighetens namn är det lite för tidigt att prata om differentiella funktioner redan nu, men differentialekvationer är en av de viktigaste modellerings verktyg vi har, så det är lika bra att börja prata om dem tidigt. Under kursens gång kommer vi att förfina våra kunskaper om hur man jobbar med dem. Just nu är det viktigaste att ni förstår **vad en differentialekvation är**. Lyckas vi med det, då är vi nöjda. Vi kommer att göra det genom att betrakta några enkla exempel: den enklaste formen av en differentialekvation dyker upp naturligt när man försöker hitta en primitiv funktion.

En primitiv funktion är ganska nära besläktad med obestämd integral, och då är det inte konstig att integraler och differentialekvationer hänger tätt ihop. Vi kommer att se flera exempel av detta under kursens gång, t.ex. när vi kommer att prata om den linjära approximationen. Man kan tycka att det är konstig att differentialekvationer och linjära approximationen hänger ihop med det gör de faktisk. Men, vi kan inte prata om allt på en gång, inte sant?

1.3.1 Derivata av enklare funktioner och deras kombinationer vi behöver

Under gymnasiet ni har bekantat är med derivata av enkla funktioner. Antar att ni har sett liknande tabeller under gymnasiet:

$f(x)$	$f'(x)$
x^n	nx^{n-1}
e^x	e^x
$sin\ x$	$cos\ x$
$\cos x$	$-\sin x$

Ni har också jobbat med derivata av algebraiska kombinationer av enklare funktioner och är mer eller mindre bekanta med regler hur man jobbar med sådana uttryck

Tabell: derivataregler	
uttryck	derivata regeln
$f(x) \pm g(x)$	$f'(x) \pm g'(x)$
$f(x)g(x)$	$f'(x)g(x) + f(x)g'(x)$
$\frac{f(x)}{g(x)}$	$\frac{f'(x)g(x) - f(x)g'(x)}{g(x)^2}$
$h(x) = f(g(x))$	$h'(x) = \left[f\big(g(x)\big)\right]' = f'\big(g(x)\big)g'(x)$

Ser ni detta för första gången? Om ”ja”, var inte oroliga. Vi kommer att prata om allt under kursens gång och kommer att diskutera var identiteter i tabeller ovan kommer ifrån. Dock antar att de flesta har sett detta tidigare. Men, just nu, för att diskutera ytterligare relation mellan trigonometriska och hyperboliska funktioner behöver vi bara två **regler** som jag kommer att skriva explicit. Vi kommer att behöva en regel till för att kunna diskutera några exempel av den primitiva funktionen. Vi kommer att bevisa alla sådana regler senare (ner vi bevisar alla regler i tabellen ovanför). Just nu är det viktig att ni vet hur man använder dessa två regler.

Regel 1: Derivatan av en linjär kombination av funktioner:

$$\big(af(x) + bg(x)\big)' = af'(x) + bg'(x)$$

Som ett exempel

$$(2\sin x + 3e^x)' = 2(\sin x)' + 3(e^x)' = 2\cos x + 3e^x$$

Regel 2: Den andra regeln uttalar sig om derivator av funktioner med argument som är produkt av en konstant a och x

$$\big(f(ax)\big)' = af'(ax)$$

Som ett exempel

$$(\sin(2x))' = 2\sin'(2x) = 2\cos(2x)$$

Regel 3: Derivatan av potens när exponenten är heltal:

$$(x^n)' = nx^{n-1}$$

Senare vi kommer att visa att samma gäller när exponenten är ett rationellt tal. Några exempel

$$1' = (x^0)' = 0x^{-1} = 0$$

$$(x^1)' = 1x^0 = 1$$

$$(x^2)' = 2x^1 = 2x$$

$$(x^3)' = 3x^2$$

Regel 1 är ganska enkelt att fatta och använda. Samma gäller för Regel 3. Dock, jag har märk att studenter har problem med Regel 2. För det första de fattar inte riktig det där med prickar, för jag flyttar de omkring på ett ganska konstigt sätt. För det andra, de kan fundera vad sinus med prick betyder, alltså vad $\sin' \blacksquare$ föreställer. Här har jag gömt argumentet med flyt och bara skriver en lite ”svartlåda” för att ögat inte fastnar vid fel ställe. Jag vill att ni fokuserar på ”sin” tecknet och pricken ”’” som jag har klämt intill.

Vi kommer att använda en regel till när vi diskuterar några exempel av den primitiva funktionen.

Jag har märkt att studenter har ofta problem att ”skriva” matematiken. Jag har också upptäckt att brist på denna förmåga ofta relaterar med brister i kunskaper. Anledning att studenter skriver ”smutsig” är att studenter ibland inte riktig förstår konceptet som de försöker utrycka i ekvationen. Inte alltid, men det händer. Inte så konstig att ett viktigt lärande mål i kursen är att träna studenter att kunna skriva ”ren” matematik, tex att alla prickar och parenteser kommer på rätt plats. Så det här med derivatanotationen förtjänar ett litet kapitel, för att ni kan lära er arbeta med en bra notation och utöva en hög precision när ni skriver ner matematiska uttryck, så att alla fattar vad ni menar.

1.3.2 Ett typiskt misstag studenter gör med den enklaste kedjeregeln

Finemang, nu undrar vi: Vad betyder f' och när man kan flytta pricken runt? Ni vet att $f'(x)$ betyder derivata av $f(x)$. Alltså vi menar att $f'(x) = (f(x))'$. Men, **ser ni skillnaden mellan den vänstra och den högra sidan?** Vi kan skriva den vänstra sidan utan argument: f'. Detta är en funktion som kan stå på sina egna ben, och har ett namn: f'. Den är vad den är. Den högra sidan däremot vi kan inte skriva utan argument x. Det skulle se konstig ut, $(f(\,))$, inte sant? Den högra sidan liknar snarare en algoritm, där vi förklarar vad är det som pågår:

- Steg 1: mata in ett uttryck in parentesen $(\dots)$; uttrycket har ett namn: $f(x)$.
- Steg 2: räkna derivatan av det som står i parentesen, $(\dots)'$.

Senare vi kommer att förklara vad algoritmen egentligen är.

Notera att väldigt ofta istället för $(f(x))'$ skriver vi bara $f(x)'$. Så i det här fallet det spelar ingen roll vad man lägger pricken. Man bara utrycker ett dugg annorlunda vad man menar, men innehållet är exakt densamma.

Däremot, ett typisk fel studenter gör är följande. Vi kommer att titta på ett exempel där det inte att flytta derivatatecken, men studenter gör det ändå. Inte kanske explicit, men de tänker att det går bra att göra så. Alltså, det är fel att skriva $f'(ax) = (f(ax))'$. Varför? Det räcker att vi hittar ett motexempel. Så, låt oss testa med $f(x) = x^2$.

VS: Ni vet att $f'(u) = 2u$, så med $u = ax$ får vi $f'(ax) = 2(ax) = 2ax$

HS: Den högra sidan är derivatan av en sammansatt funktion. Först vi måste räkna $f(ax)$ vilket blir $(ax)^2$ och senare räkna derivata av det vi fick, vilket alltsammans blir $(a^2x^2)' = a^2(x^2)' = a^2 2x$. Här har vi använt regeln som säger att vi alltid kan ta ut konstanten utanför derivatatecken. Vi kommer att kalla sådana steg för **"städa konstanter"** principen.

1.3.3 Varifrån kommer den enklaste kedjeregeln?

Det kommer att hända ofta att vi betraktar en funktion $g(x)$ och senare en modifierat version av den $h(x) = g(ax)$ där a är en konstant. Att räkna derivatan av $h'(x)$ blir nästa steg, och ofta utmaningen är att relatera h' till g' för de är inte lika dana (se diskussionen ovan). Hur räknar man på ett korrekt sätt? Jag har märkt att många studenter har problem, eller "slarvar bort" den under den skriftliga tentamen. Det här lilla kapitlet är menad att hjälpa studenter med detta.

Man måste använda kedjeregelreceptet med korrekt identifierade funktioner. Vi vet vad den yttre funktionen är: den är g. Den inre funktionen är $f(x) = ax$. Så g' är vad den är, det beror på problemet, men som tur är f' kan vi räkna ganska enkelt, $f'(x) = (ax)' = a \cdot (x)' = a \cdot 1 = a$. Användning av kedjeregeln ger $h'(x) = g'(ax)a$. Vi ofta brukar skriva konstanten först: $h'(x) = ag'(ax)$. Kanske enklast att tänka är att ha den mentala bilden att $(g(ax))'$ är lika med $ag'(u)$ där $u = ax$. Några exempel finns i tabellen nedan som ni kan träna på.

$g(u)$	$g'(u)$	$(g(ax))' = ag'(ax)$
u^2	$2u$	$((ax)^2)' = a \cdot 2ax$
u^7	$7u^6$	$((ax)^7)' = a \cdot 7(ax)^6$
e^u	e^u	$(e^{ax})' = a \cdot e^{ax}$
$ln\, u$	$\frac{1}{u}$	$(ln\, ax)' = a \cdot \frac{1}{ax} = \frac{1}{x}$
$sin\, u$	$cos\, u$	$(sin\, ax)' = a \cdot cos\, ax$
$cos\, u$	$-sin\, u$	$(cos\, ax)' = a \cdot (-sin\, ax)$
$sinh\, u$	$cosh\, u$	$(sinh\, ax)' = a \cdot cosh\, ax$
$cosh\, u$	$sinh\, u$	$(cosh\, ax)' = a \cdot sinh\, ax$

Om tabellen ovan känns främmande försök gå igenom 2.4.EX4a, 2.5.3, 2.5.4, och 2.5.5.

1.3.4 Trigonometriska och hyperboliska funktioner kommer från liknande DE

Nu använder vi allt vi har diskuterat hittills till att analysera relationen mellan derivatan av trigonometriska funktioner och hyperboliska funktioner. Det är bara att räkna:

$(\sin x)' = cos\ x$; $(sin\ x)'' = (cos\ x)' = -\ sin\ x$

$(\cos x)' = -\ sin\ x$; $(cos\ x)'' = -(sin\ x)' = -\ cos\ x$

Vi ser att för $y = sin\ x$ eller $y = cos\ x$ det gäller att $y'' = -y$ eller

$$y'' + y = 0$$

Om vi räknar med hyperboliska funktioner är det ganska lika uttryck men inte exakt

$(sinh\ x)' = cosh\ x$; $(sinh\ x)'' = (cosh\ x)' = sinh\ x$

$(cosh\ x)' = sinh\ x$; $(cosh\ x)'' = (sinh\ x)' = cosh\ x$

Vi ser att för $y = sinh\ x$ eller $y = cosh\ \ x$ det gäller att $y'' = y$ eller

$$y'' - y = 0$$

Om man gör samma sak med $y = sin\ \ kx$ eller $y = cos\ kx$

$(\sin kx)' = k\ cos\ kx$; $y'' = (sin\ kx)'' = (k\ cos\ kx)' = k(cos\ kx)' = k(-k\ sin\ kx)$

$(\sin kx)' = -k\ sin\ kx$; $y'' = (cos\ kx)'' = (-k\ sin\ kx)' = (-k)(sin\ kx)' = -k \cdot k\ cos\ kx$

och vi ser att vi får

$$y'' + k^2 y = 0\ (*)$$

Med $y = sinh\ \ kx$ eller $y = cosh\ cosh\ kx$

$(\sinh kx)' = k\ cosh\ kx$; $y'' = (sinh\ kx)'' = (k\ cosh\ kx)' = k(cosh\ kx)' = k(k\ sinh\ kx)$

$(\cosh kx)' = k\ sinh\ kx$; $y'' = (cosh\ kx)'' = (k\ sinh\ kx)' = k(sinh\ kx)' = k(k\ cosh\ kx)$

och vi ser att följande stämmer

$$y'' - k^2 y = 0\ (\times)$$

Kan vi se varifrån skillnaden på tecken i $(*)$ och $(\times)$ kommer ifrån? Ju, om man gör samma sak med $y = e^{\kappa x}$ får vi ganska lika uttryck

$(e^{\kappa x})' = \kappa e^{\kappa x}$; $(e^{\kappa x})'' = (\kappa e^{\kappa x})' = \kappa(e^{\kappa x})' = \kappa(\kappa e^{\kappa x}) = \kappa^2 e^{\kappa x}$

som kan skrivas om som

$$y'' - \kappa^2 y = 0$$

Bokstaven κ uttalas som ”kappa”. Vi har inte än bestämt vad κ är. Vi tittar på två fall.

Fal 1: Om vi tar $\kappa = k \in R$ då på grund att $\kappa^2 = k^2$ får vi samma ekvation som i $(*)$**.**

Fal 2: Om vi tar $\kappa = ik$, alltså när κ är ren komplex tal, då på grund av $\kappa^2 = (ik)^2 = i^2 k^2 = -k^2$ får vi samma ekvation som i $(\times)$.

Ekvationer $(*)$ och $(\times)$ beskriver två viktiga problem mekaniken. Ekvationen $(*)$ beskriver fri oscillator, en massa som **dras till origo** ju mera dessutom längre är den ifrån origo. Det finns ofta i naturen. Ekvationen $(\times)$ beskriver den raka motsatsen, en massa som **dras ifrån origo** ju mera dessutom längre är den ifrån origo, och sådan sannolikt finns i naturen men jag vågar inte svära på heder och samvete att det är så.

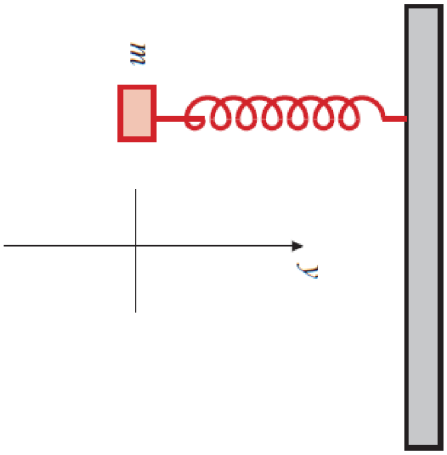


Figure 3.29

Om man vill utveckla en matematisk modell för problemet i bilden då ekvationen $(*)$ kan användas där y är avstånd mellan massan och plankan och x föreställer tid. Så i tid, position av massan skulle beskrivas som

Fal 1: $y(t) = A\sin kt + B\cos kt$

Fal 2: $y(t) = A\sinh kt + B\cosh kt$

I bägge fall är A och B fria konstanter som beror på den ursprungliga positionen och hastigheten av massan. I fal 1 massan rör sig fram och tillbaka, men i fall 2 massan för sig ifrån origo (och kommer aldrig tillbaka). Så, man kan uppleva att sin och $sinh$ är väldigt annorlunda men från den matematiska synpunkten är de ganska lika. Tokig, inte sant?

1.3.5 Differentiella ekvationer intro

I exemplet ovan vi har börjat från en funktion och sedan konstruerat en ekvation funktionen tillfredsställer. Vi har hittat en kombination av funktionen och dess derivata som tillfredsställer ett villkor av typ

$$\phi\big(y(x), y'(x), y''(x)\big) = f(x) \qquad (*)$$

Vi har sett att i fall av $y(x) = \sin x$ hade vi $\phi(y(x), y'(x), y''(x)) = y(x) + 0 \cdot y'(x) + y''(x)$ med $f(x) = 0$. Man kan konstruera flera sådana ekvationer, tex låt oss börja från $\sin x + \cos x - \sin x - \cos x = 0$. Ni vet vi att $y(x) = \sin x$, $y'(x) = \cos x$, och $y''(x) = -\sin x$. Då kan vi skriva om: $y(x) + y'(x) + y''(x) - \cos x = 0$ och då får vi ekvation

$$y(x) + y'(x) + y''(x) = \cos x$$

Vi vet att lösningen till ekvationen ovan är $y(x) = \sin x$ för vi har börjat just från detta. Men, i tillämpningar av matematiken har vi ofta det motsatta problemet. Det dyker ofta upp situationer när vi vet hur (*) ser ut med vi vet inte hur $y(x)$ ser ut. Sådana ekvationer heter **differentiella ekvationer**, och vi kommer att prata om dessa senare. Just nu räcker det med att ni vet vad en DE är.

1.3.6 Den primitiva funktionen och obestämd integral

Det finns ett speciellt fall av DE som är väldigt viktig, som ni faktisk har sett tidigare i en annan form. Kanske den enklaste differentiella ekvationen har formen

$$y'(x) = f(x)$$

och vi antar att vi vet $f(x)$ men vi letar efter $y(x)$ sådan att $y'(x) = f(x)$. Just i det här fallet, funktionen $y(x)$ som vi letar efter har ett speciellt namn: **den primitiva funktionen**. Speciellt för det här fallet är att man skriver ofta den primitiva funktionen som $F(x)$. Vi bara byter den lilla bokstaven y till den stora F. Så, om f är given vi letar efter F så att $F' = f$ gäller. Som ett exempel, vi kan undra vilket F skall vi använda för att det här stämmer

$$F'(x) = \sin x$$

Den primitiva funktionen kan utryckas som **en obestämd integral**

$$F(x) = \int f(x)dx$$

och för exemplet ovan

$$F(x) = \int \sin x \, dx$$

och tittar man på tabeller med integraler då hittar man lätt att $\int \sin x \, dx = -\cos x + c$ där c är den fria konstanten som är viktig. Det finns tekniker som man kan använda för att räkna sådana obestämda integraler, och vi kommer att prata om dessa senare. Ni är kanske bekanta med de enklaste tekniker att ”städa konstanter” och ”att dela integral i två”:

$$\int kf(x)dx = k\int f(x)dx$$

$$\int (f+g)dx = \int f dx + \int g dx$$

På det sättet kan man säga att derivata och integral är motsatta operationer. Nämligen

$$\int f'(x)dx = f(x) + c$$

Hur vet vi det? Ju, ekvationen som motsvara integralen ovan är

$$y'(x) = f'(x)$$

Vi frågar vilken funktion skall vi derivera för att få $f'(x)$. Självklart, vi skall derivera $f(x)$. Av någon konstig anledning studenter brukar ha problem med detta.

Exempel 1: Anta att vi har en bit paper där det står skrivet $x' = 1$ och vi har den gömd i fickan? Nu kommer den vanliga frågan, vilken funktion skall vi derivera för att få 1? Med andra ord, vad är det primitiva funktionen av 1? Eller, för vilken F är det sant att $F' = 1$. Med informationen vi har i fickan är det lite enklare att lösa problemet, men är F lika med x eller är F lika med 1? Det kan våra svårt att bestämma har jag märkt. Vilken skall man välja? Pausa läsning en smula och tänk.[10] Om det här känns svårt gå igenom 2.10.EX1 och 2.10.EX2. Följ upp med 2.10.1, 2.10.2, och 2.10.13.

Exempel 2: Samma situation som i exempel 1. Det står på pappret att $(x^2)' = 2x$. Vad är den primitiva funktionen av $2x$? Är det x^2 eller $2x$?[11]

Exempel 3: Nu antar vi att det finns inget papper. Vad är den primitiva funktionen av x^3? Vilken funktion skall vi derivera för att få x^3? Nu är det svårt, för det finns ingen pappret i fickan som vi kan

[10] Svar: $F(x) = x$ för derivatan ”slår” på x, inte på 1.
[11] Svar $F(x) = x^2$ för $F'(x) = (x^2)' = 2x$.

kika i, inte sant? Det enda vi ser är resultat av derivatan. Vi vet att vi har deriverat ett uttryck, $F(x)$, men har inte det blekaste aning vad den är. Så, nu måste vi räkna $F(x) = \int x^3 dx$ på något sätt. Och jag antar att ni har gjort detta under gymnasiet; $F(x) = \frac{x^4}{4} + c$ där c är det fria konstanten.

Exempel 4: En kropp rör sig med hastigheten varierande hastigheten $v(t) = 2t$. Hur beror position av kroppen s på tid t? Alltså, hitta s som funktion av t: $s(t)$. Svar: vi vet att $v(t) = s'(t)$ då måste det vara så att $s(t) = \int v(t)dt = \int (2t)dt$. Med andra ord, $s(t)$ är funktionen vi behöver derivera för att få $2t$? Antar att ni har gjort detta under gymnasiet; $s(t) = t^2 + c$. Om det här känns svårt, kolla två exempel under 2.11 ”Velocity and Acceleration”: 2.11.EX1 och 2.11.EX2. Problemet 2.3.3 kan bli ett bra stöd.

Exempel 5: En kropp accelererar med en konstant acceleration a. Hur ändrar sig kroppens position s och hastighet v som funktion av tid t? Svar: Senare, vi kommer att prata om det exakta sättet att definiera acceleration och hastighet är $a(t) = v'(t)$ och $v(t) = s'(t)$. Då kan man räkna $v(t) = \int \quad a(t)dt = \int \quad adt = a\int \quad dt = at + k_1$ och sedan använda $v(t) = at + k_1$ att visa $s(t) = \int \quad v(t)dt = \int (at + k_1)dt = a\int tdt + k_1\int dt = \frac{at^2}{2} + k_1 t + k_2$.