HÖGSKOLAN SKÖVDE

NATURVETENSKAP

Introduktion till predikatlogik

Jörgen Sjögren

Högskolan i Skövde
Institutionen för naturvetenskap
2002

Förord

Det som följer på dessa dryga hundra sidor är ett av otaliga försök som gjorts att presentera satslogik och predikatlogik på ett förhoppningsvis begripligt sätt. Därtill har författaren ivrigt uppmanats av ett antal studentkullar vid Högskolan i Skövde. Nedanstående är en fysisk manifestation av uppmaningarna. Huruvida framställningen är smaklig återstår att upptäcka.

Texten är tänkt att presentera sats- och predikatlogik, inklusive Gödels fullständighetssats, på ett både begripligt och korrekt sätt. Den förutsätter viss kännedom om mängdteori, funktionsbegreppet, relationer och induktionsbevis. Detta kan inhämtas i till exempel standardlitteratur om diskret matematik såsom Rosen, Discrete Mathematics (se litteraturlista). Texten förutsätter också en viss tidigare kännedom om satslogikens elementa. Det är inget krav, men underlättar läsningen, eftersom vissa delar av satslogiken är tämligen kortfattat presenterat.

I ett första möte med formell logik är det viktigt att se många lösta exempel, och kompendiet är rikligt utrustat med dylika. Med dessa exempel bör man arbeta aktivt. Det är också viktigt att lösa många uppgifter själv, för att logik ska kunna bli ett redskap och inte ett problem. Förslag till lösningar på övningsuppgifter bör diskuteras med studiekamrater och handledare.

Kompendiet består av fyra delar. Inom respektive avsnitt är definitioner, teorem och exempel internt numrerade. En referens inom ett avsnitt sker enbart med detta nummer. En referens mellan avsnitten sker med avsnittsnummer och till exempel definitionsnummer.

Mer initierade läsare av detta kompendium inser snart att författaren är kraftigt påverkad av bland annat [Mates], [Mendelson] och [Bennet m fl].

Jörgen Sjögren
Skövde, juni 01

Den version som här föreligger skiljer sig från föregångaren framför allt i att diverse fel av språklig karaktär rättats. En del oklarheter av logisk karaktär har också korrigerats.

Jörgen Sjögren
Skövde, maj 02

Innehåll

1 EN INLEDNING TILL LOGIK

Logik kan förenklat beskrivas som studiet av *giltiga slutledningar*, korrekta resonemang. Så är t ex följande två *informella slutledningar* båda giltiga.

Exempel 1: Alla människor är dödliga.
Sokrates är en människa.
Sokrates är dödlig.

Exempel 2: Alla kaniner tycker om morötter.
Sebastian är en kanin.
Sebastian tycker om morötter.

Satserna ovanför strecket kallas *premisser*. Strecket markerar en slutsatsdragning och kan utläsas ”alltså”. Satsen under strecket kallas *slutsats*. Skilj noga mellan slutledning och slutsats. Observera att de två slutledningarna i exemplen har samma *logiska form*. Båda är uppbyggda enligt mönstret

Alla A är B.
S är ett A.
S är ett B.

Alla slutledningar, som är uppbyggda enligt detta mönster, är giltiga. Här finns två idéer som är centrala.

(I) En slutledning är giltig om och endast om slutsatsen är sann under förutsättning att premisserna är sanna.

Detta innebär t ex att det inte samtidigt skall gå att bejaka premisserna och förneka slutsatsen utan att motsäga sig själv, dvs utan att hamna i absurditeter.

(II) En slutledning är giltig om och endast om den har en viss logisk form.

Vad som avses med ”logisk form” är komplicerat. Vi skall i detta kompendium studera två hjälpmedel vid analys av satsers logiska form - *satslogik* och *predikatlogik*.

Om semantik och syntax

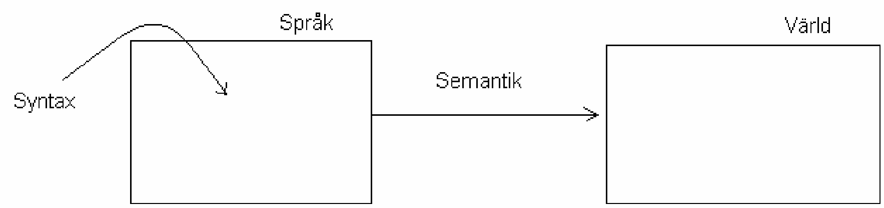
Studiet av tecknen, semiotiken, brukar delas in i följande områden.

Semantik, eller betydelselära, handlar om tecknens relation till världen, dvs hur språk och värld hänger ihop.

Syntax, eller ordfogningslära, handlar om hur ord fogas samman till ordgrupper eller satser. Ordet ”syntax” används ibland också för de regler som bestämmer hur giltiga uttryck i ett språk ser ut.

Pragmatik handlar om språkets användning.

Vi kan tillägga att ”sema” betyder just tecken. För oss är semantik och syntax viktigast. Nedanstående bild är illustrativ.

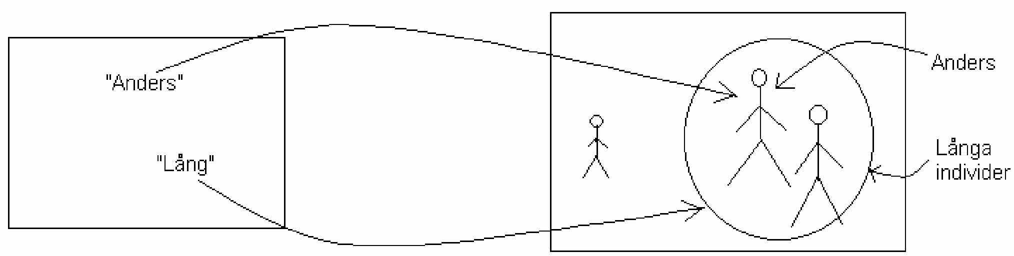


Figuren är tänkt att illustrera att syntax är något internt språkligt, medan semantik handlar om relationen mellan språk och värld.

Något om semantik

Semantik handlar, som sagt, om hur språk och värld hänger ihop, om hur och vad ord betyder. I semantik är vissa begrepp, så kallade *semantiska begrepp*, viktiga. Exempel på dylika är *sann*, *satisfierbar* och *giltig*. Vi kan illustrera att sanning är ett semantiskt begrepp med hjälp av följande exempel.

Exempel 3: Betrakta satsen ”Anders är lång.”. För att kunna avgöra huruvida satsen är sann eller ej, måste vi veta vilken individ ordet ”Anders” är namn på och huruvida denna individ hör till mängden av långa individer.



Här har ”Anders” parats ihop med en individ som har egenskapen att vara lång. I ovanstående ”värld” (*struktur* eller *modell*) är satsen således sann. Huruvida en sats är sann eller ej är, inte helt förvånande, beroende av hur världen är beskaffad, och hur språk och värld hänger ihop. Därmed kan man förhoppningsvis inse att sanning är ett semantiskt begrepp.

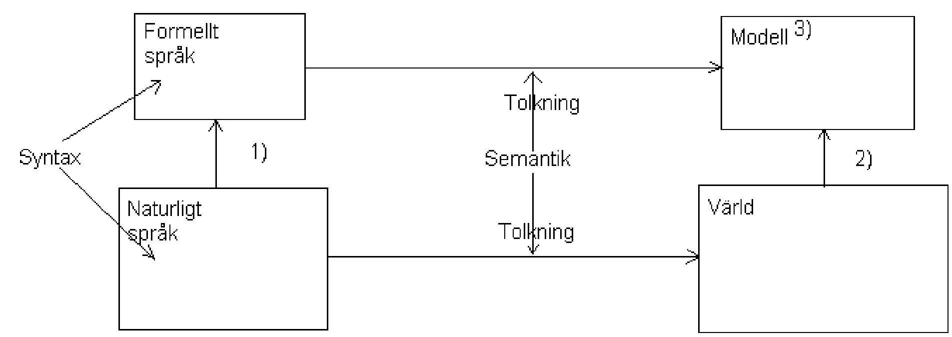
Vi säger att en (*öppen*) *formel*, t ex ”x är lång”, *satisfieras* av en individ, om vi erhåller en sann sats då vi ersätter variabeln med ett namn på denna individ.

Exempel 4: Individen Anders satisfierar formeln ”x är lång” eftersom ordet ”Anders” insatt i stället för variabeln ”x” ger en sann sats.

En formel är *satisfierbar*, om den satisfieras av någon (eller några) individer. Eftersom ”satisfierbar” här är definierat utifrån sanningsbegreppet, som är semantiskt, så är begreppet satisfierbar också semantiskt. Ytterligare ett exempel som illustrerar ovanstående är följande.

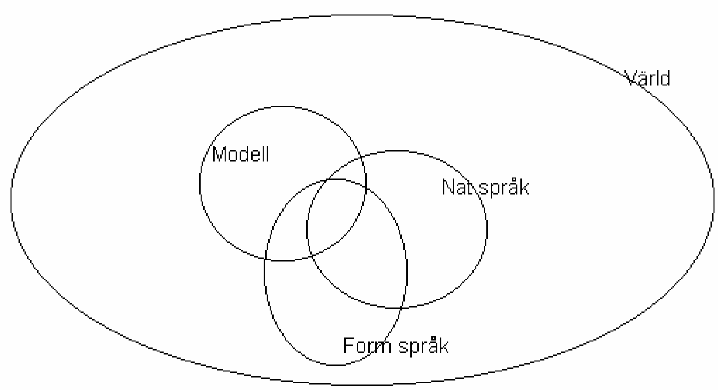
Exempel 5: Talet tre satisfierar formeln ”x + 2 = 5” eftersom satsen ”3 + 2 = 5” är sann.

När vi parar ihop ord i ett språk med ett fragment av en ”riktig” eller en konstgjord värld ger vi en tolkning av symbolerna i språket. För *naturliga språk*, som svenska, kinesiska, engelska etc, sker detta via konventioner (icke överenskomna dylika dock) i och med att vi lär oss ett modersmål. När vi använder *formella språk*, som satslogik, predikatlogik, C++, PROLOG etc, styr vi själva hur kopplingen mellan språk och värld skall göras. Betrakta följande bild.



1) är tänkt att avbilda någon intressant struktur i ett (ofta naturligt) språk.
2) är tänkt att avbilda någon intressant struktur i världen.
3) Med begreppet struktur (modell) avses ofta paret tolkning – individmängd.

Egentligen finns ju allt detta i världen, så bilden borde ju vara något i stil med nedanstående, men då det inte lika klart hur de olika komponenterna hänger ihop.



Ett sätt att förklara skillnaden mellan formella språk och naturliga språk är att säga att ett formellt språk har en medvetet konstruerad, precis syntax, medan naturliga språk inte är medvetet konstruerade och saknar precis syntax. Ett formellt språk är skapat för att användas i någon form av analys eller beskrivning av någon intressant struktur, t ex matematisk argumentering eller lingvistisk analys. Ett naturligt språk har växt fram under årtusenden tillsammans med vårt människoblivande.

Något om syntax

Alla ord (symboler) i ett språk utgör ett *lexikon* (*vokabulär*, *alfabet*). Syntax handlar t ex om vilka symbolsammanställningar (ordsammanställningar) som är korrekta, enligt en uppsättning regler.

Hur skulle man kunna avgöra om en given symbolsammanställning är giltig (korrekt)? En möjlighet skulle vara att skapa en lista över alla korrekta symbolsammanställningar (giltiga strängar) och sedan helt enkelt kontrollera om den givna strängen finns med i listan. Finns strängen med är den giltig, finns den inte med är den ogiltig. Listan kan förstås inte göras färdig, om det totala antalet korrekta strängar är mycket stort eller oändligt. Man kan i och för sig tänka sig att vi skulle kunna generera listan i någon alfabetisk ordning, och att den är klar åtminstone så långt som vi skulle behöva för att kunna testa en aktuell sträng, men för ett oändligt språk, behöver vi inte ha några garantier för att orden kan räknas upp i alfabetisk ordning. En annan möjlighet skulle vara att ha tillgång till en (liten?) uppsättning regler som genererar precis de giltiga strängarna. Vi utgår från ett exempel från [Hofstadter] för att illustrera några syntaktiska begrepp.

Vår vokabulär består av tre symboler M, I och U, dvs V = {M, I, U}. En *sträng över V* är en ändlig följd av symboler från V, t ex

MU
UM
MIU
MIIUI
IUM
osv

Det finns dessutom en uppsättning regler med hjälp av vilka man kan generera giltiga strängar utifrån tidigare skapade giltiga strängar. En giltig sträng kallar vi ett *teorem*. Nedan betecknar variabelsymbolerna x och y strängar, eventuellt tomma, över vokabulären V. (Se till exempel [Rosen] ss208f och kap 10.1 om strängterminologi.)

Regel 1: Om xI är ett teorem, så är också xIU ett teorem.

Regel 2: Om Mx är ett teorem, så är Mxx ett teorem.

Regel 3: Om xIIIy är ett teorem, så är xUy ett teorem.

Regel 4: Om xUUy är ett teorem, så är xy ett teorem.

Exempel 6: Om MII är ett teorem, så är MIIU ett teorem enligt regel 1 (x = MI).
Om MII är ett teorem, så är MIIII ett teorem enligt regel 2 (x = II).
Om MIIII är ett teorem, så är MUI ett teorem enligt regel 3 (vad är x resp y här?).Vi inför beteckningen "$\vdash$ x", som utläses "x är ett teorem".
$\vdash$ MUUU $\Rightarrow$ $\vdash$ MU enligt regel 4.

I detta lilla "spel" kan man generera teorem, givet att man har en uppsättning teorem att utgå från. Ett startteorem, ett gratisteorem, kallas ett *axiom*. I detta system har vi ett axiom.

Axiom: MI

Exempel 7: Problem: Generera MUIIU.

Lösning:	MI	(Axiom)
	MII	(Regel 2)
	MIIII	(Regel 2)
	MIIIIU	(Regel 1)
	MUIU	(Regel 3)
	MUIUUIU	(Regel 2)
	MUIIU	(Regel 4)

Ovan har vi ett exempel på en härledning (bevis). En *härledning* består av en svit av strängar sådan att en sträng erhålls från tidigare strängar med hjälp av någon av reglerna. Begreppet härledning (bevis) är syntaktiskt. Detta innebär att vi i en härledning manipulerar symboler utan att ta hänsyn till vad de betyder, likt en dators manipulerande av symboler.

Att fundera på 1: Härled (generera) MU.

Det skulle vara bra om det funnes en metod med hjälp av vilken man skulle kunna avgöra om en given sträng är ett teorem eller ej. Om så är fallet, är det formella systemet *avgörbart*.

Att fundera på 2: Fundera över huruvida MIU-systemet avgörbart? Om detta kan man läsa i [Hofstadter].

Observera att en dator enbart fungerar på ett syntaktiskt plan, medan en människa även opererar på ett semantiskt plan. Vill vi att en dator skall vara ”intelligent” krävs att vi lyckas konstruera dem så att de även kan operera på ett semantiskt plan.

Om giltiga slutledningar igen

Betrakta följande informella slutledning

Exempel 8: Om det regnar, blir gräsmattan blöt.
Det regnar.
Gräsmattan blir blöt.

Detta är ett exempel på en giltig slutledning. Varför är den giltig?

Lösning: Vi kan analysera slutledningen satslogiskt och ser då att den har följande logiska struktur

$$\begin{array}{l} A \rightarrow B \\ \underline{A \qquad\qquad} \\ B \end{array}$$

där A står för satsen Det regnar, och
B för satsen Gräsmattan blir blöt.

Detta är en formell slutledning, och denna kan analyseras med hjälp av satslogikens sanningsvärdestabeller. Om sanningstabeller kan man läsa i avsnittet om satslogik.

A B	$A \to B$	A	B
s s	s	s	s
s f	f	s	f
f s	s	f	s
f f	s	f	f

I alla sanningsvärdestilldelningar som gör båda premisserna sanna är också slutsatsen sann (rad 1 i tabellen). Vi säger då att den formella slutledningen är *satslogiskt giltig* och därmed säger vi också att den informella slutledningen är satslogiskt giltig.

Om slutledningen

$$\begin{array}{l} A_1 \\ A_2 \\ \vdots \\ \underline{A_n} \\ B \end{array}$$

är giltig, skriver vi $\{A_1, A_2, \ldots , A_n\} \models B$.

Observera att begreppet *giltig slutledning* är semantiskt.

Exempel 9: Om solen skiner, blir bonden glad.
Bonden blir glad.
Solen skiner.

Är denna informella slutledningen giltig?

Lösning: Formalisering ger med uppenbara beteckningar

$$\begin{array}{l} A \to B \\ \underline{B} \\ A \end{array}$$

Analys med hjälp av sanningstabeller ger

A B	$A \to B$	B	A
s s	s	s	s
s f	f	f	s
f s	s	s	f
f f	s	f	f

Här finns en sanningsvärdestilldelning som gör båda premisserna sanna och slutsatsen falsk (rad 3 i tabellen). Den formella slutledningen, och därmed även den informella, är då satslogiskt ogiltig.

En syntaktisk infallsvinkel skulle vara att ge en uppsättning regler med hjälp av vilka vi kan härleda en slutsats från en uppsättning premisser, som då skulle fungera som en sorts tillfälliga axiom i härledningen. En regel skulle till exempel kunna vara att alla slutledningar på formen

$$\frac{\begin{array}{l}\varphi \rightarrow \psi \\ \varphi\end{array}}{\psi}$$

är korrekta (φ och ψ är godtyckliga satser). Om vi, givet en uppsättning regler, kan härleda satsen ψ från satserna $\varphi_1, \varphi_2, \ldots, \varphi_n$, så skriver vi $\{\varphi_1, \varphi_2, \ldots, \varphi_n\} \vdash \psi$. Observera återigen att *härledning* är ett syntaktiskt begrepp.

Det återstår nu att undersöka huruvida härledningsrelationen ($\vdash$) och medförrelationen ($\models$) är *extensionellt ekvivalenta*, dvs genererar precis samma uppsättning giltiga slutledningar. Logik handlar mycket om hur relationerna ”$\vdash$” och ”$\models$” hänger ihop inom olika typer av logiska system.

Vår uppgift blir att konstruera syntaktiska system, som förhoppningsvis kan manipuleras av en dator, och som sammanfaller med vår intuition av semantiska system.

Objektspråk och metaspråk

När man studerar språk, och det är väsentligen det vi gör i detta kompendium, är det viktigt att hålla isär det språk man studerar, det språk man talar om, *objektspråket*, och det språk man använder när man talar om objektspråket, *metaspråket*. Vi kan till exempel använda svenska som metaspråk vid studier av objektspråket engelska. Vid diskussioner om naturliga språk, så använder man ofta samma språk som objekt- och metaspråk. Vanskligheter i detta syns om vi betraktar följande två satser, som på ytan ser ut att ha samma struktur. Fido är tänkt att vara en hund.

(1) Fido har fyra ben.
(2) Fido har fyra bokstäver.

Här är det uppenbart att ordet ”Fido” i de två satserna har olika betydelse. I (1) refererar det till hunden, och i (2) till hundens namn. Vi kan tydliggöra detta på ett standardmässigt sätt genom att använda citationstecken enligt

(2’) ”Fido” har fyra bokstäver.

Det gäller att (1) hör till objektspråket, och (2) och (2’) hör till metaspråket.

För att illustrera med det ovan diskuterade MIU-systemet, så är

MI

ett uttryck som hör till objektspråket, och

MI är ett teorem
Mxx

är uttryck som hör till metaspråket.

I framställningen nedan kommer vi ibland att arbeta i respektive objektspråk, och ibland i ett metaspråk. Vi kommer att använda olika typer av symboler i objekt- respektive metaspråk, så förväxlingar ska egentligen inte vara möjliga.

Några grekiska bokstäver

Som skymtat ovan kommer vi att använda grekiska bokstäver i vår presentation.

Gemener	Versaler	Namn
φ	Φ	fi
ψ	Ψ	psi
θ	Θ	theta
π	Π	pi
σ	Σ	sigma
γ	Γ	gamma

Vi kommer normalt att använda små grekiska bokstäver för formler och satser, och stora för mängder av formler och satser. De kommer inte att tillhöra respektive objektspråk, utan de kommer att användas i ett metaspråk när vi refererar till formler i ett objektspråk.

2 SATSLOGIK

Det satslogiska språket

För att definiera ett formellt språk måste man specificera

1. vilka symboler som ingår i språket
2. ange hur komplicerade formler byggs upp från enklare.

För vårt satslogiska språk skall följande gälla.

Definition 1: 1. *symboler*

Logiska symboler: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \bot$
Markörer: (,)
Satsparametrar: $A_1, A_2, A_3, \ldots$

2. *Välbildade formler (vbf)*

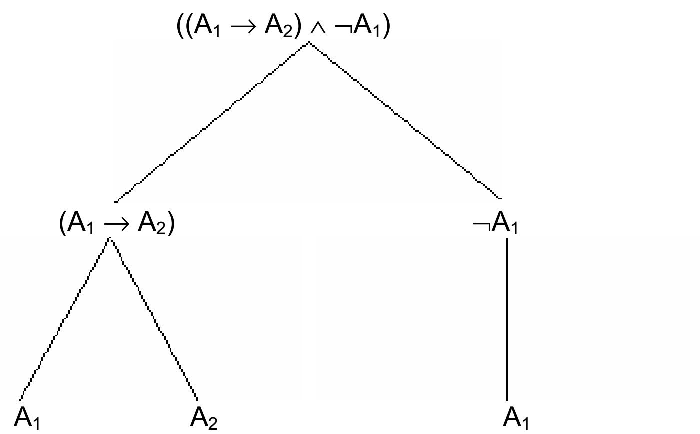
Vi specificerar vilka formler som är välbildade med en induktiv (rekursiv) definition.

a. Alla satsparametrar samt $\bot$ är vbf.
b. Om φ är en vbf, så är $\neg\varphi$ en vbf.
c. Om φ och ψ är vbf, så är $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$ och $(\varphi \leftrightarrow \psi)$ vbf.

Den vokabulär $L = \{\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \bot, (,), A_1, A_2, A_3, \ldots\}$ som specificerats ovan innehåller oändligt många symboler. Vill vi att vokabulären skall innehålla ändligt många symboler kan vi förfara på följande sätt med satsparametrarna. Vi använder oss av två symboler, A och ’, och låter satsparametrarna vara A’, A’’, A’’’, A’’’’ och så vidare. På detta sätt kan vi generera oändligt många satsparametrar med endast två symboler, och skulle då kunna erhålla en ändlig vokabulär. Ett *uttryck* över vår vokabulär L är en ändlig sträng av symboler från L. Med definitionen av vbf ovan skiljer vi ut de välbildade strängarna bland uttrycken. Observera att φ och ψ inte är symboler i vokabulären. De har karaktären av metasymboler, och skall i sin tur ersättas med strängar över vokabulären. Notera också att det går att mekaniskt avgöra huruvida ett uttryck är en välbildad formel eller ej.

Exempel 1: Är $((A_1 \rightarrow A_2) \wedge \neg A_1)$ en vbf?

Lösning: Vi konstaterar inledningsvis att strängen har formen $(\varphi \wedge \psi)$ där $\varphi = (A_1 \rightarrow A_2)$ och $\psi = \neg A_1$. Därefter konstaterar vi att $(A_1 \rightarrow A_2)$ är på formen $(\varphi \rightarrow \psi)$ där $\varphi = A_1$ och $\psi = A_2$. Eftersom A_1 och A_2 är satsparametrar, så är de vbf enligt 2a i definition 1, och därmed är även $(A_1 \rightarrow A_2)$ vbf enligt 2c. Vad slutligen beträffar $\neg A_1$ så är den på formen $\neg\varphi$, där $\varphi = A_1$, och eftersom A_1 är en satsparameter, och därmed välbildad enligt 2a, så är också $\neg A_1$ välbildad enligt 2b. Vi har konstaterat att både $(A_1 \rightarrow A_2)$ och $\neg A_1$ är vbf. Då är också $((A_1 \rightarrow A_2) \wedge \neg A_1)$ en vbf. Ovanstående analys utförs enklast med hjälp av träd som illustrerar hur formler byggs upp.



Vi kan här konstatera att formlerna i löven är vbf, eftersom de är satsparametrar, och arbetar oss sedan successivt upp i trädet.

Är $(A_1 \rightarrow A_2) \wedge \neg A_1$ en vbf? Svaret här är ”nej” eftersom yttre parentespar saknas. Av praktiska skäl skall vi emellertid undvika att skriva ”onödiga” parenteser, varför formeln i exempel 1 kommer att skrivas på ovanstående sätt, och vi uppfattar de saknade parenteserna som underförstådda. Vi inför dessutom följande parenteskonventioner.

Parenteskonventioner: $\neg$ binder starkare än $\wedge$ och $\vee$, som i sin tur binder starkare än $\rightarrow$ och $\leftrightarrow$.

Exempel 2: Formeln $\neg A_1 \wedge A_3 \rightarrow A_5 \vee A_2$ är en förkortning av $((\neg A_1 \wedge A_3) \rightarrow (A_5 \vee A_2))$.

För att slippa skriva indicerade variabler, kommer vi dessutom i fortsättningen ofta använda andra versaler som satsparametrar.

Konjunktion och disjunktion är associativa operationer, dvs $(A \,¤\, B) \,¤\, C \Leftrightarrow A \,¤\, (B \,¤\, C)$, där ¤ är någon av operatorerna $\wedge$ eller $\vee$. Detta innebär att vi kan utelämna parenteser och skriva $A \,¤\, B \,¤\, C$ i stället för till exempel $(A \,¤\, B) \,¤\, C$.

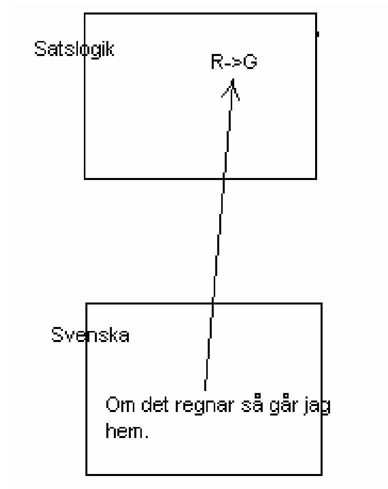
Formalisering

Med en satslogisk formalisering uppvisar vi den satslogiska formen hos en sats i ett naturligt språk.

Grovt svarar de satslogiska konnektiven mot (har avsedda tolkning) följande språkliga uttryck inklusive dess synonymer.

Konnektiver	Språklig motsvarighet	Namn
$\neg$	inte, det är inte så att …	negation
$\wedge$	och, men	konjunktion
$\vee$	Eller (icke uteslutande)	disjunktion
$\rightarrow$	om …, så …	(materiell) implikation
$\leftrightarrow$	… om och endast om	(materiell) ekvivalens

Vid en formalisering ersätts atomära språkliga satser med satsparametrar, och den språkliga satsens logiska form skall uppvisas så troget som möjligt.



Vi gör följande försök till definition.

Definition 2: En *enkel* sats (*atomär* sats) är en fullständig påståendesats, som inte innehåller någon motsvarighet i naturligt språk till något satslogiskt konnektiv. *Sammansatta* satser (*molekylära* satser) byggs upp av atomära satser och språkliga motsvarigheter till konnektiv.

Observera att vi nu använder ”atomär” respektive ”molekylär” sats (formel) både om satser i naturliga språk och om satslogiska formler. Inget missförstånd torde kunna inträffa på grund av detta. Vi ger nu en uppsättning exempel på satslogisk formalisering.

Exempel 3: Formalisera satsen Per och Pål är starka.

Lösning: Denna sats kan uppfattas som en förkortning av den längre satsen

Per är stark och Pål är stark

där vi för att få ett naturliga flyt i språket inte upprepar subjektet andra gången det dyker upp. Vi kan då identifiera två atomära satser och sätter

A: Per är stark,
B: Pål är stark.

Vi får formaliseringen

$$A \wedge B.$$

Nu är detta inte hela sanningen, för satsen är inte entydig. Vi har ovan valt att läsa den så att båda individerna har egenskapen att vara stark. Då binder ”och” ihop två satser. Ett annat sätt att läsa satsen är att säga att den betyder, att Per och Pål är starka tillsammans. Ordet ”och” binder då ihop de två subjekten till ett kollektiv. I detta fall är det inte frågan om satslogikens konjunktion, varför satsen i denna tolkning måste uppfattas som atomär.

Exempel 4: Formalisera satsen

Steken skall kryddas med svartpeppar och vitpeppar eller kryddpeppar.

Lösning: Vi identifierar de atomära satserna och sätter

S: Steken skall kryddas med svartpeppar,
V: Steken skall kryddas med vitpeppar,
K: Steken skall kryddas med kryddpeppar.

När vi nu ska rekonstruera satsen finner vi två möjligheter.

1. $S \wedge (V \vee K)$
2. $(S \wedge V) \vee K$

De två satslogiska formlerna svarar mot två olika läsningar av satsen, som inte är entydig. Mångtydigheten i satsen är en så kallad strukturell mångtydighet. Detta kan vi jämföra med mångtydigheten hos satsen i föregående exempel, där mångtydigheten berodde på mångtydigheten hos ordet ”och”.

Det kan vara intressant att reflektera över att en automatiserad språköversättare skall klara av de mångtydigheter som finns i exempel 2 och 3. Beroende på kontext (sammanhang) skall språköversättaren kunna välja vilken av de två innebörderna som avses i det sammanhang meningen står. Det är inte alltför långsökt att hävda att (stark) AI väsentligen har med semantik att göra. Språket är kanske det mest specifikt mänskliga, och då är det kanske inte så långsökt att tänka att när en dator kan simulera mänskligt språk, kan den också simulera vissa andra mänskliga egenskaper.

Exempel 5: Formalisera satsen

Jag kan varken spela piano eller orgel, men jag äger ett piano.

Lösning: Här har vi att försöka formalisera frasen ”varken … eller …”. Ett sätt att klara detta är att försöka skriva om satsen så att uttrycket ”varken … eller …” inte förekommer, men förstås så att satsen bibehåller sin mening. Ett försök är följande:

Jag kan inte spela piano och jag kan inte spela orgel, men jag äger ett piano.

Vi får då följande atomära satser:

P: Jag kan spela piano
O: Jag kan spela orgel
A: Jag äger ett piano

och formaliseringen blir

$$\neg P \wedge \neg O \wedge A,$$

där vi som påpekats ovan kan utelämna parenteser.

Exempel 6: Formalisera satsen

Du skall ha svart kostym, annars får du inte leka med oss.

Lösning: Det besvärliga uttrycket här är ”annars”. Detta ord signalerar att det är ett nödvändigt villkor att man har svart kostym för att man skall få deltaga i leken. Vi kan uttrycka samma sak på följande sätt:

Om du inte har svart kostym, så får du inte leka med oss.

Denna sats är enkel att formalisera, och vi får med de enkla satserna

S: Du har svart kostym
L: Du får leka med oss

formeln

$\neg S \rightarrow \neg L$

Exempel 7: Formalisera satsen

Vaktposten får skjuta, endast om han ropat halt.

Lösning: Denna sats innehåller samma typ av konstruktion som i föregående exempel, men i andra språkliga kläder. Här sägs att det är nödvändigt att vaktposten ropar halt för att han skall få skjuta. Enkla satser är

H: Vaktposten har ropat halt
S: Vaktposten får skjuta

Vi får formeln

$\neg H \rightarrow \neg S$.

Vad skulle kunna hända om en vaktpost händelsevis skulle tro att satsen betydde samma som

Vaktposten får skjuta, om han ropat halt,

respektive

Vaktposten får skjuta om och endast om han ropat halt?

Vid formalisering finns väsentligen två strategier, som vi kan kalla nerifrån-och-upp (bottom up parsing) respektive uppifrån-och-ner (top down parsing). Den metod vi använt ovan är nerifrån och upp, och denna metod går ut på att vi först identifierar de enkla satserna (botten) och därefter rekonstruerar hur den givna satsen är uppbyggd (toppen). Den andra strategien går ut på att vi först analyserar vilket den givna satsens huvudkonnektiv är. Satsen sönderfaller då i två delsatser. Vi upprepar sedan denna procedur på dessa två delsatser tills bara enkla satser återstår. Vi har då ett rekursivt förfarande. Oftast är en kombination av de två strategierna lämplig, därför att det inledningsvis kan vara svårt att se vilka de enkla satserna är, medan det kan vara lätt att se detta när man kommit något steg ner i satsen. Vi avslutar detta avsnitt med två lite svårare exempel, i vilka vi utnyttjar den senare tekniken. Denna teknik kommer vi sedan att flitigt utnyttja när vi skall formalisera satser i predikatlogik.

Exempel 8: Formalisera satsen

Både ifall frankeringen är otillräcklig och ifall adressen är ofullständig, så återsänds brevet till avsändaren, förutsatt att det finns avsändarbeteckning.

Lösning: Inledningsvis konstaterar vi att ”förutsatt” är huvudkonnektiv, och satsen kan då skrivas

(Det finns avsändarbeteckning) $\rightarrow$ (Både ifall frankeringen är otillräcklig och ifall adressen är ofullständig, så återsänds brevet till avsändaren)

Nu är försatsen enkel. I eftersatsen kan ”och” uppfattas som huvudkonnektiv. Observera att det finns fler möjligheter här. Vi får då

(Det finns avsändarbeteckning) $\rightarrow$ ((Om frankeringen är otillräcklig, så återsänds brevet till avsändaren) $\wedge$ (Om adressen är ofullständig, så återsänds brevet till avsändaren))

Här har båda de två nya delsatserna ”om … så ..” som huvudkonnektiv. Detta ger oss

(Det finns avsändarbeteckning) $\rightarrow$ (((Frankeringen är otillräcklig) $\rightarrow$ (Brevet återsänds till avsändaren)) $\wedge$ ((Adressen är ofullständig) $\rightarrow$ (Brevet återsänds till avsändaren)))

Två av de uppkomna satserna har ”inte” som huvudkonnektiv, medan de övriga är enkla, och vi får slutligen:

(Det finns avsändarbeteckning) $\rightarrow$ ((($\neg$(Frankeringen är tillräcklig)) $\rightarrow$ (Brevet återsänds till avsändaren)) $\wedge$ (($\neg$(Adressen är fullständig)) $\rightarrow$ (Brevet återsänds till avsändaren)))

Ersätt nu de atomära satserna med satsparametrar.

A: Det finns avsändarbeteckning.
B: Frankeringen är tillräcklig.
C: Brevet återsänds till avsändaren.
D: Adressen är fullständig.

Vi uppvisar slutligen den givna satsens satslogiska struktur.

$$A \rightarrow ((\neg B \rightarrow C) \wedge (\neg D \rightarrow C))$$

Exempel 9: Formalisera satsen

Ett nödvändigt villkor för att vakten skall få använda våld är att risk för skadegörelse föreligger om han inte ingriper, och förutsatt att han inte är angripen, gäller dessutom att han får använda våld endast om han först givit muntlig varning; att han är angripen är däremot en tillräcklig förutsättning för att han skall få använda våld.

Lösning: Satsen är en konjunktion bestående av tre konjunkter:

(1) Ett nödvändigt villkor för att vakten skall få använda våld är att risk för skadegörelse föreligger om han inte ingriper.
(2) Förutsatt att vakten inte är angripen, gäller dessutom att han får använda våld endast om han först givit muntlig varning.

(3) Att vakten är angripen är däremot en tillräcklig förutsättning för att han skall få använda våld.

(1) I denna sats utsägs att

> om det inte är så att (risk för skadegörelse föreligger om vakten inte ingriper), så får vakten inte använda våld.

Frasen inom parentesen är nödvändigt villkor för att vakten skall få använda våld. Frasen ”det är inte så att” negerar uttrycket inom parentesen.

Vi inför följande satsparametrar

A: Vakten får använda våld.
B: Risk för skadegörelse föreligger.
C: Vakten ingriper.

och får $\neg(\neg C \rightarrow B) \rightarrow \neg A.$

(2) Denna sats är en implikation med församlingsats

> vakten är inte angripen,

och med eftersats

> vakten får använda våld endast om han först givit muntlig varning.

I eftersatsen är den muntliga varningen ett nödvändigt villkor för att vakten skall få använda våld.

Inför vi ytterligare satsparametrar för de nya atomära satserna

D: Vakten är angripen.
E: Vakten har givit muntlig varning.

får vi $\neg D \rightarrow (\neg E \rightarrow \neg A).$

(3) är slutligen en vanlig implikation och vi får

$$D \rightarrow A.$$

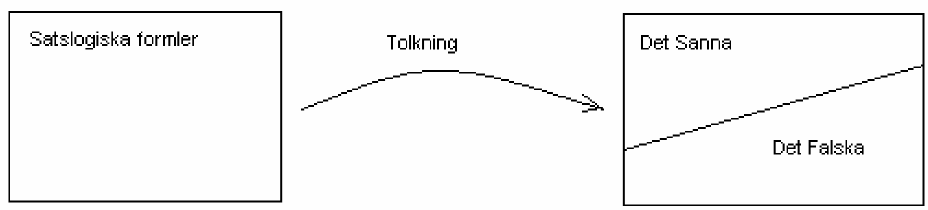
Till sist binder vi ihop de tre konjunkterna och erhåller följande formalisering av den givna satsen

$$(\neg(\neg C \rightarrow B) \rightarrow \neg A) \wedge (\neg D \rightarrow (\neg E \rightarrow \neg A)) \wedge (D \rightarrow A).$$

Sanningstabeller

Vi presenterar i detta avsnitt satslogikens semantik och några definitioner på ett intuitivt sätt, för att återkomma till en mer precis framställning.

Vi noterar först att när vi skall definiera de satslogiska konnektivens mening är det två egenskaper vi är intresserade av. Den ena är att den enda egenskapen vi är intresserade av hos en sats är att den har ett sanningsvärde. Den ”verklighet” som ”svarar mot” det satslogiska språket består av ”Det Sanna” och ”Det Falska”. Vår satslogik är tvåvärd.



Den andra idén är att konnektiven skall vara funktionella, dvs varje formels sanningsvärde skall vara entydigt bestämd av de ingående enkla satsernas sanningsvärde, och varje formel skall ha ett sanningsvärde. Vi slår fast konnektivens mening (tolkning) med hjälp av nedanstående tabell, som är att uppfatta som en definition.

A B	$A \wedge B$	$A \vee B$	$A \rightarrow B$	$A \leftrightarrow B$	$\neg A$
S S	S	S	S	S	F
S F	F	S	F	F	F
F S	F	S	S	F	S
F F	F	F	S	S	S

Symbolen $\perp$, *falsum* eller det falska, antar alltid sanningsvärdet F. Av tabellen ovan framgår att ”$\vee$” svarar på något sätt mot icke-uteslutande ”eller”. Den andra svårigheten i definitionen är varför vi väljer ”S” för materiell implikation i det fall då försatsen är falsk. Vi kan först konstatera att ”om …, så …” har flera betydelser, och att vi bara kan välja en av dem. Ofta används ”om …, så …” när sambandet mellan två satser är starkare än det vi vill komma åt ovan. En variant är att försatsens sanning skall generera eftersatsens sanning. Denna användning kommer vi att benämna *logisk följd* och vi återkommer till denna. Om implikationen finns en vidlyftig litteratur. Varför väljer vi då att definiera implikationen som ovan?

Vi har följande möjliga definitioner av materiell implikation, om vi vill behålla funktionalitet

A B	$A \rightarrow B$ Alt 1	$A \rightarrow B$ Alt 2	$A \rightarrow B$ Alt 3	$A \rightarrow B$ Alt 4	Kommentar
S S	S	S	S	S	OK
S F	F	F	F	F	OK
F S	S	F	F	S	
F F	S	S	F	F	

Vårt val är alternativ 1. Vi ser att alternativ 2 är identisk med definitionen av materiell ekvivalens, alternativ 3 är identisk med definitionen av konjunktionen och alternativ 4 är precis de sanningsvärden vi tilldelat B. Inget av alternativen 2 – 4 är alltså ens rimliga.

Vi vill också i matematik kunna bevisa till exempel att den tomma mängden är delmängd till varje mängd, dvs att $\emptyset \subseteq A$ för varje mängd A. Vi har då att visa att

$$a \in \emptyset \rightarrow a \in A \text{ är sann för godtyckligt element } a.$$

Ovanstående påstående är sant, eftersom försatsen $a \in \emptyset$ är falsk då ju a inte kan vara element i tomma mängden.

Med vårt val av definition av materiell implikation gäller det också att

$$\varphi \leftrightarrow \psi \;\Leftrightarrow\; (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi),$$

där $\Leftrightarrow$ betecknar *logisk ekvivalens* (se definition 4). Undersök för vilka av de andra möjliga definitionerna av $\rightarrow$, som ovanstående logisk ekvivalens gäller.

Definition 3: En formel φ är en *tautologi* om och endast om φ är sann i varje sanningsvärdestilldelning. φ är en *kontradiktion* om och endast om φ är falsk i varje sanningsvärdestilldelning. φ är *kontingent* om och endast om φ varken är en tautologi eller en kontradiktion.

Tabellmetoden är en mekanisk metod med hjälp av vilken man kan avgöra vilket av de tre fallen som föreligger. Vi förutsätter i texten nedan att metoden är känd, men ger för fullständighetsskäl två exempel på dess användning.

Exempel 10: Konstruera en sanningstabell för formeln
$A \vee \neg B \;\rightarrow\; (\neg A \;\rightarrow\; C)$.

Lösning: Eftersom formeln innehåller tre satsparametrar, finns $2^3 = 8$ möjliga sanningsvärdestilldelningar på parametrarna, och vi får följande tabell

A	B	C	$A \vee \neg B \rightarrow (\neg A \rightarrow C)$
S	S	S	S S F S F S S
S	S	F	S S F S F S F
S	F	S	S S S S F S S
S	F	F	S S S S F S F
F	S	S	F F F S S S S
F	S	F	F F F S S F F
F	F	S	F S S S S S S
F	F	F	F S S F S F F
			2 1 5 3 4

Siffrorna I den nedersta raden indikerar i vilken ordning formeln har utvärderats. Vi noterar att formeln är kontingent.

Exempel 11: Konstruera en sanningstabell för formeln
$((\neg A \vee B) \wedge C) \to \neg B$.

Lösning: Som i exemplet ovan indikerar sista radens siffror den ordning i vilken respektive sanningsvärde räknats ut.

A	B	C	$((\neg A \vee B) \wedge C) \to \neg B$
S	S	S	F S S S S F F
S	S	F	F S S F F S F
S	F	S	F F F F S S S
S	F	F	F F F F F S S
F	S	S	S S S S S F F
F	S	F	S S S F F S F
F	F	S	S S F S S S S
F	F	F	S S F F F S S
			1 2 3 5 4

Vi noterar att formeln är kontingent.

Många problem i satslogik återförs på problemet att avgöra huruvida en given formel är en tautologi eller ej. Vid stora problem, dvs problem i vilka många satsparametrar förekommer är metoden ohanterlig. Med n st satsparametrar så innehåller tabellen 2^n rader, vilket innebär att metoden har tidskomplexitet $O(2^n)$. Det finns ingen känd metod med bättre tidskomplexitet. Det är ett av den moderna matematikens, logikens och datalogins viktigaste problem att avgöra huruvida det finns någon väsentligen effektivare metod för att avgöra huruvida en given formel är en tautologi eller ej. Med ”väsentligen effektivare” menar jag en algoritm med komplexitet $O(n^p)$, dvs polynomiell komplexitet, där n är inputstorlek och p är en konstant.

Ibland kan man komma lindrigare undan vad gäller arbete om man räknar ”baklänges” i tabellerna. Vi kallar metoden för *snabbmetoden*. Det man då gör är ett motsägelsebevis. Vi illustrerar tekniken med några exempel.

Exempel 12: Visa att $\varphi = (A \to B) \vee (B \to C)$ är en tautologi. Här är φ tänkt att vara namn på den givna formeln, och likhetstecknet betyder likhet mellan formler.

Lösning: Antag för motsägelse att φ inte är en tautologi. Då finns en sanningsvärdestilldelning i vilken φ är falsk. En icke fullständig sanningsvärdestabell, där vi fokuserar på just situationen att φ är falsk skulle kunna se ut som nedan.

A B C	$(A \to B) \vee (B \to C)$
	S_2 F_1 F_2 F S_2 F_1 F_2

Det F som inte är indicerat är det F vi får från antagandet. De två förekomsterna av F_1 följer av att disjunkterna i en falsk disjunktion måste vara falska . Slutligen följer de sanningsvärdena som är indicerade med en tvåa av den materiella implikationens sanningsvillkor. Vi ser att satsparameter B måste vara både sann och falsk om formeln är falsk. Detta är en motsägelse och den givna formeln måste därmed vara en tautologi. Indexen på sanningsvärdena i tabellen ovan är tänkta att illustrera den ordning i vilka sanningsvärdena är framräknade.

Definition 4: Två formler φ och ψ är *satslogiskt ekvivalenta*, $\varphi \Leftrightarrow \psi$, om och endast om formeln $\varphi \leftrightarrow \psi$ är en tautologi.

Vi ser här ett första problem där vi kan lösa problemet genom att återföra det på att avgöra huruvida en formel är en tautologi.

Exempel 13: Visa att $A \rightarrow B \Leftrightarrow \neg A \vee B$.

Lösning: Observera att det inte behövs några parenteser i uttrycket ovan, eftersom det är två formler som är relaterade till varandra via relationen logisk ekvivalens. Symbolen $\Leftrightarrow$ hör inte till det satslogiska språket. Vi har enligt definitionen att visa att formeln $(A \rightarrow B) \leftrightarrow (\neg A \vee B)$ är en tautologi. Observera att vi här måste sätta ut parenteser eftersom detta är en formel, och vi måste då med parenteser ange formelns uppbyggnad. Vi använder snabbmetoden och antar för motsägelse att formeln inte är en tautologi. Då finns en sanningsvärdestilldelning som gör formeln falsk. Vi försöker illustrera räkningarna i en tabell.

A B	$(A \rightarrow B) \leftrightarrow (\neg A \vee B)$	
S_3 F_3	S_1 F F_2 F_1 F_2	Fall 1
S_3 F_3	S_2 F_1 F_2 F S_1	Fall 2

Här finns två möjligheter. I fall 1, så är implikationen sann och disjunktionen falsk. Eftersom disjunktionen är falsk, så måste båda disjunkterna vara falska. Då är A sann och B falsk, varför implikationen är falsk, och vi har en motsägelse. Enligt den andra möjligheten är disjunktionen sann och implikationen falsk. Eftersom implikationen är falsk, så måste A vara sann och B vara falsk, men då är disjunktionen falsk, och vi får en motsägelse även i detta fall. Eftersom båda fallen leder till motsägelse, så har vi en motsägelse och därmed är det klart att formeln är en tautologi. Självklart tjänar vi väldigt lite tid genom att använda ”snabbmetoden” här. Vi gör det enbart för att illustrera tekniken. Redovisningen i tabell ovan är inte särskilt lyckad, och vi skall se nedan hur detta kan göras på ett bättre sätt.

Definition 5: Låt $\varphi_1, \varphi_2, \dots, \varphi_n$ och ψ vara formler. ψ är en *satslogisk följd (konsekvens)* av $\varphi_1, \varphi_2, \dots, \varphi_n$ om och endast om $\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \rightarrow \psi$ är en tautologi. Vi skriver då $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$. Om ψ inte är en satslogisk följd av $\varphi_1, \varphi_2, \dots, \varphi_n$ skriver vi $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \not\models \psi$.

Här har vi ytterligare ett problem vi kan lösa genom att återför det på att avgöra huruvida en formel är en tautologi.

Exempel 14: Visa att $\{A \rightarrow B, B \rightarrow C\} \models A \rightarrow C$.

Lösning: Vi ska visa att $((A \rightarrow B) \wedge (B \rightarrow C)) \rightarrow (A \rightarrow C)$ är en tautologi. Antag för motsägelse att formeln inte är en tautologi. Då finns en sanningsvärdes-tilldelning sådan att formeln är falsk. Vi illustrerar återigen beräkningarna i en tabell.

A	B	C	((A	$\rightarrow$	B)	$\wedge$	(B	$\rightarrow$	C))	$\rightarrow$	(A	$\rightarrow$	C)
S_3		F_3		S_4	S_6	S_1	F_5	S_4		F	S_2	F_1	F_2

Vi noterar att B måste vara både sann och falsk, vilket är orimligt. Då är $((A \rightarrow B) \wedge (B \rightarrow C)) \rightarrow (A \rightarrow C)$ en tautologi, och därmed är det klart att $\{A \rightarrow B, B \rightarrow C\} \models A \rightarrow C$.

Satslogikens semantik formaliserad

För att kunna bevisa påståenden om ett satslogiskt språk, måste dess semantik vara precist formulerad. Denna precisa formulering skall förstås stämma överens med de intuitioner vi presenterat ovan. Vi förfar på följande sätt.

Definition 6: Låt VBF = $\{\varphi : \varphi$ är en vbf i satslogik$\}$.

En *värdering* V är en funktion

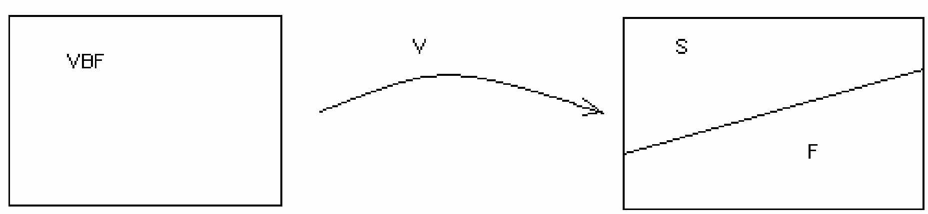
$$V : VBF \rightarrow \{S, F\}$$

som uppfyller följande villkor

(i) $V(\neg\varphi) = S$ om och endast om $V(\varphi) = F$,
(ii) $V(\varphi \wedge \psi) = S$ om och endast om $V(\varphi) = V(\psi) = S$,
(iii) $V(\varphi \vee \psi) = S$ om och endast om $V(\varphi) = S$ eller $V(\psi) = S$,
(iv) $V(\varphi \rightarrow \psi) = S$ om och endast om $V(\varphi) = F$ eller $V(\psi) = S$,
(v) $V(\varphi \leftrightarrow \psi) = S$ om och endast om $V(\varphi) = V(\psi)$.
(vi) $V(\perp) = F$

Vi skriver ibland $V \models \varphi$ om $V(\varphi) = S$, och $V \not\models \varphi$ om $V(\varphi) = F$.

En värdering tilldelar alltså varje formel ett sanningsvärde.



Exempel 15: Beräkna $V(A_1 \rightarrow (A_2 \wedge \neg A_7))$ då $V(A_1) = S$ och $V(A_i) = F$ för $i > 1$.

Lösning: Vi får
$V(\neg A_7) = S$
$V(A_2 \wedge \neg A_7) = F$, och slutligen
$V(A_1 \rightarrow (A_2 \wedge \neg A_7)) = F$.

I stort sett svarar en värdering mot en rad i en sanningsvärdestabell. Skillnaden är att en rad i en sanningsvärdestabell bara tilldelar ett sanningsvärde till vissa satsparametrar, och därmed till vissa formler, medan en värdering tilldelar alla satsparametrar och formler ett sanningsvärde.

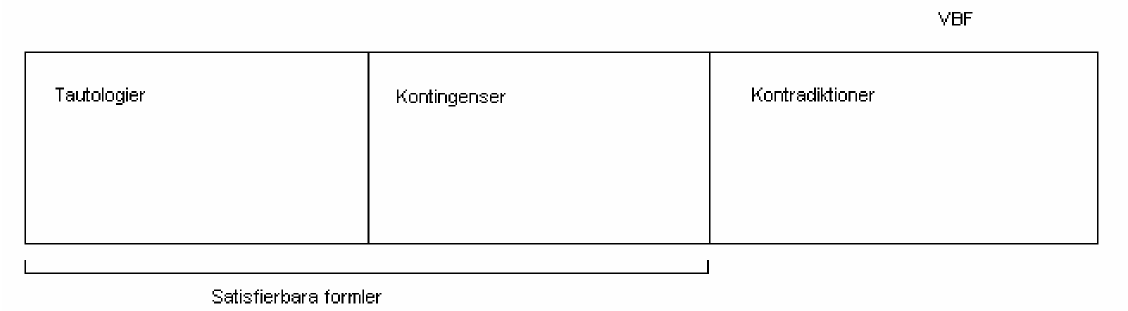
Vi kan nu precisera våra tidigare definitioner.

Definition 7: Låt $\varphi \in$ VBF. Vi säger att φ är en *tautologi* (satslogisk sanning) om och endast om $V(\varphi) = S$ för varje värdering V. Vi skriver då $\models \varphi$, och vi skriver $\not\models \varphi$ om φ inte är en tautologi. Vi säger att φ är en *kontradiktion* om och endast om $V(\varphi) = F$ för varje värdering V. Vi säger att φ är *kontingent* om och endast om φ varken är en tautologi eller en kontradiktion.

Vi inför också följande, nya, begrepp.

Definition 8: Låt $\varphi \in$ VBF. Vi säger att φ är *satisfierbar* om och endast om $V(\varphi) = S$ för någon värdering V.

Vi får då följande bild av hur satslogiska formler kan klassificeras.



Problemet att avgöra till vilken kategori en formel hör är avgörbart, dvs det finns en mekanisk metod med hjälp av vilken man kan lösa problemet. Metoden är i det här fallet tabell-metoden, men observera att denna metod har exponentiell komplexitet, varför den egentligen är ohanterlig för stora problem, problem med många satsparametrar. Som nämnts ovan är det ett viktigt problem inom logik, matematik och datalogi att undersöka huruvida det finns snabbare algoritmer för detta problem eller ej.

Exempel 16a: Visa att $\models A \to (B \to A)$.

Lösning*:* Antag för motsägelse att $\not\models A \to (B \to A)$.

Då finns en värdering V sådan att $V(A \to (B \to A)) = F$, dvs

(1) $V(A) = S$ och
(2) $V(B \to A) = F$.

Enligt (2), måste
(3) $V(A) = F$,

men då har vi en motsägelse, eftersom (1) och (3) motsäger varandra, och det följer att

$\models A \to (B \to A)$.

I exemplet ovan har vi använt oss av snabbmetoden, och här har vi dessutom en redovisning som är klar och tydlig till skillnad från de röriga tabellerna vi använt i exemplen 12 – 14. En normal taktik, när man använder snabbmetoden, är att spåna med tekniken i exempel 12 – 14, och för en eventuell redovisning av problemet så använder man tekniken i exempel 16. Notera också att formeln i exemplet ovan visserligen bara innehåller satsparametrar, men att det inte finns något i resonemanget som förutsätter detta, så A och B skulle lika gärna kunna vara godtyckliga formler. Vi illustrerar med samma exempel en gång till, men använder ett annat framställningssätt

Exempel 16b: Visa att $\models \varphi \to (\psi \to \varphi)$.

Lösning: Antag för motsägelse att $\not\models \varphi \to (\psi \to \varphi)$.

Då finns en värdering V sådan att $V \not\models \varphi \to (\psi \to \varphi)$, dvs

(1) $V \models \varphi$ och
(2) $V \not\models \psi \to \varphi$.

Enligt (2), måste

(3) $V \not\models \varphi$,

men då har vi en motsägelse, eftersom (1) och (3) motsäger varandra.

$\therefore \models \varphi \to (\psi \to \varphi)$.

Ovanstående teknik att presentera lösningar kommer att vara viktig längre fram i framställningen, varför vi rekommenderar läsaren att flitigt träna detta.

Exempel 17: Visa att $\models (\varphi \land \psi \to \theta) \to (\varphi \to (\psi \to \theta))$.

Lösning: Antag för motsägelse att $\not\models (\varphi \land \psi \to \theta) \to (\varphi \to (\psi \to \theta))$.

Då finns en värdering V sådan att $V \not\models (\varphi \land \psi \to \theta) \to (\varphi \to (\psi \to \theta))$, dvs
(1) $V \models \varphi \land \psi \to \theta$ och
(2) $V \not\models \varphi \to (\psi \to \theta)$.

På grund av (2), måste

(3) $V \models \varphi$ och
(4) $V \not\models \psi \to \theta$, som i sin tur ger
(5) $V \models \psi$ och
(6) $V \not\models \theta$.

Men då ger (3), (5) och (6) att

$V \not\models \varphi \land \psi \to \theta$, som motsäger (1), och slutsatsen blir att

$\models (\varphi \land \psi \to \theta) \to (\varphi \to (\psi \to \theta))$.

Definition 9: Låt $\varphi_1, \varphi_2, \dots, \varphi_n$ och ψ vara välbildade formler. Vi säger att ψ är en *satslogisk följd* av $\varphi_1, \varphi_2, \dots, \varphi_n$, om för varje värdering V sådan att $V(\varphi_1) = V(\varphi_2) = \dots = V(\varphi_n) = S$ det gäller att $V(\psi) = S$. Vi skriver då $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$. Om ψ inte är en satslogisk följd av $\varphi_1, \varphi_2, \dots, \varphi_n$, skriver vi $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \not\models \psi$.

Vi noterar att slutledningen $\varphi_1, \varphi_2, \dots, \varphi_n / \psi$ är satslogiskt giltig om och endast om $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$.

Exempel 18: Avgör om följande informella slutledning är satslogiskt giltig.

Endast om Person studerar flitigt, så blir han lärd.
Persons liv blir fattigare, om han inte blir lärd.
Persons liv blir inte fattigare.
____
Person studerar flitigt.

Lösning: Vi börjar med att formalisera slutledning för att erhålla en formell sådan. Observera att den första premissen innehåller ett nödvändigt villkor. Vi får då följande formalisering av slutledningen, där vi använder följande satsparametrar.

P: Person studerar flitigt,
L: Person blir lärd;
A: Persons liv blir fattigare.

$$\begin{array}{l} \neg P \to \neg L \\ \neg L \to A \\ \neg A \\ \hline P \end{array}$$

Antag nu för motsägelse att $\{\neg P \to \neg L, \neg L \to A, \neg A\} \not\models P$.

Då finns en värdering V sådan att

(1) $V(\neg P \to \neg L) = S$,
(2) $V(\neg L \to A) = S$,
(3) $V(\neg A) = S$ och
(4) $V(P) = F$.

Enligt (1) och (4), måste

(5) $V(\neg L) = S$ (varför?),

och enligt (2) och (3), måste

(6) $V(\neg L) = F$ (varför?)

Vi har då en motsägelse och därmed är det klart att

$\{\neg P \to \neg L, \neg L \to A, \neg A\} \models P$.

Med detta har vi att den informella slutledningen är satslogiskt giltig.

Teorem 1: Låt $\varphi_1, \varphi_2, \dots, \varphi_n, \psi$ och θ vara välbildade formler. Då gäller

a/ $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$ om och endast om $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \rightarrow \theta$.
b/ $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$ om och endast om $\models \varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \rightarrow \psi$.
c/ $\psi \models \theta$ om och endast om $\models \psi \rightarrow \theta$.

Bevis: a/ $\Rightarrow$/ Antag att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$ och antag för motsägelse att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \not\models \psi \rightarrow \theta$.

Då finns en värdering V så att $V(\varphi_1) = V(\varphi_2) = \dots = V(\varphi_n) = S$ och $V(\psi \rightarrow \theta) = F$, som ger att $V(\psi) = S$ och $V(\theta) = F$. Vi har alltså att $V(\varphi_1) = V(\varphi_2) = \dots = V(\varphi_n) = V(\psi) = S$ och $V(\theta) = F$, men detta motsäger förutsättningen $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$, som ju säger att θ är sann i alla värderingar i vilka $\varphi_1, \varphi_2, \dots, \varphi_n$ och ψ är sanna.

$\therefore$ $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \rightarrow \theta$.

$\Leftarrow$/ Antag att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \rightarrow \theta$ och antag för motsägelse att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \not\models \theta$.

Då finns en värdering V så att $V(\varphi_1) = V(\varphi_2) = \dots = V(\varphi_n) = V(\psi) = S$ och $V(\theta) = F$.
I denna värdering gäller då att $V(\varphi_1) = V(\varphi_2) = \dots = V(\varphi_n) = S$ och $V(\psi \rightarrow \theta) = F$, men detta motsäger förutsättningen.

$\therefore$ $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$.

Därmed är det klart att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$ om och endast om $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \rightarrow \theta$.

b och c är lämpliga övningar. Beviset för b är snarlikt det för a, och c är ett specialfall av a.

Definition 10: Låt φ och ψ vara två välbildade formler. Vi säger att φ och ψ är *satslogiskt ekvivalenta* om och endast om $V(\varphi) = V(\psi)$ för alla värderingar V, och vi skriver $\varphi \Leftrightarrow \psi$ eller $\varphi =\!\!|\!\!\models \psi$.

Observera att det följer direkt av satsen ovan att $\varphi \Leftrightarrow \psi$ om och endast om $\models \varphi \leftrightarrow \psi$. Vi ser här ytterligare ett exempel på ett problem som kan ”översättas till” problemet att avgöra huruvida en formel är en tautologi.

Exempel 19: Undersök om $\neg(\varphi \rightarrow \psi) \Leftrightarrow \varphi \wedge \neg\psi$.

Lösning: I kraft av observationen ovan inser vi att vi har att visa att $\models \neg(\varphi \rightarrow \psi) \leftrightarrow (\varphi \wedge \neg\psi)$.

Antag för motsägelse att $\not\models \neg(\varphi \rightarrow \psi) \leftrightarrow (\varphi \wedge \neg\psi)$.

Då finns värdering V så att $V \not\models \neg(\varphi \rightarrow \psi) \leftrightarrow (\varphi \wedge \neg\psi)$, som ger att
(1) $V(\neg(\varphi \rightarrow \psi)) = S$ och $V(\varphi \wedge \neg\psi) = F$, eller
(2) $V(\neg(\varphi \rightarrow \psi)) = F$ och $V(\varphi \wedge \neg\psi) = S$.

Det första påståendet i (1) ger att $V(\varphi \to \psi) = F$, som i sin tur ger att $V(\varphi) = S$ och $V(\psi) = F$ och vi har därmed att $V(\varphi \wedge \neg\psi) = S$, som motsäger det andra påståendet i (1).

Det andra påståendet i (2) ger att $V(\varphi) = S$ och $V(\neg\psi) = S$, som ger att $V(\varphi \to \psi) = F$, som i sin tur motsäger det första påståendet i (2).

Båda fallen leder alltså till motsägelse, och därmed är det klart att $\models \neg(\varphi \to \psi) \leftrightarrow (\varphi \wedge \neg\psi)$, och vi har därför $\neg(\varphi \to \psi) \Leftrightarrow \varphi \wedge \neg\psi$.

Ett vanligt problem i logik är att man har en formel som man av någon anledning vill skriva på ett annat sätt. Man kanske vill ha en kortare formel som är logiskt ekvivalent med den givna formeln. Betrakta till exempel följande svit av logiska ekvivalenser.

$$(A \vee B) \wedge (A \to B) \Leftrightarrow (A \vee B) \wedge (\neg A \vee B) \Leftrightarrow (A \wedge \neg A) \vee B \Leftrightarrow \bot \vee B \Leftrightarrow B.$$

I “förenklingen” ovan har i det första steget den logiska ekvivalensen $\varphi \to \psi \Leftrightarrow \neg\varphi \vee \psi$ utnyttjats tillsammans med *substitutionsprincipen*, som formuleras nedan. I det andra steget har en distributiv lag utnyttjats. I steg tre har den logiska ekvivalensen $\varphi \wedge \neg\varphi \Leftrightarrow \bot$ och substitutionsprincipen utnyttjats, och i det sista steget en logisk ekvivalens. Kalkyler sådana som den ovan utförs ofta och i dylika ”algebraiska” omskrivningar är substitutionsprincipen central. Det är också viktigt att man förfogar över en arsenal med logiska ekvivalenser för att smidigt kunna genomföra omskrivningarna. Vi formulerar först substitutionsprincipen och presenterar sedan en lista med några viktiga logiska ekvivalenser.

Teorem 2 *(Substitutionsprincipen)*: Låt θ vara en vbf, som innehåller en eller flera förekomster av delformeln φ, och antag att $\varphi \Leftrightarrow \psi$, där φ och ψ är två vbf. Antag dessutom att θ' erhålls från θ genom att ersätta en eller flera förekomster av φ med ψ. Då gäller det att $\theta \Leftrightarrow \theta'$.

Observera att det är substitutionsprincipen vi utnyttjat ovan, där vi i det första steget med beteckningarna i teorem 2 har

$\theta = (A \vee B) \wedge (A \to B)$,
$\varphi = (A \to B)$,
$\psi = \neg A \vee B$ och
$\theta' = (A \vee B) \wedge (\neg A \vee B)$.

Bevis: Låt V vara en godtycklig värdering. Vi har att visa att $V(\theta) = V(\theta')$.
Eftersom $V(\varphi) = V(\psi)$, så innebär varje ersättning av delformeln φ i θ med ψ att vi ersätter en delformel med en delformel med samma sanningsvärde. Men då påverkar vi inte sanningsvärdet hos θ, varför θ och θ' måste ha samma sanningsvärde. Därmed är det klart att $\theta \Leftrightarrow \theta'$.

Några viktiga logiska ekvivalenser

I formlerna nedan är φ, ψ och θ godtyckliga vbf.

(1)	$\neg\neg\varphi \Leftrightarrow \varphi$	
(2)	$\varphi \wedge \psi \Leftrightarrow \psi \wedge \varphi$	(Kommutativ lag)
(3)	$\varphi \vee \psi \Leftrightarrow \psi \vee \varphi$	(Kommutativ lag)
(4)	$(\varphi \wedge \psi) \wedge \theta \Leftrightarrow \psi \wedge (\varphi \wedge \theta)$	(Associativ lag)
(5)	$(\varphi \vee \psi) \vee \theta \Leftrightarrow \psi \vee (\varphi \vee \theta)$	(Associativ lag)
(6)	$\varphi \wedge (\psi \vee \theta) \Leftrightarrow (\varphi \wedge \psi) \vee (\varphi \wedge \theta)$	(Distributiv lag)
(7)	$\varphi \vee (\psi \wedge \theta) \Leftrightarrow (\varphi \vee \psi) \wedge (\varphi \vee \theta)$	(Distributiv lag)
(8)	$\neg(\varphi \wedge \psi) \Leftrightarrow \neg\varphi \vee \neg\psi$	(De Morgans lag)
(9)	$\neg(\varphi \vee \psi) \Leftrightarrow \neg\varphi \wedge \neg\psi$	(De Morgans lag)
(10)	$\varphi \rightarrow \psi \Leftrightarrow \neg\psi \rightarrow \neg\varphi$	(Kontraposition)
(11)	$\varphi \rightarrow \psi \Leftrightarrow \neg\varphi \vee \psi$	
(12)	$\neg(\varphi \rightarrow \psi) \Leftrightarrow \varphi \wedge \neg\psi$	
(13)	$\varphi \leftrightarrow \psi \Leftrightarrow (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$	
(14)	$\neg(\varphi \leftrightarrow \psi) \Leftrightarrow \neg\varphi \leftrightarrow \psi$	
(15)	$\neg(\varphi \leftrightarrow \psi) \Leftrightarrow \varphi \leftrightarrow \neg\psi$	
(16)	$\varphi \leftrightarrow \psi \Leftrightarrow \neg\varphi \leftrightarrow \neg\psi$	
(17)	$\varphi \wedge \psi \rightarrow \theta \Leftrightarrow \varphi \rightarrow (\psi \rightarrow \theta)$	
(18)	$\varphi \rightarrow \psi \wedge \theta \Leftrightarrow (\varphi \rightarrow \psi) \wedge (\varphi \rightarrow \theta)$	
(19)	$\varphi \vee \psi \rightarrow \theta \Leftrightarrow (\varphi \rightarrow \theta) \wedge (\psi \rightarrow \theta)$	
(20)	$\varphi \rightarrow \psi \vee \theta \Leftrightarrow (\varphi \rightarrow \psi) \vee (\varphi \rightarrow \theta)$	
(21)	$\varphi \wedge \psi \rightarrow \theta \Leftrightarrow (\varphi \rightarrow \theta) \vee (\psi \rightarrow \theta)$	

Vi avslutar så detta avsnitt med att i ett exempel visa på den teknik som normalt används när man vill bevisa något om strukturer hos formler. Man utnyttjar då induktion över formelns längd mätt som antal konnektiver i formeln.

Exempel 20: Visa att det finns minst en värdering V sådan att $V(\varphi) = S$, om formeln φ inte innehåller någon förekomst av symbolen ”$\neg$”. Visa också att det finns formler φ som inte innehåller ” $\neg$”, och som är sådana att $V(\varphi) = S$ för varje värdering V.

Lösning: Betrakta den värdering V i vilken $V(A_i) = S$ för alla $i \geq 1$.
Vi visar med induktion att $V(\varphi) = S$, om φ inte innehåller någon förekomst av symbolen ”$\neg$”.

Grundsteg: φ innehåller inte något konnektiv. Då är $\varphi = A_i$ för något i och därmed gäller enligt definition av V att $V(\varphi) = S$.

Induktionssteg: Antag för induktion att $V(\varphi) = S$ för alla formler φ som innehåller högst n konnektiver. Vi ska visa att $V(\varphi) = S$, om φ innehåller $n + 1$ konnektiver. Antag att φ innehåller $n + 1$ konnektiver. Det finns nu fyra möjligheter.

(1) φ är en konjunktion. Då gäller det att $\varphi = \psi \wedge \theta$ för några formler ψ och θ, där ψ och θ båda innehåller högst n konnektiver (varför?). Då gäller enligt induktionsantagandet att $V(\psi) = V(\theta) = S$, och därmed att $V(\varphi) = V(\psi \wedge \theta) = S$.

(2) φ är en disjunktion. På samma sätt som (1), men med $\vee$ i stället för $\wedge$.

(3) φ är en materiell implikation. På samma sätt som (1), men med $\rightarrow$ i stället för $\wedge$.
(4) φ är en materiell ekvivalens. På samma sätt som (1), men med $\leftrightarrow$ i stället för $\wedge$.

Induktionsaxiomet ger då för denna värdering V att $V(\varphi) = S$ för godtycklig formel φ, som inte innehåller någon förekomst av symbolen ”$\neg$”.

För att bevisa det andra påståendet är det tillräckligt att konstruera en tautologi, som inte innehåller symbolen ”$\neg$”. Betrakta till exempel formeln $(A \rightarrow B) \vee (B \rightarrow C)$, som man enkelt visar är en tautologi (gör det!).

Disjunktiv och konjunktiv normalform

I många sammanhang är det viktigt att ha tillgång till standardiserade sätt att skriva formler. I detta avsnitt skall vi visa på två sätt att skriva satslogiska formler på ett standardiserat sätt. Den disjunktiva normalformen, DNF, har viktiga tillämpningar vid till exempel konstruktion av logiska kretsar, medan den konjunktiva normalformen, KNF, har tillämpningar inom logikprogrammering. Huvudidén är att varje satslogisk formel skall kunna skrivas om så att den är logiskt ekvivalent med en formel som är uppbyggd enbart med hjälp av konnektiven $\wedge, \vee$ och $\neg$.

Vi inleder med några definitioner.

Definition 11: En *literal* är en satsparameter eller en negerad satsparameter.
En *itererad konjunktion* är en formel på formen $\varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n$ där φ_i inte innehåller någon konjunktion. Det är tillåtet att $i = 1$, varför en itererad konjunktion kan innehålla en konjunkt.
En *itererad disjunktion* är en formel på formen $\varphi_1 \vee \varphi_2 \vee \ldots \vee \varphi_n$ där φ_i inte innehåller någon disjunktion. Även här är det tillåtet att $i = 1$.

Exempel 21: A_7 och $\neg A_3$ är exempel på literaler.

$(A_1 \vee A_2) \wedge \neg A_1$ är en itererad konjunktion, där den första konjunkten är en itererad disjunktion.

$(A_1 \rightarrow A_2) \vee \neg A_7$ är en itererad disjunktion.

Observera att A_3 kan uppfattas både som en itererad disjunktion med en disjunkt och som en itererad konjunktion med en konjunkt.

Med detta i åtanke kan vi kanske inse att $A_3 \wedge \neg A_2 \wedge \neg A_5 \wedge A_2$ kan uppfattas som en itererad konjunktion med fyra konjunkter, eller som en itererad disjunktion med en disjunkt.

Definition 12: En formel φ är på *konjunktiv normalform (KNF)* om och endast om φ är en itererad konjunktion av itererade disjunktioner av literaler.
φ är på *disjunktiv normalform (DNF)* om och endast om φ är en itererad disjunktion av itererade konjunktioner av literaler.

Detta innebär att formeln φ är på KNF om φ kan skrivas på formen $\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n$, där varje φ_i i sin tur är på formen $\psi_{i1} \vee \psi_{i2} \vee \ldots \vee \psi_{im_i}$, där i sin tur varje ψ_{ij} är en literal. Notera att n respektive m_i kan vara 1. På motsvarande sätt har vi att en formel φ är på DNF om φ kan skrivas på formen $\varphi_1 \vee \varphi_2 \vee \dots \vee \varphi_n$, där varje φ_i i sin tur är på formen $\psi_{i1} \wedge \psi_{i2} \wedge \ldots \wedge \psi_{im_i}$, där som ovan ψ_{ij} är literaler.

Exempel 22: Formeln $(A_1 \vee A_2) \wedge \neg A_1$ är på KNF.

Formeln $A_3 \wedge \neg A_2 \wedge \neg A_5 \wedge A_2$ är på KNF, men den är också på DNF.

Även en formel som A_5 är på både KNF och DNF.

Observera också att kontradiktionen $A_1 \wedge \neg A_1$ och tautologin $A_1 \vee \neg A_1$ båda är på både KNF och DNF.

Vi skall nu bevisa att varje formel kan skrivas på KNF.

Teorem 3: För varje satslogisk formel φ, finns en satslogisk formel ψ så att
(i) $\varphi \Leftrightarrow \psi$, och
(ii) ψ är på KNF

Innan vi genomför själva beviset skall vi illustrera en teknik att skriva formler på KNF. Det är denna teknik vi sedan formulerar allmänt i själva beviset.

Exempel 23: Ange en formel på KNF som är logiskt ekvivalent med $\neg A \vee B \to C$.

Lösning: Den givna formeln har nedanstående sanningsvärdestabell. Kontrollera själv att tabellen är korrekt konstruerad.

A B C	$\neg A \vee B \to C$
S S S	S
S S F	F
S F S	S
S F F	S
F S S	S
F S F	F
F F S	S
F F F	F

Vi ser att formeln är falsk I tilldelningarna på rad 2, 6 och 8. Bildar vi formeln $\neg A \vee \neg B \vee C$, så har vi bildat en formel som är falsk i precis den sanningsvärdestilldelningen vi har på rad 2 och sann i de andra sju. På motsvarande sätt är formlerna $A \vee \neg B \vee C$ och $A \vee B \vee C$ falska enbart i de tilldelningar vi har på raderna 6 respektive 8. Bildar vi sedan en itererad konjunktion av dessa itererade disjunktioner

$$(\neg A \vee \neg B \vee C) \wedge (A \vee \neg B \vee C) \wedge (A \vee B \vee C),$$

har vi konstruerat en formel som måste ha värdet F exakt i sanningsvärdestilldelningarna på raderna 2, 6 och 8. Varför?

Vi har alltså att $\neg A \vee B \rightarrow C \;\Leftrightarrow\; (\neg A \vee \neg B \vee C) \wedge (A \vee \neg B \vee C) \wedge (A \vee B \vee C)$ där den andra formeln är på KNF.

Själva beviset för teorem 3 går nu ut på att skriva ner denna procedur i generella termer.

Bevis av teorem 3: Vi delar upp beviset i två fall, för observera att tekniken i exempel 3 ovan förutsätter att sanningstabellen innehåller minst ett F, och en formel ju faktiskt kan vara en tautologi.

Fall 1: φ är en tautologi.

Välj en satsparameter, till exempel A_1, och bilda $A_1 \vee \neg A_1$. För denna formel gäller då att

(i) $\varphi \Leftrightarrow A_1 \vee \neg A_1$, och
(ii) $A_1 \vee \neg A_1$ är på KNF.

Fall 2: φ är inte en tautologi.

Vi förutsätter att φ innehåller satsparametrarna $A_1, A_2, \ldots, A_m$.

Bilda för varje värdering V_i sådan att $V_i(\varphi) = F$ följande itererade disjunktion

$\psi_{i1} \vee \psi_{i2} \vee \ldots \vee \psi_{im}$, där

$$\psi_{ij} = \begin{cases} A_j, & \text{om } V(A_j) = F \\ \neg A_j, & \text{om } V(A_j) = S \end{cases} \qquad \text{för } j = 1, 2, \ldots, m$$

Bilda sedan en itererad konjunktion av dessa itererade disjunktioner. Denna itererade konjunktion kommer då att innehålla lika många konjunkter som det finns F i sanningstabellen för formeln φ. Det gäller nu att formeln konstruerad enligt ovan är på KNF och logiskt ekvivalent med φ.

Teorem 4: För varje satslogisk formel φ, finns en satslogisk formel ψ sådan att
(i) $\varphi \Leftrightarrow \psi$, och
(ii) ψ är på DNF.

Bevisskiss: Detta bevis är analogt med föregående och följer idén i nedanstående exempel. Vi urskiljer två fall.

Fall 1: φ är en kontradiktion.

Fall 2: φ är inte en kontradiktion.

Respektive fall hanteras sedan analogt med ovanstående bevis, och i överensstämmelse med nedanstående exempel.

Exempel 24: Skriv formeln $A \leftrightarrow (B \wedge \neg C)$ på DNF.

Lösning: Formeln har följande sanningsvärdestabell.

A B C	$A \leftrightarrow (B \wedge \neg C)$
S S S	F
S S F	S
S F S	F
S F F	F
F S S	S
F S F	F
F F S	S
F F F	S

Betrakta de värderingar I vilka formeln är sann.

Rad 2: Formeln $A \wedge B \wedge \neg C$ är sann i värderingen på rad 2, och falsk för övrigt.
Rad 5: Formeln $\neg A \wedge B \wedge C$ är sann i värderingen på rad 5, och falsk för övrigt.
Rad 7: Formeln $\neg A \wedge \neg B \wedge C$ är sann i värderingen på rad 7, och falsk för övrigt.
Rad 8: Formeln $\neg A \wedge \neg B \wedge \neg C$ är sann i värderingen på rad 8, och falsk för övrigt.

Bilda sedan en itererad disjunktion av ovanstående itererade konjunktioner. Vi får då formeln

$(A \wedge B \wedge \neg C) \vee (\neg A \wedge B \wedge C) \vee (\neg A \wedge \neg B \wedge C) \vee (\neg A \wedge \neg B \wedge \neg C)$.

Denna formel är på DNF och logiskt ekvivalent med $A \leftrightarrow (B \wedge \neg C)$. Varför?

Ovanstående metod att skriva formler på DNF respektive KNF är mekanisk men i många fall onödigt omständlig. Ofta kan man genom att utnyttja logiska ekvivalenser nå målet betydligt snabbare.

Exempel 25: Skriv $\neg A \vee B \rightarrow C$ på DNF.

Lösning: Vi har följande svit av logiska ekvivalenser.

$\neg A \vee B \rightarrow C \;\Leftrightarrow\; \neg(\neg A \vee B) \vee C \;\Leftrightarrow\; (A \wedge \neg B) \vee C$, där den sista formeln är på DNF.

Exempel 26: Skriv $A \leftrightarrow B$ och $\neg A \rightarrow (B \leftrightarrow \neg C)$ på KNF.

Lösning: $A \leftrightarrow B \;\Leftrightarrow\; (A \rightarrow B) \wedge (B \rightarrow A) \;\Leftrightarrow\; (\neg A \vee B) \wedge (\neg B \vee A)$, som är på KNF.

$\neg A \rightarrow (B \leftrightarrow \neg C) \;\Leftrightarrow\; \neg\neg A \vee (B \leftrightarrow \neg C) \;\Leftrightarrow\; A \vee ((B \rightarrow \neg C) \wedge (\neg C \rightarrow B)) \;\Leftrightarrow$
$\Leftrightarrow\; A \vee ((\neg B \vee \neg C) \wedge (\neg\neg C \vee B)) \;\Leftrightarrow\; (A \vee \neg B \vee \neg C) \wedge (A \vee B \vee C)$, där den sista formeln är på KNF.

Fullständiga konnektivmängder

Vi såg ovan hur varje satslogisk formel φ kan skrivas på en form innehållande enbart konnektiven $\wedge$, $\vee$ och $\neg$. Man kan då fråga sig om det går att reducera antalet använda konnektiv ytterligare.

Definition 13: En mängd M med konnektiv är *fullständig* om och endast om det för varje formel φ finns en formel ψ så att
(i) $\varphi \Leftrightarrow \psi$, och
(ii) samtliga konnektiv i ψ tillhör M.

Teorem 5: $\{\wedge, \vee, \neg\}$ är fullständig.

Bevis: Följer omedelbart av teorem 3 ovan.

Teorem 6: a/ $\{\wedge, \neg\}$ är fullständig.
b/ $\{\vee, \neg\}$ är fullständig.
c/ $\{\rightarrow, \neg\}$ är fullständig.

Bevis: a/ Låt φ vara en godtycklig formel.. Då finns enligt teorem 1 en formel ψ sådan att ψ är på KNF och logiskt ekvivalent med φ. Ersätt sedan varje förekomst av en delformel på formen $\psi_1 \vee \psi_2$ i ψ med $\neg(\neg\psi_1 \wedge \neg\psi_2)$. Observera att $\psi_1 \vee \psi_2 \Leftrightarrow \neg(\neg\psi_1 \wedge \neg\psi_2)$.
Vi får då en formel θ som är sådan att de enda konnektiven i θ är $\wedge$ och $\neg$, och dessutom gäller enligt substitutionsprincipen att $\varphi \Leftrightarrow \theta$.

b och c bevisas på motsvarande sätt och lämnas som övning.

Exempel 27: Konstruera en formel som enbart innehåller konnektiven $\wedge$ och $\neg$, och som är logiskt ekvivalent med $\neg(A \vee B) \rightarrow \neg C$.

Lösning: Vi skulle kunna åstadkomma detta genom att först, med hjälp av sanningstabeller, konstruera en formel på KNF ekvivalent med den givna formeln, och därefter eliminera disjunktioner med hjälp av DeMorgans lag. Vi väljer i stället att skriva om formeln via en svit av logiska ekvivalenser.

$\neg(A \vee B) \rightarrow \neg C \Leftrightarrow (A \vee B) \vee \neg C \Leftrightarrow \neg\neg A \vee \neg\neg B \vee \neg C) \Leftrightarrow \neg(\neg A \wedge \neg B \wedge C)$.

Definition 14: Definiera det tvåställiga konnektivet | (NotAND, NAND) enligt följande

A B	A|B
S S	F
S F	S
F S	S
F F	S

Eller formellt $V(\varphi|\psi) = F$ om och endast om $V(\varphi) = V(\psi) = S$.

Vi ser att $\varphi|\psi \Leftrightarrow \neg(\varphi \wedge \psi)$. Det är detta som motiverar namnet på konnektivet.

Exempel 28: Skriv $\neg A \vee B$ enbart med hjälp av konnektivet $|$.

Lösning: Vi visar först att $\neg\varphi \Leftrightarrow \varphi|\varphi$.
Låt V vara en godtycklig värdering och antag först att $V(\neg\varphi) = S$.
Då måste $V(\varphi) = F$, och därmed måste $V(\varphi|\varphi) = S$ enligt definitionen ovan.
Antag sedan att $V(\neg\varphi) = F$, dvs $V(\varphi) = S$. Då måste $V(\varphi|\varphi) = F$. I båda fallen gäller alltså att $V(\neg\varphi) = V(\varphi|\varphi)$, och några andra fall finns inte.

Därmed är det klart att $\neg\varphi \Leftrightarrow \varphi|\varphi$.

Betrakta sedan följande svit av logiska ekvivalenser.

$\neg A \vee B \Leftrightarrow \neg(A \wedge \neg B) \Leftrightarrow A|\neg B \Leftrightarrow A|(B|B)$.

Teorem 7: $\{\,|\,\}$ är en fullständig konnektivmängd.

Bevis: Vi vet enligt teorem 6 att $\{\neg, \wedge\}$ är fullständig.
Enligt ovanstående exempel gäller det att $\neg\varphi \Leftrightarrow \varphi|\varphi$.
Detta resultat tillsammans med $\varphi|\psi \Leftrightarrow \neg(\varphi \wedge \psi)$ ger då att

$$\varphi \wedge \psi \Leftrightarrow (\varphi|\psi)|(\varphi|\psi).$$

Substitutionsprincipen ger då att varje förekomst av konnektiven $\neg$ och $\wedge$ kan elimineras till förmån för $|$.

Definition 16: Definiera det tvåställiga konnektivet $\downarrow$ (NotOR, NOR) enligt följande

A B	A↓B
S S	F
S F	F
F S	F
F F	S

Formellt $V(\varphi\downarrow\psi) = S$ om och endast om $V(\varphi) = V(\psi) = F$.

Vi ser att $\varphi\downarrow\psi \Leftrightarrow \neg(\varphi \vee \psi)$. Återigen har vi här en motivering av namnet på konnektivet.

Vi avslutar med att formulera ytterligare två teorem. Beviset av det första är analogt med beviset av teorem 7.

Teorem 8: $\{\downarrow\}$ är en fullständig konnektivmängd.

Bevis: Övning.

Teorem 9: Det finns inga andra fullständiga konnektivmängder med ett element än $\{\,|\,\}$ och $\{\downarrow\}$.

Kunniga i elektronik vet att kretsar kan konstrueras enbart med hjälp av till exempel grinden NAND. Detta är en konsekvens av teorem 7.

Naturlig deduktion i satslogik

Ofta är det smidigare att använda syntaktiska tekniker (härledningar) för att avgöra huruvida slutledningar är giltiga eller ej. Detta förutsätter förstås att vi har ett ”bra” syntaktiskt system, dvs ett system som är sådant att φ kan härledas från premissmängden Γ om och endast om $\Gamma \models \varphi$. Med tanke på automatisering av bevis, så är det förstås också intressant att studera syntaktiska system, system som kan manipuleras av en dator. Det system för deduktioner vi presenterar i det följande är en variant av ett system som har skapats av den amerikanske logikern Benson Mates.

Ett *formellt system* består av en *kalkyl* och en *tolkning*. Kalkylen i sin tur består av

(i) Vokabulär (lexikon),
(ii) Formationsregler,
(iii) Axiom och
(iv) Deduktionsregler (härledningsregler).

Den kalkyl vi skall skapa har samma vokabulär och samma formationsregler som vi presenterat för satslogik ovan. Kalkylen saknar axiom, men har en relativt stor uppsättning deduktionsregler, som är uppdelade i introduktionsregler (som anger hur en formel med ett givet konnektiv får införas) och eliminationsregler (som anger hur en formel med ett givet konnektiv kan uppdelas i delformler). Vi benämner vårt formella system för satslogik med S.

Grundläggande deduktionsregler

Introduktionsregler		Eliminationsregler	
$\dfrac{\varphi \quad \neg\varphi}{\bot}$	$\bot$I		
$\dfrac{\begin{array}{c}\varphi\\ \bot\end{array}}{\neg\varphi}$	$\neg$I	$\dfrac{\begin{array}{c}\neg\varphi\\ \bot\end{array}}{\varphi}$	$\neg$E
$\dfrac{\varphi \quad \psi}{\varphi \wedge \psi}$	$\wedge$I	$\dfrac{\varphi \wedge \psi}{\varphi} \quad \dfrac{\varphi \wedge \psi}{\psi}$	$\wedge$E
$\dfrac{\varphi}{\varphi \vee \psi} \quad \dfrac{\psi}{\varphi \vee \psi}$	$\vee$I	$\dfrac{\varphi \vee \psi \quad \varphi \to \theta \quad \psi \to \theta}{\theta}$	$\vee$E
$\dfrac{\begin{array}{c}\varphi\\ \psi\end{array}}{\varphi \to \psi}$	$\to$I	$\dfrac{\varphi \to \psi \quad \varphi}{\psi}$	$\to$E
$\dfrac{\varphi \to \psi \quad \psi \to \varphi}{\varphi \leftrightarrow \psi}$	$\leftrightarrow$I	$\dfrac{\varphi \leftrightarrow \psi}{\varphi \to \psi} \quad \dfrac{\varphi \leftrightarrow \psi}{\psi \to \varphi}$	$\leftrightarrow$E

I reglerna $\neg$E, $\neg$I, $\to$I är $\neg\varphi$, φ respektive φ provisoriska premisser. Mer om detta nedan. Reglerna benämns negationsintroduktion, negationselimination etc. Observera att samtliga härledningsregler opererar på respektive sats huvudoperator. Man kan alltså inte manipulera delformler. System som ovanstående brukar benämnas system för *naturlig deduktion*. Deduktionsreglerna kodifierar hur argumentation går till i mer formaliserade sammanhang som matematik, datavetenskap osv. De kodifierar också hur man formellt skall argumentera även i icke-formella sammanhang, givet att man vill argumentera logiskt bindande.

Den, logiskt sett, mest omdiskuterade regeln är $\neg$E, reductio ad absurdum. I så kallad intuitionistisk logik accepteras den inte. Man får då ett svagare system. Dylika studeras flitigt bland annat inom datalogi.

Reglerna är av två olika typer.

(i) $\neg$E, $\neg$I, $\rightarrow$I

Dessa regler involverar ett antagande, en provisorisk premiss som man använder sig av i själva härledningen, men som man gör sig av med vid själva användningen av respektive regel.

(ii) Övriga regler.

Definition 17: En *deduktion (härledning)* är en ändlig följd av konsekutivt numrerade rader, som var och en innehåller en premissnummermängd X_i och en välbildade formel φ_i sådan att

1. φ_i är en premiss. Som X_i tags radnumret.
2. φ_i följer från tidigare satser med hjälp av någon av reglerna i grupp (ii). Som premissnummermängd väljs unionen av de åberopade satsernas premissnummermängder.
3. φ_i följer från tidigare satser med hjälp av någon av reglerna i grupp (i). Som premissnummermängd väljs mängddifferensen av den åberopade satsens premissnummermängd och den provisoriska premissens.

Slutsatsen i en härledning är den sista satsen i härledningen. Vi skriver $\{\varphi_1, \varphi_2, \ldots, \varphi_n\} \vdash \psi$ om ψ är slutsats i en härledning där $\varphi_1, \varphi_2, \ldots, \varphi_n$ är premisser. Egentligen borde vi här skriva $S + \{\varphi_1, \varphi_2, \ldots, \varphi_n\} \vdash \psi$, men eftersom S förekommer i bakgrunden i varje satslogisk deduktion kan vi utelämna S. Vi skriver också $\Gamma \vdash \psi$, om Γ är premissmängd i härledningen av ψ.

I detta finns minst tre svårigheter. Den första är att använda själva deduktionsreglerna korrekt. Den andra är att konstruera premissnummermängderna, som är tänkta att hålla reda på vilka premisser ett härlett resultat beror på. Den tredje svårigheten är att utföra en hel härledning för att bevisa det man vill bevisa. Det är viktigt att läsa många exempel, och att själv lösa många uppgifter själv för att skaffa sig rutin på detta.

Vi illustrerar först en härledning, och beskriver därefter något mer ingående hur premissnummermängdermängder konstrueras.

Exempel 29: Visa att $\{A_1, \neg(A_1 \wedge \neg A_2)\} \vdash A_2$.

Lösning:

$\{1\}$	1	A_1	Premiss
$\{2\}$	2	$\neg(A_1 \wedge \neg A_2)$	Premiss
$\{3\}$	3	$\neg A_2$	Provisorisk premiss (för $\neg$E)
$\{1, 3\}$	4	$A_1 \wedge \neg A_2$	1, 3, $\wedge$I
$\{1, 2, 3\}$	5	$\perp$	2, 4, $\perp$I
$\{1, 2\}$	6	A_2	3, 5, $\neg$E

Slutsatsen A_2 beror endast på premisserna på rad 1 och 2. Därför skriver vi upp dessa premisser i premissmängden.

Vi har då att $\{A_1, \neg(A_1 \wedge \neg A_2)\} \vdash A_2$.

Notera att vi skulle kunna genomföra precis samma resonemang med godtyckliga, satslogiska, välbildade formler i stället för A_1 och A_2, dvs det gäller att $\{\varphi, \neg(\varphi \wedge \neg\psi)\} \vdash \psi$ för godtyckliga vbf φ och ψ. Skriv själv ner härledningen i detta fall.

Vi formulerar och exemplifierar nu hur premisnummermängder konstrueras.

1. En *premiss* får införas på valfri rad. Det enda elementet i premissnummermängden är radnumret (se exempel 29, rad 1).

2. En sats får införas på ny rad, om den följer av satser som förekommer på rader med lägre radnummer i sekvensen med någon av reglerna $\perp$I, $\wedge$I, $\wedge$E, $\vee$I, $\vee$E, $\rightarrow$E, $\leftrightarrow$I eller $\leftrightarrow$E. Som premissnummermängd tas unionen av de åberopade radernas premissnummermängder (se exempel 29, rad 4, 5).

   Vi illustrerar med $\vee$E. Antag att vi har en härledning med följande struktur:

X	i	$\varphi \vee \psi$	
Y	j	$\varphi \rightarrow \theta$	
Z	k	$\psi \rightarrow \theta$	
$X \cup Y \cup Z$	l	θ	i, j, k, $\vee$E

   Här måste l > i, j, k.

3. En sats får införas på ny rad, om den följer av satser som förekommer på rader med lägre radnummer i sekvensen med någon av reglerna $\neg$I, $\neg$E eller $\rightarrow$I.

   Vi illustrerar med $\neg$E (se exempel 29). Antag att vi har en härledning med följande struktur.

{i}	i	$\neg\varphi$	Premiss
X	j	$\perp$	
X-{i}	k	φ	i, j, $\neg$E

   Här måste i < j < k.

   Premissnummermängden för rad k är premissnummermängden för rad j, den rad på vilken $\perp$ förekommer, differens premissnummermängden för rad i, dvs för den provisoriska premissen i tillämpningen av $\neg$E.

   På motsvarande sätt används $\neg$I.

   För regeln $\rightarrow$I gäller följande. För att bilda premissnummermängden för den rad på vilken $\varphi \rightarrow \psi$ förekommer, gör vi på följande sätt. Tag premissnummermängden för raden på vilken ψ förekommer differens premissnummermängden för raden på vilken φ förekommer (se nedan exempel 30, 38).

Definition 18: En härledning från en tom premissmängd kallas ett *bevis*. Vi skriver då $\vdash \varphi$ i stället för $\varnothing \vdash \varphi$. φ kallas ett *teorem*. Även här borde vi egentligen skriva $S + \varnothing \vdash \varphi$, men låter som tidigare S vara underförstådd.

Exempel 30: Visa att $\varphi \to (\psi \to \varphi)$ är ett teorem.

Lösning:

{1}	1	φ	Prov Prem (för $\to$I)
{2}	2	ψ	Prov prem (för $\to$I)
{3}	3	$\neg\varphi$	Prov prem (för $\neg$E)
{1, 3}	4	$\bot$	1, 3, $\bot$I
{1}	5	φ	3, 4, $\neg$E
{1}	6	$\psi \to \varphi$	2, 5, $\to$I
$\varnothing$	7	$\varphi \to (\psi \to \varphi)$	1, 6, $\to$I

Därmed är det klart att $\vdash \varphi \to (\psi \to \varphi)$.

Exempel 31: Visa att $\vdash \varphi \to \varphi$.

Lösning:

{1}	1	φ	Prov prem (för $\to$I)
{2}	2	$\neg\varphi$	Prov prem (för $\neg$E)
{1, 2}	3	$\bot$	1, 2, $\bot$I
{1}	4	φ	2, 3, $\neg$E
$\varnothing$	5	$\varphi \to \varphi$	1, 4, $\to$I

Därmed är det klart att $\vdash \varphi \to \varphi$.

En annan härledning av ovanstående teorem är följande.

{1}	1	φ	Prov prem (för $\to$I)
$\varnothing$	2	$\varphi \to \varphi$	1, $\to$I

Här används rad 1 två gånger, både som premiss och slutsats.

Exempel 32: Visa att $\vdash \neg\neg\varphi \to \varphi$ och $\vdash \varphi \to \neg\neg\varphi$.

Lösning: Vi visar först att $\vdash \neg\neg\varphi \to \varphi$.

{1}	1	$\neg\neg\varphi$	Prov prem (för $\to$I)
{2}	2	$\neg\varphi$	Prov prem (för $\neg$E)
{1, 2}	3	$\bot$	1, 2, $\bot$I
{1}	4	φ	2, 3, $\neg$E
$\varnothing$	5	$\neg\neg\varphi \to \varphi$	1, 4, $\to$I

$\therefore\ \vdash \neg\neg\varphi \to \varphi$.

Vi visar sedan $\vdash \varphi \to \neg\neg\varphi$.

{1}	1	φ	Prov prem (för $\to$I)
{2}	2	$\neg\varphi$	Prov prem (för $\neg$I)
{1, 2}	3	$\bot$	1, 2, $\bot$I
{1}	4	$\neg\neg\varphi$	2, 3, $\neg$I
$\varnothing$	5	$\varphi \to \neg\neg\varphi$	1, 4, $\to$I

Därmed är det också klart att $\vdash \varphi \to \neg\neg\varphi$.

Härledda deduktionsregler

Vissa mönster återkommer ofta i härledningar. Till exempel är det vanligt att man har en formel som innehåller två på varandra följande negationssymboler, som man skulle vilja eliminera. Det är då praktiskt att ha en uppsättning deduktionsregler som tar hand om dessa vanliga mönster. Vi kan jämföra detta med situationen i vanlig algebra. Där är det till exempel vanligt att behöva skriva en kvadrat $(a + b)^2$ som en summa, och då är det praktiskt att ha en räkneregel för detta. Med en enkel modifikation av bevisen i exempel 32 ser vi att $\neg\neg\varphi \vdash \varphi$ och $\varphi \vdash \neg\neg\varphi$, och därmed kan vi införa följande härledda deduktionsregler.

$$\frac{\neg\neg\varphi}{\varphi} \qquad\qquad \frac{\varphi}{\neg\neg\varphi}$$

Båda dessa regler benämner vi *dubbel negation*, som vi förkortar DN.

Exempel 33: Visa att $\{\varphi \to \psi, \neg\psi\} \vdash \neg\varphi$.

Lösning:				
	{1}	1	$\varphi \to \psi$	Premiss
	{2}	2	$\neg\psi$	Premiss
	{3}	3	φ	Prov prem (för $\neg$I)
	{1, 3}	4	ψ	1, 3, $\to$E
	{1, 2, 3}	5	$\perp$	2, 4, $\perp$I
	{1, 2}	6	$\neg\varphi$	3, 5, $\neg$I

Därmed är det klart att $\{\varphi \to \psi, \neg\psi\} \vdash \neg\varphi$.

Även ovanstående samband inför vi som en härledd deduktionsregel, dvs

$$\frac{\begin{array}{c}\varphi \to \psi \\ \neg\psi\end{array}}{\neg\varphi}$$

Vi benämner denna regel med dess klassiska namn *modus tollens*, som vi förkortar MT.

Nedanstående härledningsregel är tacksam, eftersom härledningen av regeln är en aning omständlig, och man ofta har behov av denna regel.

Exempel 34: Visa att $\{\varphi \vee \psi, \neg\varphi\} \vdash \psi$.

Lösning: I denna härledning skall vi använda oss av regeln $\vee$E. Denna är den kanske besvärligast av satslogisken deduktionsregler. Regeln förutsätter att vi har en disjunktion. Ofta har man att själv skapa de två implikationer som behövs. Det är inte alltid helt klart vad som är lämpligt val av eftersats i de två implikationerna. Ofta kan man försöka härleda det man vill ha fram direkt, i vårt fall ψ. En annan strategi är att ur de båda disjunkterna försöka härleda $\perp$, om man nu försöker skapa en härledning med hjälp av $\neg$E eller $\neg$I.

{1}	1	$\varphi \vee \psi$	Premiss
{2}	2	$\neg\varphi$	Premiss
{3}	3	φ	Prov prem (för $\rightarrow$I)
{4}	4	$\neg\psi$	Prov prem (för $\neg$E)
{2, 3}	5	$\perp$	2, 3, $\perp$I
{2, 3}	6	ψ	4, 5, $\neg$E
{2}	7	$\varphi \rightarrow \psi$	3, 6, $\rightarrow$I
{8}	8	ψ	Prov prem (för $\rightarrow$I)
$\varnothing$	9	$\psi \rightarrow \psi$	8, $\rightarrow$I
{1, 2}	10	ψ	1, 7, 9, $\vee$E

Då är det klart att $\{\varphi \vee \psi, \neg\varphi\} \vdash \psi$.

Vi lämnar några kommentarer till härledningen ovan. För att härleda $\varphi \rightarrow \psi$, antar vi φ som provisorisk premiss på rad 3. Vi noterar att formlerna på rad 2 och 3 motsäger varandra. Detta innebär att vi kan härleda vad som helst. Vi vill komma fram till ψ, och en teknik för att göra detta är att antaga $\neg\psi$, som vi gör på rad 4. Vi har en motsägelse, och inför $\perp$ på rad 5 för att därefter använda $\neg$E. Konstruktionen på rad 8 har presenterats i exempel 31. Notera också att användning av $\vee$E kräver att vi refererar till tre rader i härledningen.

Vi formulerar resultatet i exemplet ovan som en härledningsregel. Denna förekommer i två varianter, där beviset av den andra är fullständigt analogt med beviset i exempel 34.

$$\frac{\varphi \vee \psi \qquad \neg\varphi}{\psi} \qquad\qquad \frac{\varphi \vee \psi \qquad \neg\psi}{\varphi}$$

Båda dessa regler benämns *disjunktion syllogism* och förkortas DS.

Härledningsstrategier

För att kunna genomföra en härledning på ett lyckosamt sätt, måste man ha en strategi. Betrakta exempel 33. I detta exempel vill vi deducera $\neg\varphi$ från premisserna $\varphi \rightarrow \psi$ och $\neg\psi$. Vi vill alltså härleda en negerad formel $\neg\varphi$. Huvudstrategin för detta är att antaga φ, för att sedan härleda en motsägelse (falsum). Motsägelsen härleds ur premisserna tillsammans med extraantagandet. Därefter använder vi negationsintroduktion. Undersöker vi exempel 34 så vill vi där härleda ψ från premisserna $\varphi \vee \psi$ och $\neg\varphi$. Minst två strategier är här möjliga. Den ena är att genomföra ett *motsägelsebevis*. Då antar vi $\neg\psi$ och försöker härleda $\perp$. Eftersom $\varphi \vee \psi$ är den ena premissen bör vi rimligen använda oss av $\vee$E. Vi har då att skapa två implikationer, $\varphi \rightarrow \perp$ och $\psi \rightarrow \perp$. Lyckas vi med detta ger $\vee$E just $\perp$. För att härleda en implikation antar vi försatsen som provisorisk premiss och försöker härleda eftersatsen, och sedan använder vi $\rightarrow$I. Den andra strategin är att genomföra ett *direkt bevis* av formeln ψ. Även då skall vi utnyttja $\vee$E. Nu skapar vi i stället de två implikationerna $\varphi \rightarrow \psi$ och $\psi \rightarrow \psi$. Det är denna strategi som använts och beskrivits i exempel 34.

Vi kan urskilja följande, ofta användbara, huvudstrategier. Observera att denna förteckning över strategier inte på något sätt uttömmer möjligheterna.

1. Härledning av en negerad formel $\neg\varphi$.
   Antag φ som provisorisk premiss, härled $\perp$, och använd $\neg$I.

2. Härledning av $\varphi \wedge \psi$.
   Härled φ respektive ψ var för sig, och använd $\wedge$I.
3. Härledning av $\varphi \vee \psi$.
   Enklast brukar här vara att göra ett motsägelsebevis, dvs antag $\neg(\varphi \vee \psi)$ som provisorisk premiss. Använd sedan De Morgans lag (se nedan exempel 35) för att få $\neg\varphi \wedge \neg\psi$. Denna formel används sedan för att generera $\perp$, varefter man avslutar med $\neg$E.
4. Härledning av $\varphi \rightarrow \psi$.
   Antag φ som provisorisk premiss, härled ψ, och använd $\rightarrow$I.
5. Härledning av $\varphi \leftrightarrow \psi$.
   Härled $\varphi \rightarrow \psi$ och $\psi \rightarrow \varphi$ och använd $\leftrightarrow$I.

Observera alltså att ovanstående strategier inte på något sätt täcker upp alla möjligheter. Ofta måste man förstås använda delstrategier. Skall vi till exempel härleda en ekvivalens, så använder vi idén i punkt 5, men då skall vi härleda två implikationer, och för detta använder vi idén i punkt 4 och så vidare.

Exempel 35: Visa att

a/	$\neg(\varphi \vee \psi) \vdash \neg\varphi \wedge \neg\psi$,
b/	$\neg\varphi \wedge \neg\psi \vdash \neg(\varphi \vee \psi)$,
c/	$\neg(\varphi \wedge \psi) \vdash \neg\varphi \vee \neg\psi$,
d/	$\neg\varphi \vee \neg\psi \vdash \neg(\varphi \wedge \psi)$

Lösning: a/ Den strategi vi använder nedan är att härleda den önskade konjunktionen enligt strategin i punkt 2. Varje konjunkt som skall härledas är en negerad formel, varför strategin i punkt 1 används för dessa delhärledningar (subrutiner).

{1}	1	$\neg(\varphi \vee \psi)$	Premiss
{2}	2	φ	Prov premiss (för $\neg$I)
{2}	3	$\varphi \vee \psi$	2, $\vee$I
{1, 2}	4	$\perp$	1, 3, $\perp$I
{1}	5	$\neg\varphi$	2, 4, $\neg$I
{6}	6	ψ	Prov premiss (för $\neg$I)
{6}	7	$\varphi \vee \psi$	6, $\vee$I
{1, 6}	8	$\perp$	1, 7, $\perp$I
{1}	9	$\neg\psi$	6, 8, $\neg$I
{1}	10	$\neg\varphi \wedge \neg\psi$	5, 9, $\wedge$I

$\therefore\ \neg(\varphi \vee \psi) \vdash \neg\varphi \wedge \neg\psi$.

b/ Övning. Som ett tips kan du anta $\varphi \vee \psi$ för att med hjälp av DS härleda $\perp$.

c/

{1}	1	$\neg(\varphi \wedge \psi)$	Premiss
{2}	2	$\neg(\neg\varphi \vee \neg\psi)$	Prov premiss (för $\neg$E)
{2}	3	$\neg\neg\varphi \wedge \neg\neg\psi$	2, ex 35a
{2}	4	$\neg\neg\varphi$	3, $\wedge$E
{2}	5	$\neg\neg\psi$	3, $\wedge$E
{2}	6	φ	4, DN
{2}	7	ψ	5, DN
{2}	8	$\varphi \wedge \psi$	6, 7, $\wedge$I
{1, 2}	9	$\perp$	1, 8, $\perp$I

{1}	10	$\neg\varphi \vee \neg\psi$	2, 9, $\neg$E

Tänk genom vilken strategi som använts ovan.

d/ Övning. Som ett tips kan du anta $\varphi \wedge \psi$ för att härleda $\perp$. Även i denna härledning kan du använda DS.

Vi har då bevisat fyra härledningsregler, och vi benämner alla De Morgans lagar (DM). Vi har alltså följande fyra härledda deduktionsregler.

$$\frac{\neg(\varphi \vee \psi)}{\neg\varphi \wedge \neg\psi} \qquad \frac{\neg\varphi \wedge \neg\psi}{\neg(\varphi \vee \psi)} \qquad \frac{\neg(\varphi \wedge \psi)}{\neg\varphi \vee \neg\psi} \qquad \frac{\neg\varphi \vee \neg\psi}{\neg(\varphi \wedge \psi)}$$

Teorem 10: (Deduktionsteoremet) Låt Γ vara en mängd av vbf, och låt ψ och θ vara två vbf. Då gäller $\Gamma \cup \{\psi\} \vdash \theta \;\Leftrightarrow\; \Gamma \vdash \psi \rightarrow \theta$.

Bevis: $\Rightarrow$/ Antag att $\Gamma \cup \{\psi\} \vdash \theta$.

Eftersom varje härledning är ändlig (varför är det så?), används högst ändligt många premisser i $\Gamma \cup \{\psi\}$ i härledningen av θ. Antag att dessa är $\varphi_1, \varphi_2, \dots, \varphi_n$ och ψ. Härledningen i antagandet har då följande form

{1}	1	φ_1	Premiss
{2}	2	φ_2	Premiss
:	:	:	:
{n}	n	φ_n	Premiss
{n+1}	n+1	ψ	Premiss
:	:	:	:
{1, … , n+1}	k	θ	$\Gamma \cup \{\psi\} \vdash \theta$

Vi kan då uppfatta premissen på rad n+1 som en provisorisk premiss och använda $\rightarrow$I för att eliminera denna premiss från premissmängden.

{1, … , n}	k+1	$\psi \rightarrow \theta$	n+1, k, $\rightarrow$I

Därmed är det klart att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \vdash \psi \rightarrow \theta$, som ger att $\Gamma \vdash \psi \rightarrow \theta$, eftersom $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \subseteq \Gamma$.

$\Leftarrow$/ Antag att $\Gamma \vdash \psi \rightarrow \theta$.

Som ovan utnyttjar vi att varje härledning är ändlig, dvs använder högst ändligt många premisser i Γ i härledningen av $\psi \rightarrow \theta$. Antag att dessa är $\varphi_1, \varphi_2, \dots, \varphi_n$. Härledningen i antagandet har då följande form

{1}	1	φ_1	Premiss
{2}	2	φ_2	Premiss
:	:	:	:
{n}	n	φ_n	Premiss
:	:	:	:
{1, … , n}	k	$\psi \rightarrow \theta$	$\Gamma \vdash \psi \rightarrow \theta$

Vi lägger nu till ψ som ytterligare premiss, och får på nästa rad

{k+1}	k+1	ψ	Premiss
{1, … , n, k+1}	k+2	θ	k, k+1, $\to$E

Därmed är det klart att $\{\varphi_1, \varphi_2, \ldots, \varphi_n, \psi\} \vdash \theta$, som ger att $\Gamma \cup \{\psi\} \vdash \theta$, eftersom $\{\varphi_1, \varphi_2, \ldots, \varphi_n\} \subseteq \Gamma$.

På motsvarande sätt som vi kan skriva S + Γ om vi utökar det formella systemet S med axiomen eller premissmängden Γ, kan vi skriva $\Gamma + \psi$ om vi utökar premissmängden Γ med satsen ψ. Här använder vi alltså $\Gamma + \psi$ som en förkortning av $\Gamma \cup \{\psi\}$.

Detta resultat är inte så viktigt i vårt formella system som det presenterats här, eftersom systemet innehåller en så kraftfull uppsättning härledningsregler. Vi ska se i avsnitt 4 att i formella system med mer sparsmakad vokabulär och uppsättning härledningsregler, så är deduktionsteoremet utomordentligt användbart. Vi illustrerar hur teoremet kan användas för att bevisa ytterligare en härledd deduktionsregel.

Exempel 36: Visa att $\{\varphi \to \psi, \psi \to \theta\} \vdash \varphi \to \theta$

Lösning:

{1}	1	$\varphi \to \psi$	Premiss
{2}	2	$\psi \to \theta$	Premiss
{3}	3	φ	Premiss
{1, 3}	4	ψ	1, 3, $\to$E
{1, 2, 3}	5	θ	2, 4, $\to$E

$\therefore$ $\{\varphi \to \psi, \psi \to \theta, \varphi\} \vdash \theta$, varpå deduktionsteoremet ger $\{\varphi \to \psi, \psi \to \theta\} \vdash \varphi \to \theta$.

Detta exempel motiverar införandet av

$$\begin{array}{l} \varphi \to \psi \\ \underline{\psi \to \theta} \\ \varphi \to \theta \end{array}$$

som ytterligare en härledd deduktionsregel. Vi benämner denna *hypotetisk syllogism* (HS).

Vi visar i nästa exempel att man i ett formellt system, eller i en deduktion, som innehåller en motsägelse kan härleda vilken sats som helst. Exemplet ger också en teknik att plocka fram en önskad sats om man vet att ett formellt system, eller en härledning, innehåller en motsägelse.

Exempel 37: Vi antar att vårt formella system innehåller en motsägelse, dvs en sats φ sådan att både φ och $\neg\varphi$ är bevisbara i systemet. Vi ska sedan visa att $\{\varphi, \neg\varphi\} \vdash \psi$, där ψ är en godtycklig satslogisk formel.

Lösning:

{1}	1	φ	Premiss
{2}	2	$\neg\varphi$	Premiss
{3}	3	$\neg\psi$	Prov premiss (för $\neg$E)
{1, 2}	4	$\perp$	1, 2, $\perp$I
{1, 2}	5	ψ	3, 4, $\neg$E

$\therefore$ $\{\varphi, \neg\varphi\} \vdash \psi$, och därmed är det klart att vi kan härleda vilken formel vi vill, givet att deduktionen innehåller en motsägelse. Formlerna på rad 1 och 2 behöver inte vara premisser. De kan ha dykt upp i en härledning på något annat sätt.

Exempel 38: Visa att $\neg(\varphi \rightarrow \psi) \vdash \varphi \wedge \neg\psi$.

Lösning:

{1}	1	$\neg(\varphi \rightarrow \psi)$	Premiss
{2}	2	$\neg\varphi$	Prov premiss (för $\neg$E)
{3}	3	φ	Prov premiss (för $\rightarrow$I)
{2, 3}	4	ψ	2, 3, Ex 37
{2}	5	$\varphi \rightarrow \psi$	3, 4, $\rightarrow$I
{1, 2}	6	$\perp$	1, 5, $\perp$I
{1}	7	φ	2, 6, $\neg$E
{8}	8	ψ	Prov premiss (för $\neg$I)
{9}	9	φ	Prov premiss (för $\rightarrow$I)
{10}	10	$\neg\psi$	Prov premiss (för $\neg$E)
{8, 10}	11	$\perp$	8, 10, $\perp$I
{8}	12	ψ	10, 11, $\neg$E
{8}	13	$\varphi \rightarrow \psi$	9, 12, $\rightarrow$I
{1, 8}	14	$\perp$	1, 13, $\perp$I
{1}	15	$\neg\psi$	8, 14, $\neg$I
{1}	16	$\varphi \wedge \neg\psi$	7, 15, $\wedge$I

Därmed är det äntligen klart att $\neg(\varphi \rightarrow \psi) \vdash \varphi \wedge \neg\psi$. Det finns smartare sätt att genomföra denna härledning. Försök hitta dylika. Strategin ovan är att härleda φ (raderna 2 – 7) respektive $\neg\psi$ (raderna 8 - 15) för att därefter använda $\wedge$I. Strategin för att härleda φ är att antaga $\neg\varphi$ för att härleda $\perp$. Den enda sats vi kan få en motsägelse från är premissen, varför vi försöker härleda $\varphi \rightarrow \psi$ (raderna 3 – 5). Vi använder sedan motsvarande strategi för att härleda $\neg\psi$.

Vi inför även detta resultat som en härledd deduktionsregel.

$$\frac{\neg(\varphi \rightarrow \psi)}{\varphi \wedge \neg\psi}$$

Vi benämner den *negerad implikation* (NI).

Härledda deduktionsregler, sammanfattning

Vi har då följande uppsättning härledda deduktionsregler.

Regel				Namn	Förkortning
$\dfrac{\neg(\varphi \vee \psi)}{\neg\varphi \wedge \neg\psi}$	$\dfrac{\neg\varphi \wedge \neg\psi}{\neg(\varphi \vee \psi)}$	$\dfrac{\neg(\varphi \wedge \psi)}{\neg\varphi \vee \neg\psi}$	$\dfrac{\neg\varphi \vee \neg\psi}{\neg(\varphi \wedge \psi)}$	De Morgans Lagar	DM
$\dfrac{\varphi \vee \psi \quad \neg\varphi}{\psi}$	$\dfrac{\varphi \vee \psi \quad \neg\psi}{\varphi}$			Disjunktiv Syllogism	DS
$\dfrac{\varphi \to \psi \quad \psi \to \theta}{\varphi \to \theta}$				Hypotetisk Syllogism	HS
$\dfrac{\neg\neg\varphi}{\varphi}$	$\dfrac{\varphi}{\neg\neg\varphi}$			Dubbel Negation	DN
$\dfrac{\varphi \to \psi \quad \neg\psi}{\neg\varphi}$				Modus Tollens	MT
$\dfrac{\neg(\varphi \to \psi)}{\varphi \wedge \neg\psi}$				Negerad Implikation	NI

Vi avslutar detta avsnitt med ytterligare några exempel.

Exempel 40: Visa att $\vdash (\varphi \to \psi) \vee (\psi \to \varphi)$.

Lösning:

{1}	1	$\neg((\varphi \to \psi) \vee (\psi \to \varphi))$	Prov premiss (för $\neg$E)
{1}	2	$\neg(\varphi \to \psi) \wedge \neg(\psi \to \varphi)$	1, DM
{1}	3	$\neg(\varphi \to \psi)$	2, $\wedge$E
{1}	4	$\neg(\psi \to \varphi)$	2, $\wedge$E
{1}	5	$\varphi \wedge \neg\psi$	3, NI
{1}	6	$\psi \wedge \neg\varphi$	4, NI
{1}	7	φ	5, $\wedge$E
{1}	8	$\neg\varphi$	6, $\wedge$E
{1}	9	$\perp$	7, 8, $\perp$I
$\varnothing$	10	$(\varphi \to \psi) \vee (\psi \to \varphi)$	1, 9, $\neg$E

Vi har då att $\vdash (\varphi \to \psi) \vee (\psi \to \varphi)$.

Exempel *41*: Visa att $\neg(\varphi \leftrightarrow \psi) \vdash \varphi \leftrightarrow \neg\psi$.

Lösning: För att härleda ekvivalensen $\varphi \leftrightarrow \neg\psi$, härleder vi implikationerna $\varphi \to \neg\psi$ och $\neg\psi \to \varphi$, och använder därefter $\leftrightarrow$I.

{1}	1	$\neg(\varphi \leftrightarrow \psi)$	Premiss
{2}	2	φ	Prov premiss (för $\to$I)
{3}	3	ψ	Prov premiss (för $\neg$I)
{3}	4	$\varphi \to \psi$	2, 3, $\to$I
{2}	5	$\psi \to \varphi$	2, 3, $\to$I
{2, 3}	6	$\varphi \leftrightarrow \psi$	4, 5, $\leftrightarrow$I
{1, 2, 3}	7	$\perp$	1, 6 $\perp$I
{1, 2}	8	$\neg\psi$	3, 7, $\neg$I
{1}	9	$\varphi \to \neg\psi$	2, 8, $\to$I
{10}	10	$\neg\psi$	Prov premiss (för $\to$I)
{11}	11	$\neg\varphi$	Prov premiss (för $\neg$E)
{11}	12	$\varphi \to \psi$	11, Se nedan
{10}	13	$\psi \to \varphi$	10, Se nedan
{10, 11}	14	$\varphi \leftrightarrow \psi$	12, 13, $\leftrightarrow$I
{1, 10, 11}	15	$\perp$	1, 14 $\perp$I
{1, 10}	16	φ	11, 15, $\neg$E
{1}	17	$\neg\psi \to \varphi$	10, 16, $\to$I
{1}	18	$\varphi \leftrightarrow \neg\psi$	9, 17, $\leftrightarrow$I

Då är det klart att $\neg(\varphi \leftrightarrow \psi) \vdash \varphi \leftrightarrow \neg\psi$.

Tänk noga genom varför det är tillåtet att göra som på raderna 4 och 5.Vad gäller raderna 12 och 13, så är det en variant av Ex 37. Hela härledningen av rad 13 kan ske på följande sätt.

{1}	1	$\neg\psi$	Premiss
{2}	2	ψ	Prov premiss (för $\to$I)
{3}	3	$\neg\varphi$	Prov premiss (för $\neg$E)
{1, 2}	4	$\perp$	1, 2 $\perp$I
{1, 2}	5	φ	3, 4, $\neg$E
{1}	6	$\psi \to \varphi$	2, 5, $\to$I

Så är det klart att $\neg\psi \vdash \psi \to \varphi$.

På motsvarande sätt visas att formeln på rad 12 följer av formeln på rad 11. Det är ofta praktiskt vid långa härledningar att bryta ut delhärledningar som vi gjort ovan. Vi kan säga att vi härleder en ny deduktionsregel, och sedan använder den i uppgiften. Det är ofta praktiskt att göra så, om ett och samma mönster finns med flera gånger, eller om man kan få själva härledningen att bli mer överskådlig på så sätt.

Några metalogiska resultat

Vi har påpekat att ett formellt system skall vara adekvat i den bemärkelsen att man i det formella systemet ska kunna bevisa alla tautologier och inget annat. Vi ska nu formulera några metalogiska resultat om vårt satslogiska system. Vi bevisar inte dessa resultat, bortsett från ett av dem, utan återkommer senare för att bevisa motsvarande resultat för ett formellt system för predikatlogik. Vi inleder med några definitioner.

Definition 19: Ett formellt system T är *sunt* om alla teorem är logiska sanningar, dvs om $T \vdash \varphi \Rightarrow \models \varphi$ för alla vbf φ. T är *fullständigt* om alla logiska sanningar är teorem, dvs om $\models \varphi \Rightarrow T \vdash \varphi$.

För satslogikens del är det tautologin som är den logiska sanningen. Ett system som är både sunt och fullständigt är *adekvat*. Kom ihåg att vi benämnt vårt formella system för naturlig deduktion i satslogik S.

Teorem 11: S är sunt.

Teorem 12: S är fullständigt.

Definition 20: Ett formellt system T är *konsistent*, om det inte finns någon vbf φ sådan att T bevisar både φ och $\neg\varphi$, dvs om det inte finns någon vbf φ sådan att $T \vdash \varphi$ och $T \vdash \neg\varphi$. Vi säger också att en satsmängd Γ är *konsistent*, om det inte finns någon vbf φ sådan att $\Gamma \vdash \varphi$ och $\Gamma \vdash \neg\varphi$.

Ovanstående definition innebär att T är inkonsistent om och endast om $T \vdash \bot$, och att en satsmängd Γ är inkonsistent om och endast om $\Gamma \vdash \bot$.

Teorem 13: S är konsistent.

Att ett formellt system är konsistent innebär att ingen motsägelse kan härledas i systemet. Det är självfallet viktigt, eftersom vi ovan sett att man i ett system som innehåller motsägelser kan härleda vad som helst.

Definition 21: Ett formellt system T är *avgörbart* om det finns en mekanisk metod med hjälp av vilken man kan avgöra huruvida en given vbf är teorem i T eller ej.

Definitionen ovan är ganska oprecis såtillvida att begreppet 'mekanisk metod' inte är definierat. Begreppet kan definieras precis, till exempel med hjälp av begreppet 'turingmaskin'. Begreppet 'mekanisk metod' innebär grovt att det inte går åt någon snillrikhet för att utföra kontrollen. Den skall i princip kunna utföras av en dator, om denna har obegränsat med tid och minne, och dessutom fungerar felfritt.

Teorem 14: S är avgörbart.

Bevis: Låt φ vara en godtycklig vbf. Undersök huruvida $\models \varphi$. Observera att tabellmetoden är en mekanisk metod! Om då $\models \varphi$, så ger fullständigheten att $\vdash \varphi$, och om $\not\models \varphi$, så ger sundheten att S inte bevisar φ.

Exempel 42: Undersök huruvida det finns ett bevis för formeln

a/ $((\varphi \vee \neg\varphi) \to \psi) \to \psi$,

b/ $(\neg A \wedge B \to C) \to (A \to B \vee C)$.

Om ett bevis finns, konstruera ett, och om bevis inte finns så konstruera en värdering i vilken formeln är falsk.

Lösning: a/ Snabbmetoden ger att formeln är en tautologi. Då ger fullständigheten hos S att det finns ett bevis för formeln.

{1}	1	$(\varphi \vee \neg\varphi) \to \psi$	Prov premiss (för $\to$I)
{2}	2	$\neg(\varphi \vee \neg\varphi)$	Prov premiss (för $\neg$E)
{2}	3	$\neg\varphi \wedge \neg\neg\varphi$	2, DM
{2}	4	$\neg\varphi$	3, $\wedge$E
{2}	5	$\neg\neg\varphi$	3, $\wedge$E
{2}	6	$\perp$	4, 5, $\perp$I
$\varnothing$	7	$\varphi \vee \neg\varphi$	2, 6, $\neg$E
{1}	8	ψ	1, 7, $\to$E
$\varnothing$	9	$((\varphi \vee \neg\varphi) \to \psi) \to \psi$	1, 8, $\to$I

Därmed är beviset klart.

b/ Betrakta en värdering i vilken $V(A) = S$ och $V(B) = V(C) = F$. I denna värdering gäller att $V((\neg A \wedge B \to C) \to (A \to B \vee C)) = F$. Formeln är alltså inte en tautologi, och då ger sundheten för S att det inte finns något bevis för formeln.

Exempel 43: Visa att $\{\neg(A \to B), A \to C, B \vee \neg C\}$ är inkonsistent.

Lösning: Vi har enligt definition 20 och anmärkningen efter denna definition att härleda $\perp$ från formelmängden.

{1}	1	$\neg(A \to B)$	Premiss
{2}	2	$A \to C$	Premiss
{3}	3	$B \vee \neg C$	Premiss
{1}	4	$A \wedge \neg B$	1, NI
{1}	5	A	4, $\wedge$E
{1}	6	$\neg B$	4, $\wedge$E
{1, 2}	7	C	2, 5, $\to$E
{1, 3}	8	$\neg C$	3, 6, DS
{1, 2, 3}	9	$\perp$	7, 8, $\perp$I

$\therefore\ \{\neg(A \to B), A \to C, B \vee \neg C\} \vdash \perp$.

Eftersom vi kan härleda $\perp$ från satsmängden $\{\neg(A \to B), A \to C, B \vee \neg C\}$, så kan vi härleda vilken formel som helst från denna mängd, och den måste då vara inkonsistent.

3 PREDIKATLOGIK

Satslogiskens otillräcklighet

Det satslogiska språk vi studerat ovan är inte tillräckligt kraftfullt för att analysera alla giltiga slutledningar. Betrakta exempelvis nedanstående, uppenbart giltiga, informella slutledning:

Alla människor är dödliga.
Sokrates är en människa.
Sokrates är dödlig.

När vi analyserar vi denna slutledning med satslogiska metoder, konstaterar vi att de tre ingående satserna är atomära, varför vi erhåller

$$\frac{\begin{array}{c}A\\B\end{array}}{C}$$

med uppenbara beteckningar. Denna slutledning är inte satslogiskt giltig, eftersom formeln

$$(A \wedge B) \rightarrow C$$

inte är en tautologi. Detta innebär inte att det är något fel på satslogiken som hjälpmedel för att analysera slutledningar. Att ovanstående informella slutledning är giltig beror på den interna uppbyggnaden av respektive sats i slutledningen, samt på samspelet mellan satserna. Med satslogik kan inte den inre strukturen i respektive sats uppvisas, eftersom satserna är just satslogiskt atomära. Vad som behövs är en mer kraftfull logik med hjälp av vilken den uppbyggnad satser som ovanstående har kan analyseras. För att använda en liknelse så kan man inte fånga små fiskar (predikatlogiskt giltiga slutledningar) med ett grovt nät (satslogikens analysmetoder). Vi behöver därför utvidga det satslogiska språket till ett kraftfullare språk, predikatlogiska språk.

När vi uppvisar den satslogiska strukturen hos en sats i ett naturligt språk, så är den fullständiga påståendesatsen den minsta analysenheten. Genomför vi en predikatlogisk satsanalys, så är de minsta beståndsdelarna i analysen individer, och de egenskaper eller relationer som individerna har, respektive står i. Som en gemensam benämning på egenskaper och relationer används ordet *predikat*. Namnet *predikatlogik* kommer av att den minsta analysenheterna är individer och just predikat. Betraktar vi en sats som

Sokrates är en människa

så talar den om en individ, Sokrates, och ett predikat som tillkommer denne individ, egenskapen att vara människa. Låter vi konstanten c beteckna individen Sokrates, och predikatsymbolen P predikatet att vara en människa, så formaliseras satsen

$$P(c).$$

I ett predikatlogiskt språk kan även vissa av det naturliga språkets *kvantifikatorer* formaliseras. En kvantifikator kan sägas vara ett uttryck som anger en sorts antal, och de kvantifikatorer från naturliga språk vi vill kunna återge är *alla (varje)* och *några (det finns, minst en)* och dess språkliga varianter. Det är viktigt att notera att det finns kvantifikatorer i naturliga språk som predikatlogik inte förmår uttrycka, t ex kvantifikatorn *det finns oändligt många*. Som symbol för ”alla” används $\forall$ och som symbol för ”några” används $\exists$. Kvantifikatorerna benämns *allkvantifikator* respektive *existenskvantifikator*.

Formlerna

$\forall x P(x)$ respektive
$\exists x P(x)$,

där P är tänkt att beteckna predikatet att vara människa, utläses

Alla är människor respektive
Några är människor.

Ett viktigt argument, utöver ovanstående, för att studera predikatlogiska språk är att dylika är mycket kraftfulla. I t ex mängdteorins språk (se nedan exempel 2), som är ett predikatlogiskt språk, kan all (?) matematik uttryckas. Ett annat viktigt argument är att logikprogrammeringsspråk ofta är predikatlogiska språk. Så är exempelvis fallet med PROLOG, ett av de viktigaste AI-språken. En formalisering i pseudoPROLOG av den ovan diskuterade slutledningen skulle kunna se ut enligt följande.

Dödlig(x) ← Människa(x).
Människa(sokrates).
Dödlig(sokrates).

I premiss ett är allkvantifikatorn underförstådd, och pilen utläses ”om”.

Det härledningssystem som PROLOG utnyttjar, resolution, är ett sorts fragment av predikatlogisk härledning. Detta fragment är konstruerat så att detta system är avgörbart, till skillnad från härledningssystem för den fulla predikatlogiken, som inte är avgörbar. Predikatlogiska språk är också viktiga vid studier av andra logiker, t ex förutsätter modallogikens Kripke-semantik kunskaper i predikatlogik. Modallogik är en utvidgning av satslogik där språket utökats med de modala operatorerna *det är nödvändigt att* och *det är möjligt att*.

Predikatlogiska språk

I ett predikatlogiskt språk finns två typer av symboler, logiska och ickelogiska. De logiska symbolerna ingår i varje predikatlogiskt språk, varför vi endast behöver ange de ickelogiska symbolerna för att specificera ett predikatlogiskt språk.

I Logiska symboler

(i)	variabler:	$x_1, x_2, x_3, \ldots$	
(ii)	markörer:	(,), ,	(dvs parenteser och kommatecken)
(iii)	Konnektiver:	$\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \perp$	
(iv)	Kvantifikatorer:	$\forall, \exists$	
(v)	Identitetssymbol:	=	

II Ickelogiska symboler

(i) Konstanter:		$c_1, c_2, c_3, \ldots$
(ii) Predikatsymboler:		P^n_i, i, n $\geq$ 1, i är ett index och n anger ställighet
(iii) Funktionssymboler:		f^n_i, i, n $\geq$ 1, i är ett index och n anger ställighet

Den vokabulär som angivits ovan är riklig. I det avsnitt nedan där vi skall bevisa Gödels fullständighetssats kommer vi att använda en mer sparsmakad vokabulär. Med en omfattande vokabulär är det omständigt att bevisa påståenden om ett formellt system, men en rik vokabulär gör det enklare att använda ett system.

Vad gäller de logiska symbolerna ovan skulle vi kunna ersätta de oändligt många variabelsymbolerna med två, nämligen x och ’. Vi kan då teckna oändligt många variabler enligt följande: x’, x’’, x’’’, x’’’’ osv. Vi kommer att skriva formler i infixform och behöver då parenteser. Väljer man att skriva formler i postfix-, eller prefixform, skulle man klara sig utan parenteser. Även kommatecknet är till för att formler skall vara mer läsbara. Ovan har vi konstaterat att exempelvis $\{\neg, \rightarrow\}$ är en fullständig konnektivmängd, och vi skulle alltså kunna klara oss med dessa två konnektiver. Existenskvantifikatorn kan definieras med hjälp av allkvantifikatorn, eftersom

$$\exists x\varphi \Leftrightarrow \neg\forall x\neg\varphi$$

för varje formel φ. För identitet gäller slutligen att det är en tvåställig relation, och den skulle därför kunna införas med hjälp av en ickelogisk predikatsymbol.

Vad gäller de ickelogiska symbolerna, kan man införa motsvarande rationaliseringar. Konstanterna är tänkta att vara namn på individer, predikatsymbolerna på predikat (egenskaper och relationer) och funktionssymbolerna är tänkta att vara namn på funktioner.

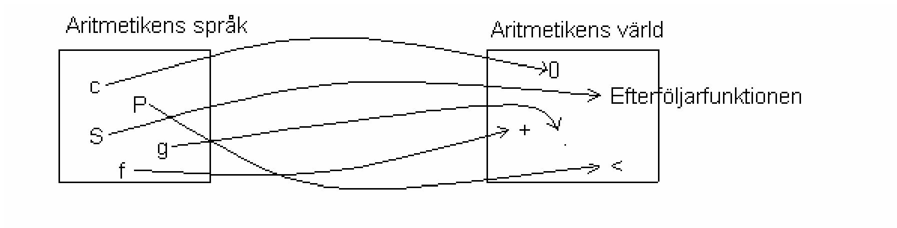
Det är opraktiskt att slaviskt följa den formella strukturen vi givit ovan. Bara vi kommer ihåg, i situationer när vi vill genomföra någon form av bevisning om ett predikatlogiskt språk, att det formella språket innehåller de givna symbolerna, så kan vi vara lite flexibla när vi skriver formler för att genom lämpligt val av symboler underlätta läsning av formler. Vi kommer t ex att använda oss av variabelsymbolerna x, y, z trots att dessa formellt inte ingår i något predikatlogiskt språk. På motsvarande sätt kommer vi att använda andra symboler än de ovan givna för individer, predikat och funktioner. Vi utelämnar också index och ställighet om ingen oklarhet uppstår på så sätt.

Innan vi formulerar definitioner av välbildade formler etc, ger vi först några exempel på predikatlogiska språk. Av pedagogiska skäl ger vi i exemplen också en tänkt tolkning av språket, dvs vi ger en struktur eller ”värld”, som språket är tänkt att kunna beskriva (vissa egenheter hos).

Exempel 1: Vi låter $L_1 = \{c_1, P^2_1, f^1_1, f^2_1, f^2_2\}$. Så här skulle språket vi avser konstruera formellt kunna beskrivas, men de formler vi då skulle teckna är mödosamma att läsa. Vi låter i stället språket vara givet genom $L_2 = \{c, P^2, S^1, f^2, g^2\}$, där c är en konstant, P är en tvåställig predikatsymbol, S är en enställig funktionssymbol och f respektive g är tvåställiga funktionssymboler. Observera att vi måste ange vilken typ av symbol respektive symbol är, eftersom vi frångått konventionerna som var givna ovan. Eftersom inga missförstånd kan uppstå, om vi utelämnar ställighetsangivelsen, så gör vi så. L_1 (L_2) är tänkt att vara ett språk i vilket aritmetiska påståenden kan uttryckas.

Symbol Tänkt tolkning

c	Talet noll
P	Relationen är mindre än
S	Efterföljarfunktionen, dvs S(x) = x + 1
f	Additionsfunktionen
g	Multiplikationsfunktionen



Pilarna illustrerar den tänkta tolkningen. I detta språk kan vi bilda formler som

Formel	Tänkt tolkning
$f(S(c), S(S(c))) = S(S(S(c)))$	$1 + 2 = 3$
$\neg g(c, S(c)) = c$	$0 \cdot 1 \neq 0$
$P(c, S(c))$	$0 < 1$
$g(x_1, x_2) = g(x_2, x_1)$	$x_1 \cdot x_2 = x_2 \cdot x_1$
$\exists x_1(g(x_1, x_1) = S(c))$	$\exists x_1(x_1 \cdot x_1 = 1)$

Ovanstående språk, aritmetikens språk, är ett av de viktigaste i matematisk logik. Ett annat viktigt språk är mängdteorins.

Exempel 2: Vi låter $L_3 = \{P^2\}$, där P är en tvåställig predikatsymbol, och har den tänkta tolkningen $\in$.

Ex på formler i L_3	Tänkt tolkning
$P(x_1, x_2)$	$x_1 \in x_2$
$\exists x_1 \forall x_2 \neg P(x_2, x_1)$	$\exists x \forall y\ y \notin x$

Exempel 3: Vi låter $L_4 = \{c_1, c_2, P^1, R^2\}$

I detta exempel har vi en tänkt domän, dvs en ”värld” som innehåller en uppsättning individer. Variabler och konstanter är tänkta att peka ut element i denna domän. Vi betecknar domänen med M och definierar

$$M = \{x : x \text{ är tillsvidaranställd vid INV vid HIS}\}$$

De ickelogiska symbolerna är tänkta att ha följande tolkning:

c_1	Jörgen,
c_2	Mikael,
$P(x)$	x är matematiker,
$R(x, y)$	x är snällare än y.

Ex på formler i L_4	Tänkt tolkning
$\exists xR(x, c_1)$	Någon är snällare än Jörgen
$\forall x(P(x) \rightarrow (R(c_1, x) \vee c_1 = x))$	?????

Vi har ovan använt vår intuition om hur välbildade formler i ett predikatlogiskt språk ser ut, men vi skall nu se till att denna brist åtgärdas. Själva definitionen sker i tre steg, och vi definierar först *term*, där termer är tänkta att peka ut individer i en domän, därefter *atomära formler*, som kan säga att två termer är lika eller att ett predikat (egenskap eller relation) tillkommer en eller flera termer, och slutligen *välbildad formel*.

Definition 1: Definition av *term*.

(i) $x_1, x_2, \ldots$ är termer, dvs alla variabler är termer,
(ii) $c_1, c_2, \ldots$ är termer, dvs alla konstanter är termer,
(iii) Om f är en n-ställig funktionssymbol och, $t_1, t_2, \ldots, t_n$ är termer, så är $f(t_1, t_2, \ldots, t_n)$ en term.

Definition 2: Definition av *atomär formel*.

(i) Om t_1 och t_2 är två termer, så är $t_1 = t_2$ en atomär formel,
(ii) Om P är en n-ställig predikatsymbol, och $t_1, t_2, \ldots, t_n$ är termer, så är $P(t_1, t_2, \ldots, t_n)$ en atomär formel.
(iii) $\perp$ är en atomär formel.

Definition 3: Definition av *välbildad formel* (vbf).

(i) Alla atomära formler är välbildade formler,
(ii) Om φ är en välbildad formel, så är $\neg\varphi$ en välbildad formel,
(iii) Om φ och ψ är välbildade formler, så är $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$ och $(\varphi \leftrightarrow \psi)$ välbildade formler,
(iv) Om φ är en välbildad formel och v en variabel, så är $\exists v\varphi$ och $\forall v\varphi$ välbildade formler.

Definitionerna 1 och 3 ovan är rekursiva (induktiva) där punkterna (i) och (ii) i definition 1 är grundsteg och punkten (i) i definition 3 är grundsteg. Definitionerna ovan är viktiga, eftersom övriga definitioner och bevis utgår från dem. De används också för att undersöka om en given formel är välbildad eller ej.

Exempel 4: Undersök om $((P_1^1(c_3) \wedge \neg P_1^1(c_1)) \rightarrow \forall x_1 P_5^2(x_1, c_3))$ är en välbildad formel.

Vi konstaterar att $\rightarrow$ är huvudoperator i formeln, som då består av två delformler

(1) $(P_1^1(c_3) \wedge \neg P_1^1(c_1))$ och
(2) $\forall x_1 P_5^2(x_1, c_3)$.

Formel (1) har huvudoperatorn $\wedge$ och består av delformlerna

(3) $P_1^1(c_3)$ och
(4) $\neg P_1^1(c_1)$.

Formel (3) innehåller inga kvantifikatorer eller konnektiver, och vi har därför att undersöka om den är atomär. Den består av en enställig predikatsymbol, och har ett argument, och eftersom c_3 är en konstant, så är argumentet en term och därmed är det klart att (3) är atomär. Vad sedan gäller (4), så har den huvudoperatorn $\neg$ och består av delformeln $P_1^1(c_1)$, som i sin tur är atomär, eftersom predikatsymbolen är enställig och konstanten c_1 är en term. Det är då klart att (3) och (4) är välbildade formler, och i och med detta, så är också (1) välbildad.
Formel (2) är allkvantifierad, och innehåller delformeln $P_5^2(x_1, c_3)$, som i sin tur är atomär, eftersom P_5^2 är en tvåställig predikatsymbol och x_1 respektive c_3, de två argumenten, är termer. Det är då klart att (2) är en välbildad formel, och därmed att den givna formeln är välbildad.

Analysen i exemplet ovan utförs enklast i ett träd i vilket den givna formeln är rot och övriga hörn sedan är delformler. Löven innehåller då de atomära formlerna, som i sin tur analyseras med hjälp av träd i vilket löven är termer, som i sin tur analyseras i träd.

Vi avslutar med ytterligare lite terminologi.

Vi har inledningsvis benämnt symbolerna $\forall$ och $\exists$ för kvantifikatorer. Samma ord kommer vi nu att använda även för uttrycken $\forall v$ och $\exists v$ där v är en variabel. Vid tveksamheter kommer mångtydigheten att undvikas på så sätt att vi använder ordet *kvantifikatorsymboler* om $\forall$ och $\exists$.

Definition 4: En *kvantifikator* är en kvantifikatorsymbol följd av en variabel.

Exempel 5: I ett formellt predikatlogiskt språk är följande uttryck exempel på kvantifikatorer $\forall x_7$, $\forall x_1$, $\exists x_3$ etc. Observera att $\exists c_2$ inte är en kvantifikator, eftersom c_2 inte är en variabel. Vi kommer frekvent, för att slippa använda index, att som tidigare "fuska" och använda även x, y och z som variabler, och vi kommer då att uppfatta t ex $\exists x$ etc som kvantifikatorer.

Definition 5: Med en kvantifikators *räckvidd* i en formel φ menas kvantifikatorn själv tillsammans med den kortaste delen av formeln som följer efter kvantifikatorn och som själv är en välbildad formel.

Exempel 6: I nedanstående två formler är respektive kvantifikators räckvidd understruken.

$\underline{\forall x(R(x, y) \to P(x))}$

$\underline{\forall x R(x, y)} \to P(x)$

Definition 6: En variabel v är *bunden* om den förekommer inom räckvidden av en kvantifikator $\forall v$ eller $\exists v$. En variabel v är *fri* om den inte är bunden.

Exempel 7: I formeln $\forall x(R(x, y) \to P(x))$, är alla tre förekomsterna av x bundna medan förekomsten av y är fri. I formeln $\forall x R(x, y) \to P(x)$, är de två första förekomsterna av x bundna, medan övriga variabelförekomster är fria.

Definition 7: En *sats* (*sluten formel*) är en välbildad formel som saknar fria variabler. En formel som innehåller minst en fri variabel kallas *öppen*.

Exempel 8: Båda formlerna i exempel 6 ovan är öppna, eftersom y är fri i båda formlerna. Formeln i exempel 4 är en sats eftersom samtliga variabelförekomster (den enda variabeln är x_1) är bundna.

Om φ är en formel, så indikerar vi med uttrycket $\varphi(v_1, v_2, \dots, v_n)$ att de fria variablerna i φ finns bland variablerna $v_1, v_2, \dots, v_n$. Observera att φ inte behöver innehålla alla dessa variabler, eller ens någon av dem. Normalt strävar vi efter att ange precis vilka variabler, om några, som är fria.

Formalisering i predikatlogik

När man formaliserar en språklig sats i predikatlogik, så försöker man uppvisa så noggrant som möjligt den predikatlogiska struktur satsen i fråga har. Vi har, vid våra studier av formalisering i satslogik, diskuterat två formaliseringsstrategier. Strategierna benämndes *top down* (uppifrån och ner) respektive *bottom up* (nerifrån och upp). Vid användning av top-down-strategin lokaliseras först huvudoperationen i satsen, och därmed de delsatser på vilken huvudoperationen verkar. Proceduren upprepas sedan på delsatserna. Vid användning av bottom-up-strategin lokaliseras först de atomära formler satsen är uppbyggd av varefter den givna satsens logiska struktur uppvisas så troget som möjligt. Ofta är det lämpligt att använda en blandning av de två strategierna. Man börjar lämpligen med top-down-strategin för att byta till bottom-up-strategin när de atomära formler som bygger upp satsen är enkelt urskiljbara.

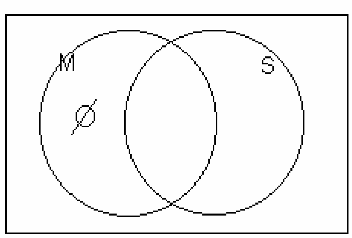
Utöver de problem vi ventilerat vid satslogisk formalisering, måste vi nu även kunna hantera kvantifikatorer i naturliga språk, och vi utgår därvid från följande fyra huvudtyper av satser som vi illustrerar med exempel.

Satstyp 1: Alla matematiker är snälla.

Detta är en generell sats som utsäger att alla individer som har en viss egenskap har en annan egenskap. Satsen utsäger följande:

För varje individ x gäller att om x är en matematiker, så är x snäll.

Satsen säger alltså att mängden matematiker är en delmängd till mängden snälla varelser. Detta kan illustreras i ett venndiagram enligt följande.



Här betecknar symbolen $\varnothing$ i mängden $M - S$ att $M - S = \varnothing$.

Låter vi $M(x)$ stå för x är en matematiker, och $S(x)$ för x är snäll, så får vi följande formalisering

$$\forall x(M(x) \rightarrow S(x)).$$

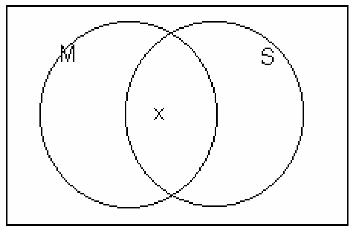
Observera att allkvantifikatorn är associerad med en (materiell) implikation.

Satstyp 2: Några matematiker är snälla.

Detta är en existenssats och den utsäger följande:

Det finns en individ x sådan att x är både matematiker och x är snäll.

Satsen säger att mängden matematiker och mängden snälla varelser inte är tom. Detta kan vi illustrera i ett venndiagram enligt följande.



Krysset betecknar att $M \cap S \neq \emptyset$, det vill säga att det finns minst en individ i mängden $M \cap S$. Detta svarar då mot att minst en individ har både egenskapen att vara matematiker och att vara snäll. Använder vi samma beteckningar som ovan får vi

$$\exists x(M(x) \wedge S(x)).$$

Observera att existenskvantifikatorn är associerad med konjunktionen.

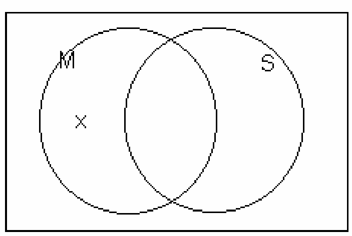
Satstyp 3: Några matematiker är inte snälla.

Denna sats utsäger följande:

Det finns en individ x sådan att x är både matematiker och x är inte snäll,

dvs det finns matematiker, som inte är snälla.

Satsen säger att snittet av mängden matematiker och mängden ickesnälla varelser inte är tom. Detta kan vi illustrera i ett venndiagram enligt följande.



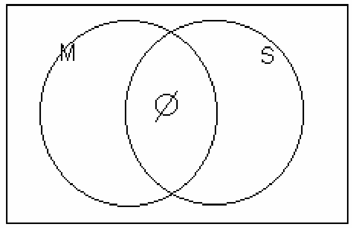
Formalisering med beteckningar enligt ovan blir

$$\exists x(M(x) \wedge \neg S(x)).$$

Observera att denna formel är logiskt ekvivalent med $\neg\forall x(M(x) \rightarrow S(x))$. Detta kan visas med tekniker som presenteras senare. Resultatet är naturligt eftersom satserna ”Alla matematiker är snälla.” respektive ”Några matematiker är inte snälla.” är varandras *kontradiktoriska motsatser*, dvs satserna har olika sanningsvärde. Detta framgår också i diagrammen där det ena diagrammet illustrerar påståendet att en mängd är tom, och det andra att samma mängd inte är tom.

Satstyp 4: Inga matematiker är snälla.

Denna sats utsäger följande:



För varje individ x gäller att om x är en matematiker, så är x inte snäll.

En alternativ formulering är följande: Det är inte så att det finns någon individ x sådan att x både är matematiker och x är snäll.

Satsen säger alltså att snittet av mängden matematiker och mängden snälla varelser är tom, dvs att mängderna är disjunkta. Detta illustrerar vi i ett venndiagram enligt nedan.
Med beteckningarna ovan får vi

$\neg\exists x(M(x) \wedge S(x))$,

eller $\forall x(M(x) \rightarrow \neg S(x))$.

De båda formlerna är logiskt ekvivalenta. Tänk också på att det logiskt sett inte är någon skillnad på singularform och pluralform hos satsen. Vi skulle få exakt samma formaliseringar, om vi betraktade satsen ”Ingen matematiker är snäll.”. Observera även att satserna ”Några matematiker är snälla.” och ”Inga matematiker är snälla.” är kontradiktoriskt motsatta.

Satstyperna kallas med klassisk terminologi A, I, E respektive O-satser. Motsatsen mellan satstyperna 1 och 4 kallas *konträr*. Båda satserna kan vara falska men båda kan inte vara sanna. Vad gäller satstyperna 2 och 3, så kan båda satserna vara sanna, men inte båda falska. Denna motsats kallas *subkonträr*.

I de formaliseringsexempel som presenteras nedan används top-down-strategin till dess de atomära formler som bygger upp satsen kan urskiljas. Notera att det är satser vi formaliserar. Den resulterande formeln skall vara en sats, dvs den får inte innehålla några fria variabler. Studera exemplen nedan väldigt noga. I exemplen finns också ett antal kommentarer om problem associerade med de satser som skall formaliseras.

Exempel 9: Formalisera satsen Alla programmerare är vänliga människor.

Lösning: Denna sats är allkvantifierad, och är av satstyp 1. Vi får då

$\forall x$(x är programmerare $\rightarrow$ x är en vänlig människa).

Här är frasen i försatsen atomär, och vi ersätter den med P(x). Frasen i eftersatsen kan analyseras vidare. Den säger att x har de två egenskaperna att vara vänlig och att vara människa, och vi får då

$\forall x$(P(x) $\rightarrow$ x är vänlig $\wedge$ x är en människa).

Delfraserna i eftersatsen är atomära, och ersätts med $V(x)$ respektive $M(x)$. Den slutgiltiga formeln som representerar den givna satsen blir

$$\forall x(P(x) \rightarrow V(x) \wedge M(x)),$$

där betydelsen av predikatsymbolerna är de ovan givna.

Exempel 10: Formalisera Minst en programmerare är flintskallig.

Lösning: Detta är ett exempel på en existenssats. Vi får som för satstyp 2

$\exists$x(x är programmerare $\wedge$ x är flintskallig).

Båda delfraserna är atomära, och vi ersätter dem med $P(x)$ respektive $F(x)$. Detta ger

$$\exists x(P(x) \wedge F(x)),$$

och därmed är satsen formaliserad.

Exempel 11: Formalisera Det finns inget största naturligt tal.

Lösning: Denna sats kan uppfattas som en negation, och läses då

Det är inte så att det finns ett största naturligt tal, dvs
$\neg$(det finns ett största naturligt tal).

Alternativt kan vi läsa den som en generell sats, och får

För alla naturliga tal finns ett större naturligt tal, som ger
$\forall$x(x är ett naturligt tal $\rightarrow$ finns ett naturligt tal som är större än x).

Vi väljer den första skrivningen, och har att konstatera att satsen inuti parentesen är en existenssats, varför vi får

$\neg\exists$x(x är ett naturligt tal $\wedge$ x är större än eller lika med alla naturliga tal).

Här är den första konjunktens delfras atomär, och vi ersätter den med $N(x)$. Delfrasen i den andra konjunkten är generell, och vi får här

$\neg\exists$x(N(x) $\wedge$ $\forall$y(y är ett naturligt tal $\rightarrow$ x är större än eller lika med y)).

Den slutgiltiga formaliseringen blir, där vi använder $\geq$ i sin vanliga matematiska betydelse,

$$\neg\exists x(N(x) \wedge \forall y(N(y) \rightarrow x \geq y)).$$

Väljer vi den andra läsningen får vi

$$\forall x(N(x) \rightarrow \exists y(N(y) \wedge y > x)).$$

Försök genomföra analysen själv i detta fall. Dessa två formler kan, som sig bör, visas vara predikatlogiskt ekvivalenta

Det senaste exemplet är mer komplicerat än de två tidigare, och detta beror framför allt på att satsen innehåller två kvantifikatorer. Det är inte heller alltid fallet att alla kvantifikatorer är väl synliga. Så är fallet i exempel 11. Observera också igen att i samtliga exempel ovan är allkvantifikatorn associerad med implikationen, och existenskvantifikatorn med konjunktionen. Detta är ingen tillfällighet.

Exempel 12: Formalisera   Det finns ett minsta naturligt tal.

Lösning: Satsen är en existenssats, och därmed har vi

$$\exists x(x \text{ är ett naturligt tal} \wedge x \text{ är det minsta naturliga talet}).$$

Den första delfrasen är som i exemplet ovan atomär, och vi använder samma beteckning här. Frasen i den andra konjunkten är inte atomär. Den är en generell sats, och kan skrivas enligt följande

$$\exists x(N(x) \wedge \forall y(y \text{ är ett naturligt tal} \rightarrow y \text{ större än eller lika med } x)).$$

Detta ger

$$\exists x(N(x) \wedge \forall y(N(y) \rightarrow y \geq x)).$$

I detta exempel kan vi tänka oss att det, liksom i föregående, finns en oklarhet. Denna gäller huruvida ordet ”ett” i satsen är obestämd artikel eller räkneord. Med tanke på vår kunskap om de naturliga talens struktur, är det i det här exemplet rimligt att uppfatta ”ett” som räkneord. Vi vet ju att det finns exakt ett minsta naturligt tal, nämligen talet noll. I föregående exempel är det rimligt att uppfatta ”en” som obestämd artikel, eftersom det ju för varje naturligt tal finns till och med oändligt många naturliga tal som är större än det givna talet. Antar vi att $\geq$ används i sin vanliga matematiska betydelse, ligger denna innebörd emellertid i symboliken, eftersom $\geq$ är en linjär (total) ordning på de naturliga talen (se t ex Rosen, definition 3, kap 6.6). Observera emellertid att vi inte kan ersätta $\geq$ med $>$, eftersom den sats vi då får är falsk under förutsättning att den handlar om ”riktiga naturliga tal” vad det nu är. Vi kan dock skriva enligt följande

$$\exists x(N(x) \wedge \forall y(N(y) \wedge y \neq x \rightarrow y > x)).$$

Vi diskuterar i exemplen 21 till 26 hur man kan formalisera antal i predikatlogik.

Exempel 13: Formalisera satsen   En flicka är en flicka.

Lösning: Detta är ett exempel på en generell sats trots att ingen allkvantifikator ”syns”. Den uttalar sig om en egenskap som alla flickor har, dvs den säger att alla flickor är flickor, och kan formaliseras

$$\forall x(F(x) \rightarrow F(x))$$

där $F(x)$ står för x är en flicka. Här kan man fundera över om de båda förekomsterna av ordet ”flicka” verkligen betyder samma sak. Gör de det? Försök tänka ut en situation där någon påstår detta och verkligen avser säga något med frasen. Som den är formaliserad ovan säger den ju inget, eftersom den är en logisk sanning.

Exempel 14: Formalisera   Ingen programmerare kan alla språk.

Lösning: Denna sats kan, likt den i exempel 11, uppfattas på två sätt. Vi väljer här att uppfatta den som en generell sats, och får

$\forall x$(x är en programmerare $\rightarrow$ x kan inte alla språk).

Eftersatsen uppfattar vi som en existenssats och skriver

$\forall x$(P(x) $\rightarrow \exists y$(y är ett språk $\wedge$ x kan inte y)),

där P(x) står för x är programmerare. Slutligen konstaterar vi att den andra konjunkten är en negation, och vi får

$\forall x(P(x) \rightarrow \exists y(S(y) \wedge \neg K(x, y)))$,

där S(y) betecknar y är ett språk, och K(x, y) står för x kan y.
Försök själv analysera exemplet då satsen uppfattas som en negation.

Exempel 15: Formalisera satsen Primtal större än två är udda.

Lösning: Satsen är generell, och kan skrivas

$\forall x$(x är ett primtal större än två $\rightarrow$ x är udda).

Försatsen är en konjunktion, eftersatsen är atomär och vi får

$\forall x$(x är ett primtal $\wedge$ x större än två $\rightarrow$ x är udda).

Använder vi beteckningarna

2	konstant som betecknar talet två
P(x)	x är ett primtal,
x > y	x är större än y
U(x)	x är udda

får vi följande formalisering

$\forall x(P(x) \wedge x > 2 \rightarrow U(x))$.

Exempel 16: Formalisera satsen Alla som känner Pål är kända av Per.

Lösning: Satsen är generell, och i ett första steg, så får vi

$\forall x$(x känner Pål $\rightarrow$ x är känd av Per).

Vi observerar att y är känd av x är passiv form av x känner y, och låter K(x, y) stå för x känner y. Detta ger

$\forall x$(K(x, Pål) $\rightarrow$ K(Per, x)).

Notera ordningen mellan argumenten i respektive predikat. Till sist låter vi konstanten c_1 beteckna individen Pål, och c_2 individen Per. Vi får

$\forall x(K(x, c_1) \rightarrow K(c_2, x))$.

Vi fortsätter med några exempel som påvisar vikten med att skriva kvantifikatorer i korrekt ordning.

Exempel 17: Formalisera a/ Alla älskar någon,
b/ Någon älskar alla.

Lösning: Vi underförstår först att kvantifikationen sker över mängden av alla människor, och låter $Ä(x, y)$ stå för x älskar y. Vi får då

a/ $\forall x \exists y Ä(x, y)$, respektive
b/ $\exists x \forall y Ä(x, y)$.

De båda satserna har uppenbart väldigt olika betydelser, där situationen för denne någon i b är, så att säga, en aning mer ansträngd än för individerna i exempel a. De två formlerna som avbildar respektive sats logiska struktur har också helt olika innebörd, vilket vi skall kunna visa längre fram i framställningen. Vill vi sedan att kvantifikatorområdet skall framgå vilket normalt är det vettigaste, så blir formaliseringen något mer omständlig. Om vi exempelvis antar att x och y varierar över mängden människor, och vi vill att detta skall framgå i formeln, kan vi låta $M(x)$ stå för x är människa, och vi får

a/ $\forall x(M(x) \to \exists y(M(y) \wedge Ä(x, y)))$, respektive
b/ $\exists x(M(x) \wedge \forall y(M(y) \to Ä(x, y)$.

Vilken formalisering som är lämplig beror på vilket syfte man har med formaliseringen, men den senare är normalt att föredra.

Exempel 18: Formalisera satsen Alla människor tycker om någon (människa).

Lösning: Satsen är generell, och vi får i första steget

$\forall x$(x är en människa $\to$ x tycker om någon).

Eftersatsen kan skrivas om på passiv form för att vi tydligt ska kunna se att den är en existenssats. Vi får, med $M(x)$ för x är en människa

$\forall x(M(x) \to$ någon är omtyckt av x).

Detta i sin tur ger

$\forall x(M(x) \to \exists y(M(y) \wedge$ y är omtyckt av x).

Växlar vi så tillbaks till aktiv form så får vi

$\forall x(M(x) \to \exists y(M(y) \wedge$ x tycker om y),

som med $T(x, y)$ för x tycker om y ger oss

$\forall x(M(x) \to \exists y(M(y) \wedge T(x, y)))$.

Observera att det inte på något sätt är nödvändigt att göra dessa växlingar mellan aktiv och passiv form, om man inte tycker att det klargör något.

Exempel 19: Formalisera Någon människa tycker om alla.

Lösning: Satsen i detta exempel innehåller samma predikat som satsen i exemplet ovan, och vi använder därför samma beteckningar. Vi konstaterar att satsen är en existenssats och genomför inte resonemanget så omständigt som ovan. Vi får i tur och ordning

$\exists x(M(x) \wedge x$ tycker om alla),

$\exists x(M(x) \wedge \forall y(M(y) \rightarrow x$ tycker om $y))$,

$\exists x(M(x) \wedge \forall y(M(y) \rightarrow T(x, y)))$.

Exempel 20: Formalisera Ingen människa tycker om någon.

Lösning: Vi uppfattar satsen som en negation och får då

$\neg$(någon människa tycker om någon)

Satsen inom parentesen är en existenssats, och vi får successivt med beteckningar som ovan

$\neg\exists x(M(x) \wedge x$ tycker om någon)

$\neg\exists x(M(x) \wedge \exists y(M(y) \wedge x$ tycker om $y)$

$\neg\exists x(M(x) \wedge \exists y(M(y) \wedge T(x, y))$

I predikatlogik kan vi formalisera antal, det vill säga att si och så många har en viss egenskap. Vi illustrerar tekniken med en svit av exempel.

Exempel 21: Formalisera Det finns (minst) en jultomte.

Satsen är en existenssats, och med $J(x)$ för x är en jultomte, erhåller vi

$\exists xJ(x)$.

I denna sats uttrycks inte antal. Den utsäger endast att det finns minst en individ med nämnda egenskap. Jämför med satsen

Det finns (exakt) en jultomte.

Denna sats säger att det finns ett visst antal jultomtar, nämligen en. En möjlig formaliseringsidé för uttrycka detta är att skriva om satsen på följande sätt

Det finns (minst) en jultomte, och för varje individ som är en jultomte, gäller att denne är identisk med den förstnämnde.

Med denna omskrivning, får vi följande

$\exists x(J(x) \wedge$ varje jultomte är identisk med $x)$

$\exists x(J(x) \wedge \forall y(J(y) \rightarrow y = x))$.

I satser av den typ som diskuteras i ovanstående exempel finns samma tvetydighet som nämndes i samband med exempel 12, nämligen att ordet ”en” är mångtydigt. Ordet kan dels vara obestämd artikel, dels räkneord. I exemplet ovan är den avsedda betydelsen tydliggjord med fraserna ”minst en” respektive ”exakt en”. Motsvarande problem finns egentligen även i exemplen nedan. Det är svårt att exempelvis hävda att det är falskt att påstå att det står två kor i hagen, om det är så att det t ex finns 17 kor i hagen.

Exempel 22: Formalisera Det finns minst två jultomtar.

Lösning: Satsen uttalar sig om existensen av minst två individer som har egenskapen att vara jultomte. Detta kan uttryckas på följande sätt med beteckningar som i exemplet ovan.

$$\exists x \exists y (J(x) \wedge J(y) \wedge x \neq y)$$

där den sista konjunkten är nödvändig för att garantera att det finns minst två individer som har egenskapen att vara jultomte. Hade vi inte haft denna konjunkt med i formeln hade variablerna x och y kunnat peka ut samma individ.

Betrakta sedan satsen

Det finns (exakt) två jultomtar.

Denna sats utsäger att det finns minst två jultomtar och allt som har jultomteegenskapen är identisk med någon av två individer med denna egenskap. Detta kan formaliseras

$$\exists x \exists y (J(x) \wedge J(y) \wedge x \neq y \wedge \forall z (J(z) \rightarrow z = x \vee z = y)).$$

Exempel 23: Formalisera satsen Det finns högst en jultomte.

Lösning: Denna sats är liktydig med satsen

Det finns ingen jultomte eller det finns (exakt) en jultomte,

och denna sats kan formaliseras

$$\neg \exists x J(x) \vee \exists x (J(x) \wedge \forall y (J(y) \rightarrow y = x)).$$

Exempel 24: Formalisera satsen Det finns högst två jultomtar.

Lösning: Vi skriver om satsen och får den likabetydande satsen

Det finns ingen jultomte eller det finns exakt en jultomte eller det finns exakt två jultomtar.

Exempel 22 och 23 kombineras sedan och ger formaliseringen

$$\neg \exists x J(x) \vee \exists x (J(x) \wedge \forall y (J(y) \rightarrow y = x)) \vee \exists x \exists y (J(x) \wedge J(y) \wedge x \neq y \wedge$$
$$\wedge \forall z (J(z) \rightarrow z = x \vee z = y)).$$

Exempel 25: Formalisera satsen Det finns minst tre jultomtar.

Lösning: I detta fall får vi formeln

$$\exists x\exists y\exists z(J(x) \wedge J(y) \wedge J(z) \wedge x \neq y \wedge x \neq z \wedge y \neq z).$$

Observera att vi måste ha tre negerade identiteter för att garantera tre olika element med egenskapen att vara jultomte.

Analogt med ovan kan vi sedan formalisera exempelvis satser som Det finns exakt tre jultomtar, Det finns högst 17 sardinburkar osv.

Vi avslutar med ett exempel där antalsformaliseringen finns inuti en mer komplicerad sats.

Exempel 26: Formalisera satsen Alla människor har exakt en mor.

Lösning: Satsen är generell, och vi får först med $M(x)$ för x är en människa

$$\forall x(M(x) \rightarrow x \text{ har exakt en mor}).$$

Vi låter $R(x, y)$ stå för x är mor till y, och får följande, eftersom konsekvensen är en existenssats

$$\forall x(M(x) \rightarrow \exists y(R(y, x) \wedge y \text{ är unik})).$$

$$\forall x(M(x) \rightarrow \exists y(R(y, x) \wedge \forall z(R(z, x) \rightarrow y = z))).$$

Den avslutande allkvantifierade delformeln säger att varje individ som är mor till x är identisk med y.

Den ovan givna formaliseringen är i många sammanhang ofullständig, eftersom kvantifikatorområdet för y och z inte framgår. Man kan tänka sig att det är viktigt att det framkommer i formaliseringen att de individer som y och z får peka ut är människor. I så fall bör formaliseringen vara

$$\forall x(M(x) \rightarrow \exists y(M(y) \wedge R(y, x) \wedge \forall z(M(z) \wedge R(z, x) \rightarrow y = z))).$$

I första ordningens predikatlogik finns två kvantifikatorer, men naturligt språk innehåller många fler typer av kvantifikatorer än *minst en* och *alla*. En i matematik vanligen förekommande kvantifikator är *det finns oändligt många*. Vi såg ovan att vi kunde formalisera antal i predikatlogik, men det är omöjligt att i predikatlogik formalisera denna kvantifikator. Visserligen kan vi införa en symbol för denna kvantifikator, och vanligen används då $\exists^{\infty}$, men då lämnar vi predikatlogikens område, och det finns ett antal viktiga teorem som gäller för predikatlogik, men inte för en dylik extension av predikatlogiken. Observera emellertid att vi kan formalisera satsen

För varje naturligt tal, finns ett större naturligt tal

även om vi inte kan formalisera

Det finns oändligt många naturliga tal

inom predikatlogiken.

Om definition av tolkning och sanning i predikatlogik.

Att definiera begreppet sanning i predikatlogik är ett vanskligt företag. Den första lyckade sanningsdefinitionen formulerades i början av 30-talet av den polske logikern Alfred Tarski. Idén är egentligen enkel, den följer tankegången i exempel 1:3. Idén går under namnet *korrespondensteorin för sanning*, som finns formulerad redan hos Aristoteles. Den går ut på att en sats är sann om och endast om det i verkligheten är just så som satsen utsäger. Så är till exempel satsen ”Månen är en ost.” sann om och endast om månen är en ost. Eftersom vi vet att detta inte är fallet så vet vi också att satsen är falsk. Läs gärna det som står i inledningen om semantik innan du läser nedanstående, och försök komma ihåg att idén egentligen är enkel, men det är komplicerat att formulera denna idé precist.

Strukturer

Låt L vara ett predikatlogiskt språk. En *tolkning* (*interpretation*) av L är en hopparning av de ickelogiska symbolerna i L med lämpliga entiteter i en mängd M.

Definition 8: En *modell* (*struktur*) $\underline{M}$ för ett predikatlogiskt språk L är ett ordnat par $\underline{M} = (M, T)$ sådant att

(i) $M \neq \emptyset$, modellens *domän*.

(ii) T är en *(tolknings)funktion* sådan att
(a) $T(c) \in M$, om c är en individkonstant i L,
(b) $T(P) \subseteq M^n$, om P är en n-ställig predikatsymbol i L
(c) T(f) är en funktion från M^n till M, om f är en n-ställig funktionssymbol i L.

Funktionen T kan utläsas ”tolkningen av …”.

Om $\underline{M}$ är en struktur (modell) för ett språk L, säger vi att $\underline{M}$ är en *L-struktur*. Det språk för vilket $\underline{M}$ är en modell betecknar vi $L(\underline{M})$.

Vi skriver ofta $c^{\underline{M}}$ i stället för T(c), $P^{\underline{M}}$ i stället för T(P) och $f^{\underline{M}}$ i stället för T(f). Observera att tolkningen av en individkonstant är ett element i mängden M, att tolkningen av en n-ställig predikatsymbol är en n-ställig relation på M och att tolkningen av en n-ställig funktionssymbol är en funktion från M^n till M. Med tolkningsfunktionen anges betydelsen av en symbol, dvs vad symbolen ”pekar ut” i ”världen”. Om språket L innehåller få symboler anges ofta strukturen med domän och tolkning av respektive symbol enligt följande exempel. Om $L = \{P^2, c\}$, så skriver vi strukturen $\underline{M} = (M, P^{\underline{M}}, c^{\underline{M}})$.

En term är *sluten* om den inte innehåller några variabelförekomster. I definitionen nedan visas hur slutna termer t skall tolkas.

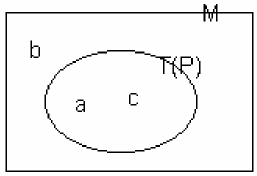
Definition 9: Om t är en individkonstant c, så gäller enligt ovan att $t^{\underline{M}} = c^{\underline{M}}$.

Om t är $f(t_1, \dots, t_n)$, där f är en n-ställig funktionssymbol och $t_1, \dots, t_n$ är slutna termer i L, låter vi $f(t_1, \dots, t_n)^{\underline{M}} = f^{\underline{M}}(t_1^{\underline{M}}, \dots, t_n^{\underline{M}})$.

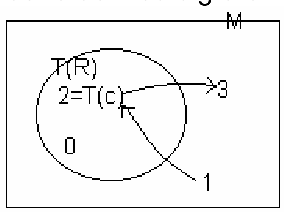
Exempel 27: Låt $L = \emptyset$, dvs språket saknar ickelogiska symboler. Då duger varje val av struktur, t ex M_1 = {Anna, Pia}, $M_2 = \{0, 7, 3\}$, M_3 = R = {reella tal} osv.

Exempel 28: Låt $L = \{P^1\}$, dvs L innehåller en enställig predikatsymbol. En möjlig struktur är då $M = \{a, b, c\}$, $P^{\underline{M}} = \{a, c\}$, dvs $P^{\underline{M}} \subseteq M$. Strukturen kan då skrivas

$\underline{M}$ = ({a, b, c}, {a, c}), och den kan illustreras med följande diagram.



Exempel 29: Låt $L = \{c, P^2, R^1\}$. En möjlig struktur för detta språk är $M = \{0, 1, 2, 3\}$, $T(c) = 2$, $T(R) = \{0, 2\}$, $T(P) = \{(1, 2), (2, 3)\}$. Observera att P är en tvåställig predikatsymbol. Även denna struktur kan illustreras med ett diagram, i vilket enställiga relationer illustreras som i exempel 28, medan tvåställiga relationer illustreras med digrafer.



Exempel 30: Aritmetikens språk

Låt $L = \{c, R^2, f^2, g^2, h^1\}$, och låt $M = N = \{0, 1, 2, 3, ...\}$.
Vi definierar T enligt följande:

$T(c) = c^{\underline{M}} = 0$ (talet noll),
$T(R) = R^{\underline{M}} = \leq$ (den vanliga ordningsrelationen),
$T(f) = f^{\underline{M}} = +$ (den tvåställiga vanliga additionsfunktionen),
$T(g) = g^{\underline{M}} = \cdot$ (den tvåställiga vanliga multiplikationsfunktionen) och
$T(h) = h^{\underline{M}} = S$ (efterföljarfunktionen, som definieras av $S(x) = x + 1$).

Med detta är t ex $f(c, h(g(c, h(c))))$ en term med tolkningen
$f^{\underline{M}}(c^{\underline{M}}, h^{\underline{M}}(g^{\underline{M}}(c^{\underline{M}}, h^{\underline{M}}(c^{\underline{M}})))) = +(0, S(\cdot(0, S(0)))) = 0 + S(0 \cdot 1)$.

När den tänkta tolkningen, som här, är ett språk för aritmetiken, använder vi gärna de vanliga symbolerna där vi markerar med ett streck under symbolen att den hör till ett språk. Vi skriver alltså $L' = \{\underline{0}, \underline{\leq}, \underline{+}, \underline{\cdot}, \underline{S}\}$ där t ex $T(\underline{0}) = 0$, dvs tolkningen av symbolen (siffran) $\underline{0}$ är talet 0. Det är viktigt att komma ihåg att symbolerna i L' ovan egentligen hör till språket och att vi markerat en avsedd tolkning. Det finns nämligen modeller, strukturer, till språket där symbolerna har en helt annan betydelse.

Exempel 31: Mängdteorins språk

Låt $L = \{R^2\}$, och låt $M = \{x : x$ är en mängd$\}$. T definieras enligt $T(R) = \in$.
Vi kan nu skriva formler som $R(x_1, x_2)$ och $\exists x_1 \forall x_2 \neg R(x_2, x_1)$ med den avsedda tolkningen $x_1 \in x_2$ respektive $\exists x_1 \forall x_2 (x_2 \notin x_1)$ där x_1 och x_2 antar värden i M, det vill säga x_1 och x_2 är mängder.

Jag kommer som tidigare då och då frångå konventionen att som variabelsymboler enbart använda $x_1, x_2, x_3, \ldots$ etc, och kommer ofta, för att slippa skriva indicerade variabler, att använda x, y och z istället.

Det senaste exemplet visar på ett problem. Vad menas med $T(x_1) = x_1^{\underline{M}}$? Klart är att x_1 är en term, och därför bör peka ut en individ (ett element) i domänen M. Rimligt är att x_1

pekar ut ett godtyckligt element, ett element vilket som helst. Då måste vi ha en teknik vilken gör det möjligt för oss att tala om element i en domän. För att på ett enkelt sätt kunna göra detta, så expanderar vi vårt språk L (eller vår struktur $\underline{M} = (M, T)$) så att varje element i M får ett namn. Vi skapar en namnfullständig expansion av L (av $\underline{M}$). Detta språk betecknas med L_M. Formellt har vi följande definition.

Definition 10: En L-struktur $\underline{M}$ är *namnfullständig*, om det för varje individ $a \in M$ finns en konstantsymbol $c \in L$ så att $c^{\underline{M}} = a$.

Vi kommer ofta att använda beteckningen c_a för den konstantsymbol som är namn på individen a, dvs $c_a^{\underline{M}} = a$.

Definition 11: Låt $\underline{M} = (M, T)$ vara en L-struktur, och låt $\underline{M}' = (M', T')$ vara en L'-struktur. Då är $\underline{M}'$ (L') en *namnfullständig expansion* av $\underline{M}$ (L) om följande gäller:

(i) $L \subseteq L'$, och elementen i $L' - L$ är individkonstanter,
(ii) $M' = M$,
(iii) T' sammanfaller med T på elementen i L och
(iv) $\underline{M}'$ är namnfullständig.

Exempel 32: Betrakta språket $L = \{\underline{0}, \underline{\leq}, \underline{+}, \underline{\cdot}, \underline{S}\}$.

Det är klart att $L_M = \{\underline{0}, \underline{\leq}, \underline{+}, \underline{\cdot}, \underline{S}, \underline{1}, \underline{2}, \underline{3}, ...\}$ är en namnfullständig expansion av $\underline{M}$, där $M = N = \{0, 1, 2, 3, \ldots\}$.

Sanningsbegreppet

Vi ska nu definiera vad det innebär att en sats i det namnfullständiga språket L_M är sann i en modell (struktur) $\underline{M}$. Definitionen är rekursiv och följer definitionen av välbildad formel. Vi använder beteckningen $\underline{M} \models \varphi$, som vi utläser ”satsen φ är sann i strukturen $\underline{M}$”.

Definition 12: En sats φ är *sann* i $\underline{M} = (M, T)$ där $\underline{M}$ är en modell för det namnfullständiga språket L definieras enligt följande:

(1) Atomära satser

Låt $t_1, \ldots, t_n$ vara variabelfria termer (slutna termer) och låt P vara en n-ställig predikatsymbol

(i) $\underline{M} \models t_1 = t_2$ om och endast om $t_1^{\underline{M}} = t_2^{\underline{M}}$,
(ii) $\underline{M} \models P(t_1, \ldots, t_n)$ om och endast om $(t_1^{\underline{M}}, \ldots, t_n^{\underline{M}}) \in P^{\underline{M}}$.
(iii) $\underline{M} \not\models \perp$

Detta innebär att en sats som uttrycker en identitet mellan två termer är sann om och endast om de två termerna pekar ut samma individ i M. En sats som uttrycker en relation mellan ett antal termer är sann om och endast om de individer som termerna betecknar står i relationen som P är namn på till varandra, och $\perp$ är falsk i samtliga strukturer.

(2) Molekylära satser

Låt φ och ψ vara satser.

(i) $\underline{M} \models \neg\varphi$ om och endast om $\underline{M} \not\models \varphi$,
(ii) $\underline{M} \models \varphi \wedge \psi$ om och endast om $\underline{M} \models \varphi$ och $\underline{M} \models \psi$,
(iii) $\underline{M} \models \varphi \vee \psi$ om och endast om $\underline{M} \models \varphi$ eller $\underline{M} \models \psi$,
(iv) $\underline{M} \models \varphi \rightarrow \psi$ om och endast om $\underline{M} \not\models \varphi$ eller $\underline{M} \models \psi$,
(v) $\underline{M} \models \varphi \leftrightarrow \psi$ om och endast om både $\underline{M} \models \varphi$ och $\underline{M} \models \psi$ eller både $\underline{M} \not\models \varphi$ och $\underline{M} \not\models \psi$,

Ovanstående svarar mot vad vi redan gjort i satslogik.

Låt $\varphi(v)$ vara en formel med högst en fri variabel v.

(vi) $\underline{M} \models \forall v\varphi(v)$ om och endast om $\underline{M} \models \varphi(c)$ för varje individkonstant c i L,
(vii) $\underline{M} \models \exists v\varphi(v)$ om och endast om $\underline{M} \models \varphi(c)$ för någon individkonstant c i L

(observera att $\underline{M}$ är namnfullständig, varför alla element i M har namn).

Exempel 33: Låt L_M vara språket i exempel 32, och låt φ vara satsen $\exists x(x \underline{\cdot} x = \underline{4})$.

Det gäller nu att $\underline{M} \models \exists x(x \underline{\cdot} x = \underline{4})$ om och endast om $\underline{M} \models c \underline{\cdot} c = \underline{4}$ för något $c \in L_M$ (eller vilket är ekvivalent för något $c^{\underline{M}} \in M$). Uppenbart gäller det att $2 \cdot 2 = 4$, varför $\underline{M} \models \underline{2} \underline{\cdot} \underline{2} = \underline{4}$. Tänk på att $T(\underline{2}) = 2$ och att $T(\underline{4}) = 4$.

$\therefore\ \underline{M} \models \exists x(x \underline{\cdot} x = \underline{4})$

För att definiera sanning i en godtycklig modell kan vi nu förfara på följande sätt.

Definition 13: Låt L vara ett predikatlogiskt språk och låt $\underline{M} = (M, T)$ vara en L-struktur. Då gäller att satsen φ är *sann* i $\underline{M}$, $\underline{M} \models \varphi$, om och endast om φ är sann i någon namnfullständig expansion $\underline{M}'$ av $\underline{M}$.

Därmed skulle det vara klart att satsen $\exists x(g(x, x) = S(S(S(S(0)))))$ är sann även i modellen i exempel 30. Observera att vi i det språket inte kan skriva $\exists x(x \underline{\cdot} x = \underline{4})$.

Det skulle nu kunna vara så att φ får olika sanningsvärde i olika expansioner. Följande teorem säger emellertid att detta inte inträffar.

Teorem 1: Om $\underline{M}'$ och $\underline{M}''$ är olika namnfullständiga expansioner av en struktur $\underline{M}$ för ett språk L, så gäller för alla satser φ i språket L att $\underline{M}' \models \varphi$ om och endast om $\underline{M}'' \models \varphi$.

Vi behöver även kunna tala om godtyckliga formlers (alltså inte bara satsers) eventuella sanningsvärde. Vanliga formler i matematik är ju t ex på former som $x + y = y + x$. Vi inför därför följande definition. Kom ihåg att beteckningen $\varphi(v_1, \ldots, v_n)$ betyder att de fria variablerna i formeln φ finns bland $v_1, \ldots, v_n$.

Definition 14: Låt $\varphi(v_1, \dots, v_n)$ vara en formel. Vi säger då att formeln φ är *sann* i $\underline{M}$, $\underline{M} \models \varphi$, om och endast om $\underline{M} \models \forall v_1 \forall v_2 \dots \forall v_n \varphi$. Vi använder alltså samma beteckning som i definition 6.

Exempel 34: Låt L vara aritmetikens språk från exempel 32, med den naturliga tolkningen och låt $\underline{M} = (N, T)$.
Då gäller att $\underline{M} \models x \pm y = y \pm x$, eftersom $\underline{M} \models \forall x \forall y (x \pm y = y \pm x)$ och eftersom kommutativa lagen för addition gäller för de naturliga talen.

Definition 15: I strukturen $\underline{M}$ *satisfieras* formeln $\varphi(v)$ av elementet $a \in M$, $\underline{M} \models \varphi[a]$, om och endast om $\underline{M}' \models \varphi(c_a)$ för någon namnfullständig expansion $\underline{M}'$ av $\underline{M}$.
I strukturen $\underline{M}$ *satisfieras* formeln $\varphi(v_1, \dots, v_n)$ av sekvensen av element $a_1, a_2, \dots, a_n \in M$, $\underline{M} \models \varphi[a_1, a_2, \dots, a_n]$, om och endast om $\underline{M}' \models \varphi(c_{a_1}, c_{a_2}, \dots, c_{a_n})$ för någon namnfullständig expansion $\underline{M}'$ av $\underline{M}$.
Formeln $\varphi(v)$ är *satisfierbar* om det finns någon struktur $\underline{M}$ sådan att $\underline{M} \models \varphi[a]$ för något $a \in M$, och motsvarande för $\varphi(v_1, \dots, v_n)$.

Exempel 35: Låt $L = \{P^2\}$, och låt $\underline{M} = (Z, <)$, dvs strukturens domän är mängden av hela tal, och tolkningen av P är den vanliga relationen ”mindre än”. Då gäller t ex att $\underline{M} \models P(x, y)[1, 3]$, eftersom $1 < 3$.

Exempel 36: Låt $L = \{c_1, c_2, R^2, P^1\}$, och låt $M = \{x : x$ är anställd på INV vid HIS$\}$.
Vi ger först en tolkning av de ickelogiska symbolerna, dvs elementen i L.

$c_1^{\underline{M}}$ = Jörgen
$c_2^{\underline{M}}$ = Stefan
$R^{\underline{M}}$ = {(Stefan, Jörgen), (Jörgen, Mikael), (Stefan, Mikael)}
$P^{\underline{M}}$ = {Håkan, Jan-Olav, Jörgen, Kerstin, Mikael, Stefan, Thomas}

Observera att språket endast innehåller två konstanter, varför endast två personer kan ha namn i detta språk. I en namnfullständig expansion $\underline{M}'$ av språket har emellertid alla anställda på institutionen ett namn. Vi har till exempel att $c_{Pia}^{\underline{M}'}$ = Pia, där alltså c_{Pia} är namn på Pia och så vidare.

Betrakta sedan följande satser:

a/ $R(c_2, c_1)$
b/ $R(c_1, c_2)$
c/ $\forall x (P(x) \to R(x, c_1))$
d/ $\exists x (P(x) \wedge R(c_2, x))$

Är dessa satser sanna i strukturen ovan?

a/ $\underline{M} \models R(c_2, c_1) \Leftrightarrow (c_2^{\underline{M}}, c_1^{\underline{M}}) \in R^{\underline{M}} \Leftrightarrow$ (Stefan, Jörgen) $\in R^{\underline{M}}$.
Att det sista påståendet är sant inses via en enkel inspektion av innehållet i mängden $R^{\underline{M}}$. Då måste det gälla att $\underline{M} \models R(c_2, c_1)$.

b/ $\underline{M} \models R(c_1, c_2) \Leftrightarrow (c_1^{\underline{M}}, c_2^{\underline{M}}) \in R^{\underline{M}} \Leftrightarrow$ (Jörgen, Stefan) $\in R^{\underline{M}}$.
Eftersom (Jörgen, Stefan) $\notin R^{\underline{M}}$, så måste $\underline{M} \not\models R(c_1, c_2)$.

c/ $\underline{M} \models \forall x(P(x) \rightarrow R(x, c_1)) \Leftrightarrow \underline{M}' \models P(c_a) \rightarrow R(c_a, c_1)$ för alla $a \in M$ i någon namnfullständig expansion $\underline{M}'$ av $\underline{M}$.
Nu gäller det t ex att Kerstin $\in P^{\underline{M}} = P^{\underline{M}'}$, som ger att $\underline{M}' \models P(c_{Kerstin})$.
Dessutom gäller att (Kerstin, Jörgen) $\notin R^{\underline{M}} = R^{\underline{M}'}$, som ger $\underline{M}' \not\models R(c_{Kerstin}, c_1)$. Därmed gäller det att $\underline{M}' \not\models P(c_{Kerstin}) \rightarrow R(c_{Kerstin}, c_1)$.
Detta ger då att
$\underline{M}' \not\models \forall x(P(x) \rightarrow R(x, c_1))$, varför det enligt teorem 1 måste gälla att $\underline{M} \not\models \forall x(P(x) \rightarrow R(x, c_1))$.

d/ $\underline{M} \models \exists x(P(x) \wedge R(c_2, x)) \Leftrightarrow \underline{M}' \models P(c_a) \wedge R(c_2, c_a)$ för något $a \in M$ i någon namnfullständig expansion $\underline{M}'$ av $\underline{M}$. Observera nu att t ex Mikael $\in P^{\underline{M}'} = P^{\underline{M}}$ och att (Stefan, Mikael) $\in R^{\underline{M}'} = R^{\underline{M}}$.
Då gäller alltså att $\underline{M}' \models P(c_{Mikael}) \wedge R(c_2, c_{Mikael})$, varför
$\underline{M}' \models \exists x(P(x) \wedge R(c_2, x))$, och därmed $\underline{M} \models \exists x(P(x) \wedge R(c_2, x))$.

För att träna på att använda vad vi kan kalla ”konkreta” modeller, som i ovanstående exempel, rekommenderas boken ”Tarski´s world” skriven av Barwise och Etchemendy. Till denna bok hör ett datorprogram, som finns för både PC och Mac, med hjälp av vilket man kan träna modellhantering.

Exempel 37: Låt $L = \{\underline{+}, \underline{\cdot}, \underline{S}, \underline{0}, \underline{1}, \underline{2}, \ldots\}$, och låt $M = \{0, 1, 2, \ldots\}$.
Då är $\underline{M}$ namnfullständig.

Det gäller att $\underline{M} \models \underline{S}(\underline{0}) = \underline{1}$, ty $S(0) = 0 + 1 = 1$, och att

$\underline{M} \models \exists x \exists y \exists z(x \underline{\cdot} x \underline{+} y \underline{\cdot} y = z \underline{\cdot} z)$, eftersom ekvationen $x^2 + y^2 = z^2$ är lösbar, dvs eftersom det finns tal som satisfierar ekvationen. Välj t ex $x = 3$, $y = 4$ och $z = 5$.

Formellt kan detta skrivas

$\underline{M} \models \exists x \exists y \exists z(x \underline{\cdot} x \underline{+} y \underline{\cdot} y = z \underline{\cdot} z) \Leftrightarrow$
$\underline{M} \models \exists y \exists z(c_a \underline{\cdot} c_a \underline{+} y \underline{\cdot} y = z \underline{\cdot} z)$ för något $a \in M \Leftrightarrow$
$\underline{M} \models \exists z(c_a \underline{\cdot} c_a \underline{+} c_b \underline{\cdot} c_b = z \underline{\cdot} z)$ för något $a \in M$ och något $b \in M \Leftrightarrow$
$\underline{M} \models c_a \underline{\cdot} c_a \underline{+} c_b \underline{\cdot} c_b = c_c \underline{\cdot} c_c$ för något $a \in M$, något $b \in M$ och något $c \in M$.

Välj då t ex $a = 3$, $b = 4$ och $c = 5$. Det gäller ju att
$\underline{M} \models \underline{3} \underline{\cdot} \underline{3} + \underline{4} \underline{\cdot} \underline{4} = \underline{5} \underline{\cdot} \underline{5}$.

Predikatlogisk sanning

Definition 16: En sats φ är *(predikatlogiskt) sann* om och endast om $\underline{M} \models \varphi$ för alla strukturer $\underline{M}$. En sats φ är *(predikatlogiskt) falsk* om och endast om $\underline{M} \not\models \varphi$ för alla strukturer $\underline{M}$. En sats är *(predikatlogiskt) kontingent* om den varken är en predikatlogisk sanning eller falskhet. Vi skriver $\models \varphi$ om φ är en predikatlogisk sanning. Om det framgår av sammanhanget att det är just predikatlogisk sanning etc som avses, brukar man stryka ordet ”predikatlogisk”.

I kraft av definition 14 ser vi att definitionen med lämplig justering är tillämpbar även på godtyckliga formler.

Tre viktiga problemtyper

I Visa att $\models \varphi$ för en formel φ.
Teknik: a/ Visa att det för godtyckligt $\underline{M}$ gäller att $\underline{M} \models \varphi$.
b/ Antag $\not\models \varphi$. Då finns en struktur $\underline{M}$ så att $\underline{M} \not\models \varphi$. Härled en motsägelse.

II Visa att $\not\models \varphi$ för en formel φ.
Teknik: Konstruera en struktur $\underline{M}$ sådan att $\underline{M} \not\models \varphi$.

III Visa att φ är satisfierbar.
Teknik: Konstruera en struktur $\underline{M}$ sådan att $\underline{M} \models \varphi$.

Observera att man inte, genom att exemplifiera med en modell i vilken en sats är sann (falsk), kan bevisa att en sats är sann (falsk) i alla modeller.

Exempel 38a: Visa att $\models \forall x(x = x)$.

Lösning: Låt $\underline{M}$ vara en godtycklig modell. Eftersom satsen vi skall undersöka saknar ickelogiska symboler, behöver vi inte bry oss om funktionen T.
Låt sedan a vara ett godtyckligt element i strukturens domän M. Eftersom $a = a$, så måste $\underline{M}' \models c_a = c_a$ för någon namnfullständig expansion $\underline{M}'$ av $\underline{M}$. Eftersom a godtycklig, så gäller $\underline{M}' \models \forall x(x = x)$, och därmed att $\underline{M} \models \forall x(x = x)$, vilket i sin tur ger att $\models \forall x(x = x)$, eftersom $\underline{M}$ godtycklig.

Vi kan alltid välja att arbeta med en namnfullständig struktur direkt, och inte som i exempel 38a ovan först utgå från en struktur och därefter bilda en namnfullständig struktur, för att slutligen gå tillbaka till den ursprungliga strukturen. Ett alternativt sätt att presentera resonemanget i exempel 38a är att direkt förutsätta att strukturen är namnfullständig och sedan använda begreppet satisfierbarhet, som vi har definierat med hjälp av namnfullständiga strukturer.

Exempel 38b: Visa att $\models \forall x(x = x)$.

Lösning: Låt $\underline{M}$ vara en godtycklig modell, som vi kan anta är namnfullständig. Låt a vara ett godtyckligt element i strukturens domän M. Eftersom $a = a$, så måste a satisfiera formeln $x = x$ i $\underline{M}$, dvs $\underline{M} \models x = x\,[a]$, och eftersom a godtyckligt, gäller det att $\underline{M} \models \forall x(x = x)$, vilket i sin tur ger att $\models \forall x(x = x)$, eftersom $\underline{M}$ godtycklig.

Exempel 39: Visa att $\forall x P(x)$ är kontingent.

Lösning: Vi har att konstruera två modeller. I den ena skall satsen vara falsk, i den andra skall satsen vara sann (problemtyp II och III ovan). Vi förutsätter ett språk som innehåller predikatsymbolen P, och låter först $\underline{M}_1$ vara en namnfullständig struktur med $M_1 = \{a, b\}$, och $P^{\underline{M}_1} = \{a\}$. I denna struktur gäller då att $\underline{M}_1 \not\models \forall x P(x)$, ty $\underline{M}_1 \not\models P(x)[b]$. Låt sedan $\underline{M}_2$ vara en namnfullständig struktur med $M_2 = \{a\}$, och $P^{\underline{M}_2} = \{a\}$. I denna struktur gäller att $\underline{M}_2 \models \forall x P(x)$, ty varje element i M_2 satisfierar $P(x)$. Vi har alltså konstruerat en modell i vilken satsen är falsk och en i vilken den är sann, och kan då dra slutsatsen att $\forall x P(x)$ är predikatlogiskt kontingent.

Exempel 40: Visa att $\models \forall x(P(x) \rightarrow Q(x)) \rightarrow (\exists xP(x) \rightarrow \exists xQ(x))$.

Lösning: Antag för motsägelse att $\not\models \forall x(P(x) \rightarrow Q(x)) \rightarrow (\exists xP(x) \rightarrow \exists xQ(x))$.

Då finns en struktur $\underline{M}$ sådan att

$$\underline{M} \not\models \forall x(P(x) \rightarrow Q(x)) \rightarrow (\exists xP(x) \rightarrow \exists xQ(x)).$$

Detta ger då tillsammans med sanningsvillkoret för den materiella implikationen att

(1) $\underline{M} \models \forall x(P(x) \rightarrow Q(x))$ och

(2) $\underline{M} \not\models \exists xP(x) \rightarrow \exists xQ(x)$.

(2) ger nu att via sanningsvillkoret för den materiella implikationen att

(3) $\underline{M} \models \exists xP(x)$ och

(4) $\underline{M} \not\models \exists xQ(x)$.

(3) och sanningsvillkoret för den existenskvantifikatorn ger att

(5) $\underline{M} \models P(c_a)$ för något element $a \in M$.

För detta a gäller enligt (1) och sanningsvillkoret för den allkvantifikatorn att

(6) $\underline{M} \models P(c_a) \rightarrow Q(c_a)$.

Implikationens sanningsvillkor tillsammans med (5) och (6) ger då att

(7) $\underline{M} \models Q(c_a)$,

vilket ger att

(8) $\underline{M} \models \exists xQ(x)$.

Detta motsäger emellertid (4).

Då måste $\models \forall x(P(x) \rightarrow Q(x)) \rightarrow (\exists xP(x) \rightarrow \exists xQ(x))$.

Försök nu använda begreppet satisfierbarhet för att presentera en lösning på problemet i exemplet ovan. Observera att du så småningom hamnar i en situation där du vet att

$\underline{M} \models P(x)[a]$ för något element $a \in M$, och att

$\underline{M} \models P(x) \rightarrow Q(x)[a]$ för samma a.

Kan man då dra slutsatsen att $\underline{M} \models Q(x)[a]$? Varför?

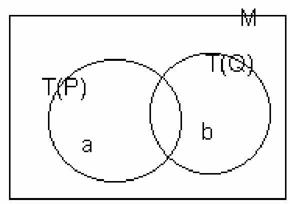
Exempel 41: Visa att $\not\models \forall x(P(x) \vee Q(x)) \rightarrow (\forall xP(x) \vee \forall xQ(x))$.

Lösning: Vi har att konstruera en modell $\underline{M}$ sådan att

$\underline{M} \not\models \forall x(P(x) \vee Q(x)) \rightarrow (\forall xP(x) \vee \forall xQ(x))$.

I denna struktur måste $\forall x(P(x) \vee Q(x))$ vara sann, och $\forall xP(x) \vee \forall xQ(x)$ vara falsk, dvs både $\forall xP(x)$ och $\vee \forall xQ(x)$ måste vara falska.

Betrakta följande struktur: $M = \{a, b\}$, $P^{\underline{M}} = \{a\}$ och $Q^{\underline{M}} = \{b\}$. Vi förutsätter att $\underline{M}$ är namnfullständig. Strukturen kan illustreras med följande diagram:



I denna struktur gäller nu att $\underline{M} \models P(x) \vee Q(x)[a]$ och att $\underline{M} \models P(x) \vee Q(x)[b]$, dvs varje element i M satisfierar $P(x) \vee Q(x)$. Då måste $\underline{M} \models \forall x(P(x) \vee Q(x))$. Dessutom gäller att $\underline{M} \not\models P(x)[b]$, varför $\underline{M} \not\models \forall xP(x)$, och att $\underline{M} \not\models Q(x)[a]$, varför $\underline{M} \not\models \forall xQ(x)$. Detta ger $\underline{M} \not\models \forall xP(x) \vee \forall xQ(x)$.

Därmed är det klart att
$\underline{M} \not\models \forall x(P(x) \vee Q(x)) \rightarrow (\forall xP(x) \vee \forall xQ(x))$, och då måste
$\not\models \forall x(P(x) \vee Q(x)) \rightarrow (\forall xP(x) \vee \forall xQ(x))$.

Exempel 42: Visa att $\models \exists x\forall y\neg R(x, y) \rightarrow \neg\forall x\exists yR(x, y)$.

Lösning: Beteckna satsen med φ. Anta för motsägelse att $\not\models \varphi$.

Då finns en struktur $\underline{M}$ sådan att $\underline{M} \not\models \varphi$, där vi utan inskränkningar kan anta att $\underline{M}$ är namnfullständig.

I denna struktur gäller då att (sanningsvillkor för materiell implikation)
(1) $\underline{M} \models \exists x\forall y\neg R(x, y)$, och att
(2) $\underline{M} \not\models \neg\forall x\exists yR(x, y)$.

(2) ger via negationens sanningsvillkor att
(3) $\underline{M} \models \forall x\exists yR(x, y)$.

Nu ger (1) och sanningsvillkoret för existenskvantifikatorn att
(4) $\underline{M} \models \forall y\neg R(c_a, y)$, för något element $a \in M$. Observera att vi har bestämt att $\underline{M}$ är namnfullständig, och att vi måste eliminera existenskvantifikatorn från (1) innan vi kan eliminera allkvantifikatorn från (3) (varför?).

För precis samma individ a vi valde i (4) ovan gäller nu via allkvantifikatorns sanningsvillkor att
(5) $\underline{M} \models \exists yR(c_a, y)$, eftersom $\exists yR(x, y)$ satisfieras av alla element $a \in M$.

Av (5) får vi på grund av existenskvantifikatorns sanningsvillkor att
(6) $\underline{M} \models R(c_a, c_b)$, för något $b \in M$.

För detta b ger nu (4) att (allkvantifikatorns sanningsvillkor)
(7) $\underline{M} \models \neg R(c_a, c_b)$, som via negationens sanningsvillkor ger att
(8) $\underline{M} \not\models R(c_a, c_b)$.

Det gäller nu att (8) och (6) motsäger varandra eftersom en och samma sats inte kan vara både sann och falsk i en och samma struktur.

Det kan då inte vara så att $\not\models \varphi$, varför $\models \exists x\forall y\neg R(x, y) \rightarrow \neg\forall x\exists yR(x, y)$.

Predikatlogisk konsekvens

Definition 17: Låt $\varphi_1, \varphi_2, \dots, \varphi_n$ och ψ vara satser. Vi säger att ψ är en *predikatlogisk följd (konsekvens)* av $\varphi_1, \varphi_2, \dots, \varphi_n$ om och endast om det för varje modell $\underline{M}$ sådan att $\underline{M} \models \varphi_i$, $1 \le i \le n$, gäller att $\underline{M} \models \psi$.
Vi skriver $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$, om ψ är en logisk följd av $\varphi_1, \varphi_2, \dots, \varphi_n$, och $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \not\models \psi$, om ψ inte är en logisk följd av $\varphi_1, \varphi_2, \dots, \varphi_n$.

Observera att en slutledning med premisserna $\varphi_1, \varphi_2, \dots, \varphi_n$ och slutsats ψ är *predikatlogiskt giltig* om och endast om $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$.

Två viktiga problemtyper

I Visa att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$.
Teknik: a/ Visa att det för godtycklig struktur $\underline{M}$ sådan att $\underline{M} \models \varphi_i$ för $1 \le i \le n$, gäller att $\underline{M} \models \psi$.
b/ Antag att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \not\models \psi$. Då finns en struktur $\underline{M}$ så att $\underline{M} \models \varphi_i$ för $1 \le i \le n$, och att $\underline{M} \not\models \psi$. Härled en motsägelse.

II Visa att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \not\models \psi$.
Teknik: Konstruera en struktur $\underline{M}$ så att $\underline{M} \models \varphi_i$, för $1 \le i \le n$, och $\underline{M} \not\models \psi$.

Exempel 43: Visa att $\{\forall x(P(x) \to Q(x)), \exists xP(x)\} \models \exists xQ(x)$.

Lösning: Låt $\underline{M}$ vara en godtycklig modell sådan att
(1) $\underline{M} \models \forall x(P(x) \to Q(x))$, och
(2) $\underline{M} \models \exists xP(x)$.

Vi har att visa att $\underline{M} \models \exists xQ(x)$, och vi kan utan inskränkningar anta att $\underline{M}$ är namnfullständig.

(2) ger via existenskvantifikatorns sanningsvillkor att
(3) $\underline{M} \models P(c_a)$ för något $a \in M$.

För denna individ gäller nu på grund av (1) och allkvantifikatorns sanningsvillkor att
(4) $\underline{M} \models P(c_a) \to Q(c_a)$.

Då ger (3), (4) och implikationens sanningsvillkor att
(5) $\underline{M} \models Q(c_a)$,
som tillsammans med existenskvantifikatorns sanningsvillkor ger att
(6) $\underline{M} \models \exists xQ(x)$,
vilket var precis vad vi eftersträvade.

Eftersom $\underline{M}$ godtycklig, gäller då $\{\forall x(P(x) \to Q(x)), \exists xP(x)\} \models \exists xQ(x)$.

Exempel 44: Visa att $\exists x(P(x) \rightarrow Q(c)) \;\; |\neq \;\; \exists xP(x) \rightarrow Q(c)$. Om, som fallet är här, premissmängden endast innehåller ett element, utelämnar vi ofta mängdklamrarna.

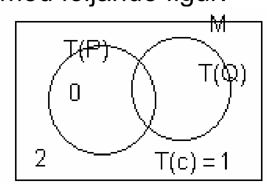
Lösning: Vi har att konstruera en modell $\underline{M}$ sådan att $\underline{M} \models \exists x(P(x) \rightarrow Q(c))$, och $\underline{M} \; |\neq \exists xP(x) \rightarrow Q(c)$.

För att det sista villkoret skall vara uppfyllt krävs att $\underline{M} \models \exists xP(x)$, och att $\underline{M} \; |\neq Q(c)$.

Betrakta följande struktur:

$$M = \{0, 1, 2\},\; P^{\underline{M}} = \{0\},\; Q^{\underline{M}} = \emptyset \text{ och } c^{\underline{M}} = 1.$$

Vi antar för enkelhets skull att modellen är namnfullständig, och vi kan illustrera med följande figur.



I denna struktur gäller då att $\underline{M} \models \exists xP(x)$, eftersom $\underline{M} \models P(x)[0]$, och att $\underline{M} \; |\neq Q(c)$, eftersom $T(c) = c^{\underline{M}} = 1 \notin T(Q) = Q^{\underline{M}}$.

Vi har då att $\underline{M} \; |\neq \exists xP(x) \rightarrow Q(c)$.

Det gäller dessutom att $\underline{M} \models \exists x(P(x) \rightarrow Q(c))$, eftersom $\underline{M} \models P(\underline{2}) \rightarrow Q(c)$, då ju $\underline{M} \; |\neq P(\underline{2})$, eftersom $T(\underline{2}) = 2 \notin P^{\underline{M}}$, och $\underline{M} \; |\neq Q(c)$, eftersom $c^{\underline{M}} = 1 \notin Q^{\underline{M}}$. Observera att siffran $\underline{2}$ är namn på talet 2.

Därmed har vi visat att modellen ovan har de önskade egenskaperna, och därmed är det klart att $\exists x(P(x) \rightarrow Q(c) \;\; |\neq \;\; \exists xP(x) \rightarrow Q(c)$.

Exempel 45: Visa att $\forall xP(x) \rightarrow Q(c) \;\; \models \;\; \exists x(P(x) \rightarrow Q(c))$.

Lösning: Antag för motsägelse att $\forall xP(x) \rightarrow Q(c) \;\; |\neq \;\; \exists x(P(x) \rightarrow Q(c))$.

Då finns en struktur $\underline{M}$, som vi förutsätter är namnfullständig, sådan att

(1) $\underline{M} \models \forall xP(x) \rightarrow Q(c)$, och

(2) $\underline{M} \; |\neq \exists x(P(x) \rightarrow Q(c))$.

(2) ger via existenskvantifikatorns sanningsvillkor att

(3) $\underline{M} \; |\neq P(c_a) \rightarrow Q(c)$, för alla $a \in M$, dvs

(4) $\underline{M} \models P(c_a)$ för alla $a \in M$, och

(5) $\underline{M} \; |\neq Q(c)$.

(4) ger via allkvantifikatorns sanningsvillkor att

(6) $\underline{M} \models \forall xP(x)$, som tillsammans med (5) ger

(7) $\underline{M} \; |\neq \forall xP(x) \rightarrow Q(c)$, som motsäger (1).

Det är då klart att $\forall xP(x) \rightarrow Q(c) \;\; \models \;\; \exists x(P(x) \rightarrow Q(c))$.

Teorem 2: Låt $\varphi, \varphi_1, \varphi_2, \dots, \varphi_n, \psi$ och θ vara satser i ett predikatlogiskt språk.

(i) $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta \iff \{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \to \theta$,

(ii) $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \iff \models \varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \to \psi$,

(iii) $\varphi \models \psi \iff \models \varphi \to \psi$.

Bevis: (i) $\Rightarrow$ Antag att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$.
Vi har att visa att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \to \theta$.

Antag för motsägelse att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \not\models \psi \to \theta$.

Då finns en struktur $\underline{M}$ sådan att $\underline{M} \models \varphi_i$ för $1 \le i \le n$, och $\underline{M} \not\models \psi \to \theta$, dvs $\underline{M} \models \psi$ och $\underline{M} \not\models \theta$.

I denna struktur är då samtliga satser i $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\}$ sanna medan θ är falsk. Detta motsäger att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$, och därmed är det klart att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \to \theta$.

(i) $\Leftarrow$ Antag att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \to \theta$.
Vi har att visa att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$.

Antag för motsägelse att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \not\models \theta$.

Då finns en struktur $\underline{M}$ sådan att $\underline{M} \models \varphi_i$ för $1 \le i \le n$, $\underline{M} \models \psi$ och $\underline{M} \not\models \theta$.

I denna struktur gäller då att $\underline{M} \models \varphi_i$ för $1 \le i \le n$, och $\underline{M} \not\models \psi \to \theta$, men detta motsäger $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi \to \theta$, varför det måste gälla att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \psi\} \models \theta$.

(ii) Övning, argumentet är snarlikt det i (i) ovan.

(iii) Specialfall av (ii).

Begreppet logisk konsekvens kan definieras för godtyckliga formler och inte bara för satser.

Definition 18: Låt $\varphi_1, \varphi_2, \dots, \varphi_n$ och ψ vara formler i ett predikatlogiskt språk, där vi antar att de fria variablerna i formlerna är bland $v_1, v_2, \dots, v_k$. Då är ψ en *(predikat)logisk konsekvens (följd)* av $\varphi_1, \varphi_2, \dots, \varphi_n$,
med symboler $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$ om och endast om
$\models \forall v_1 \forall v_2 \dots \forall v_k(\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \to \psi)$.

Exempel 46: Visa att $P(x) \not\models P(y)$.

Lösning: Vi har att visa att $\not\models \forall x \forall y(P(x) \to P(y))$.

Låt $M = \{a, b\}$, och låt $P^{\underline{M}} = \{a\}$, och vi antar som vanligt att strukturen är namnfullständig.

I denna struktur gäller då att $\underline{M} \not\models P(c_a) \to P(c_b)$, eftersom $\underline{M} \models P(c_a)$ och $\underline{M} \not\models P(c_b)$, då $a \in P^{\underline{M}}$ och $b \notin P^{\underline{M}}$.

Allkvantifikatorns sanningsvillkor ger då att $\underline{M} \not\models \forall x \forall y(P(x) \to P(y))$, varför $\not\models \forall x \forall y(P(x) \to P(y))$.

Därmed är det klart att $P(x) \not\models P(y)$.

Predikatlogisk ekvivalens

Definition 19: Låt φ och ψ vara två satser. φ och ψ är *(predikat)logiskt ekvivalenta* om och endast om både $\varphi \models \psi$ och $\psi \models \varphi$. Vi skriver då $\varphi =\!\models \psi$ eller $\varphi \Leftrightarrow \psi$.

Samma begrepp för godtyckliga formler definieras enligt följande.

Definition 20: Låt φ och ψ vara två formler, där de fria variablerna i formlerna finns bland $v_1, v_2, \dots, v_k$. Då är φ och ψ *(predikat)logiskt ekvivalenta* om och endast om $\models \forall v_1 \forall v_2 \dots \forall v_k(\varphi \leftrightarrow \psi)$. Även här skriver vi $\varphi =\!\models \psi$ eller $\varphi \Leftrightarrow \psi$.

Exempel 47: Visa att $P(x) \not\Leftrightarrow P(y)$.

Vi har egentligen att visa att $\not\models \forall x \forall y(P(x) \leftrightarrow P(y))$, och detta kan göras med samma struktur som i exempel 46.

Detta resultat är viktigt eftersom det visar att vi inte kan byta fria variabler. Däremot gäller det att $\forall v_1\varphi(v_1) \Leftrightarrow \forall v_2\varphi(v_2)$, och $\exists v_1\varphi(v_1) \Leftrightarrow \exists v_2\varphi(v_2)$, om v_1 inte förekommer i $\varphi(v_2)$, och v_2 inte i $\varphi(v_1)$. Detta visar att vi får byta bundna variabler, om vi väljer en ny variabel, dvs en variabel som inte tidigare förekommer i formeln.

Teorem 3: Låt φ och ψ vara två satser. Då $\varphi \Leftrightarrow \psi$ om och endast om $\models \varphi \leftrightarrow \psi$.

Bevis: Detta följer direkt av teorem 2 på följande sätt:
$\varphi \Leftrightarrow \psi$ om och endast om (definition 19)
$\varphi \models \psi$ och $\psi \models \varphi$ om och endast om (teorem 2)
$\models \varphi \to \psi$ och $\models \psi \to \varphi$ om och endast om
$\models (\varphi \to \psi) \land (\psi \to \varphi)$ om och endast om
$\models \varphi \leftrightarrow \psi$.

Följande substitutionsprincip är viktig och används ofta, många gånger utan att detta klart utsägs.

Teorem 4: (Substitutionsprincipen)
Låt φ, ψ och θ vara tre formler, som uppfyller följande villkor.

(i) $\varphi \Leftrightarrow \psi$,
(ii) φ förekommer som delformel i θ, och
(iii) θ^* är resultatet av att ersätta en förekomst av φ med ψ i θ.

Då är $\theta \Leftrightarrow \theta^*$.

Detta resultat kan bevisas med induktion över formeln θ:s längd.

Exempel 48: Låt $\theta = \forall x(R(x, y) \to \exists yP(y))$. I denna formel är $\varphi = \exists yP(y)$ en delformel. Denna delformel är logiskt ekvivalent med $\psi = \exists zP(z)$ enligt kommentar efter exempel 21. Ersätter vi φ med ψ i θ, så erhåller vi
$\theta^* = \forall x(R(x, y) \to \exists zP(z))$.

Teorem 4 säger då att
$\forall x(R(x, y) \to \exists yP(y)) \Leftrightarrow \forall x(R(x, y) \to \exists zP(z))$.

Problem av typen visa att $\varphi \Leftrightarrow \psi$ respektive $\varphi <\neq> \psi$ återförs på problemen på sidan 74.

Exempel 49: Visa att $\forall xP(x) \to Q(c) \Leftrightarrow \exists x(P(x) \to Q(c))$.

Lösning: Vi har att visa att

(1) $\forall xP(x) \to Q(c) \models \exists x(P(x) \to Q(c))$, och att
(2) $\exists x(P(x) \to Q(c)) \models \forall xP(x) \to Q(c)$.

(1) är redan klart i exempel 45, så det återstår att visa (2).

Antag därför för motsägelse att $\exists x(P(x) \to Q(c)) \not\models \forall xP(x) \to Q(c)$.

Då finns en struktur $\underline{M}$, som vi som vanligt kan anta är namnfullständig, sådan att

(3) $\underline{M} \models \exists x(P(x) \to Q(c))$, och
(4) $\underline{M} \not\models \forall xP(x) \to Q(c)$, dvs
(5) $\underline{M} \models \forall xP(x)$, och
(6) $\underline{M} \not\models Q(c)$.

Påståendet (3) ger via existenskvantifikatorns sanningsvillkor att
(7) $\underline{M} \models P(c_a) \to Q(c)$ för något $a \in M$.

Eftersom det enligt (5) gäller att varje element i strukturens domän M satisfierar formeln P(x), måste speciellt a göra detta, dvs vi har att
(8) $\underline{M} \models P(c_a)$.

Då ger (7), (8) och implikationens sanningsvillkor att
(9) $\underline{M} \models Q(c)$, som motsäger (6).

Därmed är det klart att $\exists x(P(x) \to Q(c)) \models \forall xP(x) \to Q(c)$.

Detta ger då tillsammans med (1) via exempel 35 att
$\forall xP(x) \to Q(c) \Leftrightarrow \exists x(P(x) \to Q(c))$.

Exempel 50: Visa att $\exists x(P(x) \wedge Q(x)) <\neq> \exists xP(x) \wedge \exists xQ(x)$.

Lösning: Man kan visa att

(1) $\exists x(P(x) \wedge Q(x)) \models \exists xP(x) \wedge \exists xQ(x)$,

men detta är ointressant i detta sammanhang, eftersom det gäller att

(2) $\exists xP(x) \wedge \exists xQ(x) \not\models \exists x(P(x) \wedge Q(x))$.

Att det bör vara så här kan man inse om man betänker att (1) utsäger att om någon individ har båda egenskaperna som P respektive Q är namn på, så har någon individ den egenskap som P är namn på och någon har den egenskap som Q är namn på. Rimligheten av påståendet (2) kan inses av att om nu någon individ har den egenskap som P är namn på, och någon, inte nödvändigtvis samma, individ har den egenskap som Q är namn på, så har vi ingen garanti för att någon individ har båda egenskaperna.

För att visa (2) har vi att konstruera en struktur $\underline{M}$ sådan att
$\underline{M} \models \exists xP(x) \wedge \exists xQ(x)$, och $\underline{M} \not\models \exists x(P(x) \wedge Q(x))$.

Låt $M = \{a, b\}$, och vi antar att $\underline{M}$ är namnfullständig. Låt dessutom $P^{\underline{M}} = \{a\}$ och $Q^{\underline{M}} = \{b\}$.

I denna struktur gäller då att $\underline{M} \models \exists xP(x)$, eftersom $\underline{M} \models P(c_a)$, och $\underline{M} \models \exists xQ(x)$, eftersom $\underline{M} \models Q(c_b)$, och därmed har vi att $\underline{M} \models \exists xP(x) \wedge \exists xQ(x)$.

Det gäller också att varken a eller b satisfierar formeln $P(x) \wedge Q(x)$, ty $\underline{M} \not\models P(c_a) \wedge Q(c_a)$ och $\underline{M} \not\models P(c_b) \wedge Q(c_b)$, dvs $\underline{M} \not\models \exists x(P(x) \wedge Q(x))$.

Därmed är det klart att $\exists xP(x) \wedge \exists xQ(x) \not\models \exists x(P(x) \wedge Q(x))$, och därmed är det klart att $\exists x(P(x) \wedge Q(x)) \not\Leftrightarrow \exists xP(x) \wedge \exists xQ(x)$.

Satisfierbarhet

Definition 21: En mängd Γ av satser är *satisfierbar* om och endast om det finns en struktur $\underline{M}$ sådan att $\underline{M} \models \varphi$ för varje $\varphi \in \Gamma$.

Man visar att en mängd av satser är satisfierbar genom att konstruera en struktur sådan att samtliga satser är sanna i strukturen, och man visar att en mängd av satser inte är satisfierbar genom att visa att det inte finns någon modell i vilken samtliga satser är sanna.

Exempel 51: Visa att $\{\exists x\forall yR(x, y), \forall x\forall y\exists z(R(x, z) \wedge R(z, y))\}$ är satisfierbar.

Lösning: Beteckna den första satsen i mängden med φ_1 och den andra med φ_2.

Vi har att konstruera en modell $\underline{M}$ sådan att $\underline{M} \models \varphi_1$ och $\underline{M} \models \varphi_2$.

Betrakta följande struktur $M = \{a\}$, $R^{\underline{M}} = \{(a, a)\}$. I denna struktur gäller att $\underline{M} \models \varphi_1$, eftersom $\underline{M} \models R(x, y)[a, a]$, och a är det enda elementet i M.

Av motsvarande skäl gäller också att $\underline{M} \models \varphi_2$.

Alltså är $\{\exists x\forall yR(x, y), \forall x\forall y\exists z(R(x, z) \wedge R(z, y))\}$ är satisfierbar.

Exempel 52: Visa att $\{P(c), \neg\exists xP(x)\}$ inte är satisfierbar.

Lösning: Vi har att visa att det inte finns någon modell $\underline{M}$ sådan att båda satserna är sanna i $\underline{M}$. Antag för motsägelse att det finns en struktur $\underline{M}$ sådan att
(1) $\underline{M} \models P(c)$, och
(2) $\underline{M} \models \neg\exists xP(x)$.

Negationens sanningsvillkor ger tillsammans med (2) att
(3) $\underline{M} \not\models \exists xP(x)$.

Av (1) tillsammans med existenskvantifikatorns sanningsvillkor följer emellertid att $\underline{M} \models \exists xP(x)$, vilket då motsäger (3).

Det finns då ingen struktur sådan att båda satserna är sanna, varför satsmängden inte är satisfierbar.

Teorem 5: Låt $\varphi_1, \varphi_2, \dots, \varphi_n$ och ψ vara satser i ett predikatlogiskt språk.
$\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$ om och endast om $\{\varphi_1, \varphi_2, \dots, \varphi_n, \neg\psi\}$ inte är satisfierbar.

Bevis: $\Rightarrow$ Antag att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \models \psi$.
Antag för motsägelse att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \neg\psi\}$ är satisfierbar.

Då finns en struktur $\underline{M}$ sådan att $\underline{M} \models \varphi_i$ för $1 \leq i \leq n$, och $\underline{M} \models \neg\psi$, dvs $\underline{M} \not\models \psi$, men detta motsäger förutsättningen.

$\Leftarrow$ Antag att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \neg\psi\}$ inte är satisfierbar.
Antag för motsägelse att $\{\varphi_1, \varphi_2, \dots, \varphi_n\} \not\models \psi$.

Detta innebär att det finns en struktur $\underline{M}$ sådan att $\underline{M} \models \varphi_i$ för $1 \leq i \leq n$, och $\underline{M} \not\models \psi$, dvs $\underline{M} \models \neg\psi$. Vi har då att $\{\varphi_1, \varphi_2, \dots, \varphi_n, \neg\psi\}$ är satisfierbar, vilket motsäger förutsättningen, och därmed är saken klar.

Prenex normalform (PNF)

Vi har i satslogik mött två standardiserade sätt att skriva satslogiska formler, nämligen disjunktiv respektive konjunktiv normalform. Vi skall här möta ett standardiserat sätt att skriva predikatlogiska formler. Formen kallas prenex normalform, och idén är att skriva om en formel φ till en formel ψ, så att $\varphi \Leftrightarrow \psi$ och samtliga kvantifikatorer som förekommer i ψ finns "först" i formeln.

Definition 22: En formel ψ är på *prenex normalform* (PNF), om ψ är på formen $Q_1v_1Q_2v_2\dots Q_nv_n\theta$, där θ är kvantifikatorfri. Med Q_i avses någon av symbolerna $\exists$ eller $\forall$, dvs en kvantifikatorsymbol, och v_i är en variabel.

Exempel 53:

$\forall x\exists y(P(x, y) \to \neg R(x))$	är på PNF.	Tänk på att kvantifikatorernas räckvidd är hela formeln.
$\forall x\exists yP(x, y) \to \neg R(x)$	är inte på PNF.	Kvantifikatorerna ingår i försatsen och har en räckvidd som begränsar sig till just försatsen.
$\forall x(\exists yP(x, y) \to \neg R(x))$	är inte på PNF,	eftersom existenskvantifikatorns räckvidd enbart är försatsen.

Teorem 6: För varje välbildad formel φ finns en välbildad formel ψ så att
(i) $\varphi \Leftrightarrow \psi$ och
(ii) ψ är på PNF.

Vi avstår från att bevisa detta teorem, men formulerar informellt en algoritm som ger önskad formel ψ som resultat. Formeln ψ i teorem 6 är inte entydigt bestämd.

I den sekvens av omskrivningar vi har att genomföra för att skriva en formel på PNF, används en uppsättning logiska ekvivalenser. Många av dem är redan kända av läsaren, men vi formulerar samtliga för fullständighetens skull.

(1)	$\varphi \leftrightarrow \psi \Leftrightarrow (\varphi \to \psi) \land (\psi \to \varphi)$	
(2)	$\forall v_1\varphi(v_1) \Leftrightarrow \forall v_2\varphi(v_2)$	v_1 förekommer inte i $\varphi(v_2)$, och v_2 inte i $\varphi(v_1)$
(3)	$\exists v_1\varphi(v_1) \Leftrightarrow \exists v_2\varphi(v_2)$	v_1 förekommer inte i $\varphi(v_2)$, och v_2 inte i $\varphi(v_1)$

(4) $\neg\exists v\varphi \Leftrightarrow \forall v\neg\varphi$ v är en variabel
(5) $\neg\forall v\varphi \Leftrightarrow \exists v\,\neg\varphi$ där v är en variabel

I formlerna (6) till (13) får variabeln v inte vara fri i φ.

(6) $\varphi \wedge \forall v\psi \Leftrightarrow \forall v(\varphi \wedge \psi)$
(7) $\varphi \wedge \exists v\psi \Leftrightarrow \exists v(\varphi \wedge \psi)$
(8) $\varphi \vee \forall v\psi \Leftrightarrow \forall v(\varphi \vee \psi)$
(9) $\varphi \vee \exists v\psi \Leftrightarrow \exists v(\varphi \vee \psi)$

Eftersom operationerna $\wedge$ och $\vee$ är kommutativa, kan vi också kasta om ordningen mellan konjunkterna och disjunkterna ovan.

(10) $\varphi \rightarrow \forall v\psi \Leftrightarrow \forall v(\varphi \rightarrow \psi)$
(11) $\varphi \rightarrow \exists v\psi \Leftrightarrow \exists v(\varphi \rightarrow \psi)$
(12) $\forall v\psi \rightarrow \varphi \Leftrightarrow \exists v(\psi \rightarrow \varphi)$
(13) $\exists v\psi \rightarrow \varphi \Leftrightarrow \forall v(\psi \rightarrow \varphi)$

Operationen $\rightarrow$ är inte kommutativ, så därför behöver vi dubbla uppsättningen formler för den materiella implikationen.

Formlerna (6) till (13) brukar kallas prenexformler, och de ser vid ett första ögonkast ganska omständiga ut, men med två undantag är det bara att ”flytta ut” kvantifikatorn framför formeln. Undantagen är formlerna (12) och (13), där kvantifikatorn finns i försatsen i en implikation. Vi byter då kvantifikator när vi flyttar ut den så att den får hela formeln som räckvidd. Vi visar med ett exempel hur omskrivningen går till innan vi formulerar själva algoritmen.

Exempel 54: Skriv formeln $\forall x\exists yP(x, y) \rightarrow \neg Q(x)$ på PNF.

Lösning: I implikationen ovan har båda kvantifikatorerna försatsen som räckvidd. Vi vill skriva om den så att kvantifikatorerna har hela formeln som räckvidd. Nu gäller det att samtliga variabelförekomster i försatsen är bundna, medan förekomsten av x i eftersatsen är fri. Skulle vi nu använda regel (12) ovan, skulle vi få en formel med en helt annan innebörd, eftersom den fria förekomsten av x skulle bli bunden. Vi använder därför regel (2) och byter *bundna* förekomster av x. Därefter kan vi flytta ut allkvantifikatorn med regel (12) så att den erhållna existenskvantifikatorn får hela formeln som räckvidd. Variabeln y förekommer inte i eftersatsen, så den lyfter vi ut direkt med hjälp av regel (13). Vi får då följande svit av logiska ekvivalenser

$\forall x\exists yP(x, y) \rightarrow \neg Q(x)$	$\Leftrightarrow$	[regel (2)]
$\forall z\exists yP(z, y) \rightarrow \neg Q(x)$	$\Leftrightarrow$	[regel (12)]
$\exists z(\exists yP(z, y) \rightarrow \neg Q(x))$	$\Leftrightarrow$	[regel (13)]
$\exists z\forall y(P(z, y) \rightarrow \neg Q(x))$		

Nu gäller det att

(i) $\forall x\exists yP(x, y) \rightarrow \neg Q(x) \Leftrightarrow \exists z\forall y(P(z, y) \rightarrow \neg Q(x))$, och
(ii) $\exists z\forall y(P(z, y) \rightarrow \neg Q(x))$ är på PNF.

Det är synnerligen viktigt att de variabelbyten man har att utföra sker på ett korrekt sätt, dvs att man vid behov byter *bundna* variabler. Kom ihåg att $\varphi(v_1)$ inte är logiskt ekvivalent med

$\varphi(v_2)$, medan $\forall v_1\varphi(v_1) \Leftrightarrow \forall v_2\varphi(v_2)$ och $\exists v_1\varphi(v_1) \Leftrightarrow \exists v_2\varphi(v_2)$, där de två logiska ekvivalenserna, för att gälla, kräver att vissa villkor är uppfyllda på variablerna v_1 och v_2.

Arbetsordningen, algoritmen, för omskrivningen ser ut på följande sätt.

1. Eliminera förekomster av $\leftrightarrow$ med regel (1)
2. Byt vid behov bundna variabler med reglerna (2) och (3), så att ingen variabel som förekommer fri även förekommer i en kvantifikator. Varje kvantifikator skall dessutom binda olika variabler.
3. Flytta därefter successivt ut kvantifikatorerna en efter en med reglerna (4) till (13). Eventuellt får man här lägga till parenteser för att formlerna skall vara korrekt uppbyggda.
4. Sluta när samtliga kvantifikatorer har hela formeln som räckvidd.

Exempel 55: Skriv $\forall xP(x) \rightarrow \forall yQ(x, y)$ på PNF.

Lösning: De två första förekomsterna av x är bundna, medan den tredje är fri. Förekomsterna av y är bundna. Detta innebär att vi måste byta variabel i den första kvantifikatorn, och vi måste välja en ny variabel. Vi får följande svit av logiska ekvivalenser.

$\forall xP(x) \rightarrow \forall yQ(x, y)$	$\Leftrightarrow$	[regel (2)]
$\forall zP(z) \rightarrow \forall yQ(x, y)$	$\Leftrightarrow$	[regel (12)]
$\exists z(P(z) \rightarrow \forall yQ(x, y))$	$\Leftrightarrow$	[regel (10)]
$\exists z\forall y(P(z) \rightarrow Q(x, y))$		

Den sista formeln är på PNF. Betrakta följande alternativa lösning.

$\forall xP(x) \rightarrow \forall yQ(x, y)$	$\Leftrightarrow$	[regel (2)]
$\forall zP(z) \rightarrow \forall yQ(x, y)$	$\Leftrightarrow$	[regel (10)]
$\forall y(\forall zP(z) \rightarrow Q(x, y))$	$\Leftrightarrow$	[regel (12)]
$\forall y\exists z(P(z) \rightarrow Q(x, y))$		

Återigen är den sista formeln på PNF. Vi har här ett exempel på att problemet att skriva en formel på PNF inte nödvändigtvis ger en entydig lösning.

Exempel 56: Skriv $\exists xP(x, y) \rightarrow (Q(x) \rightarrow \neg\exists zP(x, z))$ på PNF.

Lösning: Vi genomför först lämpliga variabelbyten. För att göra det noterar vi att förekomsten av y är fri, och att de två sista förekomsterna av x är fri. Detta innebär att vi måste byta variabel i den första existenskvantifikatorn, och att vi inte får välja x, y eller z (varför?). Vi tar då en indicerad variabel.

$\exists xP(x, y) \rightarrow (Q(x) \rightarrow \neg\exists zP(x, z))$	$\Leftrightarrow$	[regel (3)]
$\exists x_1P(x_1, y) \rightarrow (Q(x) \rightarrow \neg\exists zP(x, z))$	$\Leftrightarrow$	[regel (13)]
$\forall x_1(P(x_1, y) \rightarrow (Q(x) \rightarrow \neg\exists zP(x, z)))$	$\Leftrightarrow$	[regel (4)]
$\forall x_1(P(x_1, y) \rightarrow (Q(x) \rightarrow \forall z\neg P(x, z)))$	$\Leftrightarrow$	[regel (10)]
$\forall x_1(P(x_1, y) \rightarrow (\forall z\, Q(x) \rightarrow \neg P(x, z)))$	$\Leftrightarrow$	[regel (10)]
$\forall x_1\forall z(P(x_1, y) \rightarrow (Q(x) \rightarrow \neg P(x, z)))$		

Därmed är vi klara. Vi har ovan successivt tagit en kvantifikator i taget och flyttat den ett steg i taget tills den hamnat så att den har hela formeln som räckvidd.

Exempel 57: Skriv $\forall xP(x) \leftrightarrow Q(x)$ på PNF.

Lösning: Vi eliminerar först symbolen för materiell ekvivalens, byter därefter variabler och flyttar till slut ut kvantifikatorerna.

$\forall xP(x) \leftrightarrow Q(x)$	$\Leftrightarrow$	[regel (1)]
$(\forall xP(x) \rightarrow Q(x)) \wedge (Q(x) \rightarrow \forall xP(x))$	$\Leftrightarrow$	[regel (2)]
$(\forall yP(y) \rightarrow Q(x)) \wedge (Q(x) \rightarrow \forall zP(z))$	$\Leftrightarrow$	[regel (12)]
$\exists y(P(y) \rightarrow Q(x)) \wedge (Q(x) \rightarrow \forall zP(z))$	$\Leftrightarrow$	[regel (7)]
$\exists y((P(y) \rightarrow Q(x)) \wedge (Q(x) \rightarrow \forall zP(z)))$	$\Leftrightarrow$	[regel (10)]
$\exists y((P(y) \rightarrow Q(x)) \wedge \forall z(Q(x) \rightarrow P(z)))$	$\Leftrightarrow$	[regel (7)]
$\exists y\forall z((P(y) \rightarrow Q(x)) \wedge (Q(x) \rightarrow P(z)))$		

Det är viktigt att ha en total kontroll över delformlernas struktur för att precis veta hur man skall hantera kvantifikatorerna.

Det finns ett flertal användningar av ovanstående standardiserade skrivsätt. Det är till exempel vanligt att man mäter komplexiteten hos en predikatlogisk formel med antalet kvantifikatorväxlingar i den. Ju fler desto mer komplex formel. Så innehåller exempelvis formeln $\exists y\forall z((P(y) \rightarrow Q(x)) \wedge (Q(x) \rightarrow P(z)))$ i exempel 57 en kvantifikatorväxling. Den är en Σ_2- formel, där symbolen Σ anger att första kvantifikatorn är en existenskvantifikator. Formeln $\forall x_1\forall z(P(x_1, y) \rightarrow (Q(x) \rightarrow \neg P(x, z)))$ i exempel 56 är en Π_1-formel. Första kvantifikatorn är en allkvantifikator, och inga kvantifikatorväxlingar förekommer. Π_1-formler är en viktig delklass av formler inom aritmetik. Goldbachs gissning, varje jämnt heltal större än två kan skrivas som en summa av två primtal, är en Π_1-sats. Det är okänt huruvida denna förmodan är sann eller ej. Den sats Gödel konstruerar i sitt berömda bevis av aritmetikens ofullständighet är också en Π_1-sats.

Det härledningssystem PROLOG använder, resolution, förutsätter att formler är skrivna på ett standardiserat sätt. Ett moment i denna standardisering är att skriva formler på PNF.

Naturlig deduktion i predikatlogik

I det följande presenteras ett system för naturlig deduktion i predikatlogik. I deduktionerna utnyttjas endast satser, men detta är ingen allvarlig begränsning, eftersom $\models \varphi(v_1, \ldots, v_n)$ om och endast om $\models \forall v_1 \ldots \forall v_n \varphi(v_1, \ldots, v_n)$, och vi därmed kommer åt precis alla giltiga slutledningar och inga andra. Det system som presenteras här är en utvidgning av det system för naturlig deduktion i satslogik vi redan studerat. Detta avsnitt innehåller dessutom en del upprepningar av vad som redan tidigare är sagt, men det kan vara lämpligt att ha allt samlat på ett ställe. Definitioner är egentligen som i satslogik, men även de finns delvis upprepade i detta avsnitt.

Situationen i predikatlogik, vad gäller möjligheten att avgöra huruvida en slutledning är giltig eller ej, är mycket mer komplicerad än i satslogik. I satslogik finns tabellmetoden med hjälp av vilken vi princip, om än inte alltid i praktiken, kan avgöra huruvida en slutledning är giltig. I predikatlogik finns ingen motsvarande princip. Vi är hänvisade till att använda strukturer (modeller), om vi vill undersöka huruvida en slutledning är giltig, och det finns ju ingen mekanisk metod, likt tabellmetoden, för hantering av strukturer. Emellertid är det så att man kan skapa en kalkyl, ett formellt system, sådant att, givet att en slutledning är giltig, så kan slutsatsen härledas från premissmängden. Ofta är det avsevärt mycket enklare att arbeta inom en dylik kalkyl än att använda strukturer.

Vi inleder med att repetera begreppen *bunden* och *fri* variabelförekomst. Med en kvantifikators *räckvidd* i en formel φ menas kvantifikatorn själv tillsammans med den kortaste delen av formeln som följer efter kvantifikatorn och som själv är en vbf.

Exempel 58: $\forall x(R(x, y) \rightarrow P(x))$ Här är kvantifikatorns räckvidd hela formeln.
$\underline{\forall xR(x, y)} \rightarrow P(x)$ Här är kvantifikatorns räckvidd den understrukna delen av formeln.

En variabel v är *bunden* om den förekommer inom räckvidden av en kvantifikator $\forall v$ eller $\exists v$. En variabel är *fri* om den inte är bunden. I exempel 58 ovan är i den första formeln samtliga förekomster av x bundna medan förekomsten av y är fri. I den andra formeln är de två första förekomsterna av x bundna medan övriga variabelförekomster är fria. En *sats* är en vbf som saknar fria variabler.

Om φ är en formel så indikerar $\varphi(v_1, \ldots, v_n)$ att de fria variablerna i φ finns bland variablerna $v_1, \ldots, v_n$. Observera att φ inte behöver innehålla alla dessa variabler. Normalt anger vi precis vilka fria variabler som ingår.

Definition 23: En term t är fri för variabeln u i formeln $\varphi(u)$ om och endast om t inte innehåller någon variabel v som blir bunden av en kvantifikator $\exists v$ eller $\forall v$ om alla fria förekomster av u i $\varphi(u)$ ersätts med t.

Exempel 59: Låt $\varphi(x, y)$ vara $\forall xP(x, y) \rightarrow \forall zQ(z, x)$. I denna formel är y och den sista förekomsten av x fria variabelförekomster.

Det gäller att $g(y, z)$ är fri för y i $\varphi(x, y)$, eftersom vi, om vi ersätter fria förekomsten av y i $\varphi(x, y)$ med $g(y, z)$, erhåller formeln $\forall xP(x, g(y, z)) \rightarrow \forall zQ(z, x)$, och termen $g(y, z)$ inte innehåller något x som kommer inom räckvidden av $\forall x$.

Däremot är $g(y, z)$ inte fri för x i $\varphi(x, y)$, eftersom vi får $\forall xP(x, y) \rightarrow \forall zQ(z, g(y, z))$ då vi ersätter den fria förekomsten av x med $g(y, z)$, och z då kommer inom räckvidden av kvantifikatorn $\forall z$.

y är fri för x i $\varphi(x, y)$, eftersom vi efter substitution får $\forall xP(x, y) \to \forall zQ(z, y)$, och y inte blir bunden av $\forall z$.

x är inte fri för y i $\varphi(x, y)$, eftersom vi får $\forall xP(x, x) \to \forall zQ(z, x)$ efter substitution, och det x vi ersätter y med blir bundet av kvantifikatorn $\forall x$.

Observera att en variabel v alltid är fri för v, dvs för sig själv, och att en konstant alltid är fri för en variabel. Tänk efter varför det är så.

Deduktionsreglerna för konnektiven är som för satslogik med det tillägget att vi nu kan använda godtyckliga predikatlogiska satser i reglerna. Vi behöver dessutom regler för introduktion och elimination av kvantifikatorer. För fullständighets skull presenterar vi åter de redan kända deduktionsreglerna från satslogiken.

Grundläggande deduktionsregler

Introduktionsregler		Eliminationsregler	
$\dfrac{\varphi \quad \neg\varphi}{\bot}$	$\bot$I		
$\dfrac{\begin{array}{c}\varphi\\ \bot\end{array}}{\neg\varphi}$	$\neg$I	$\dfrac{\begin{array}{c}\neg\varphi\\ \bot\end{array}}{\varphi}$	$\neg$E
$\dfrac{\varphi \quad \psi}{\varphi \wedge \psi}$	$\wedge$I	$\dfrac{\varphi \wedge \psi}{\varphi} \qquad \dfrac{\varphi \wedge \psi}{\psi}$	$\wedge$E
$\dfrac{\varphi}{\varphi \vee \psi} \qquad \dfrac{\psi}{\varphi \vee \psi}$	$\vee$I	$\dfrac{\varphi \vee \psi \quad \varphi \to \theta \quad \psi \to \theta}{\theta}$	$\vee$E
$\dfrac{\begin{array}{c}\varphi\\ \psi\end{array}}{\varphi \to \psi}$	$\to$I	$\dfrac{\varphi \to \psi \quad \varphi}{\psi}$	$\to$E
$\dfrac{\varphi \to \psi \quad \psi \to \varphi}{\varphi \leftrightarrow \psi}$	$\leftrightarrow$I	$\dfrac{\varphi \leftrightarrow \psi}{\varphi \to \psi} \qquad \dfrac{\varphi \leftrightarrow \psi}{\psi \to \varphi}$	$\leftrightarrow$E
$\dfrac{\varphi(c)}{\forall v\varphi(v)}$	$\forall$I	$\dfrac{\forall v\varphi(v)}{\varphi(t)}$	$\forall$E
$\dfrac{\varphi(t)}{\exists v\varphi(v)}$	$\exists$I	$\dfrac{\exists v\varphi(v) \quad \begin{array}{c}\varphi(c)\\ \vdots\\ \psi\end{array}}{\psi}$	$\exists$E
		Regler för identitetssymbolen	
$t = t$	Refl	$\dfrac{t = u \quad \varphi}{\psi}$	Subst

Kom ihåg att φ respektive $\neg\varphi$ är provisoriska premisser i härledningsreglerna $\neg$I, $\neg$E och $\to$I.

Reglerna för introduktion och elimination av kvantifikatorer är förknippade med villkor som måste vara uppfyllda.

I regeln $\forall$I måste c vara namn på ett godtyckligt element för att generaliseringen skall vara giltig. Detta innebär att c inte får finnas med i någon premiss, och inte heller i formeln $\varphi(v)$. Villkoret som måste vara uppfyllt för att få använda regeln $\forall$E är att termen t är sluten. Normalt väljs här en konstant.

Villkoret för $\exists$I är detsamma som för $\forall$E, dvs att termen t är sluten. Regeln $\exists$E är ganska besvärlig. Här vill vi härleda en sats ψ från en existenssats. Det vi vet, om vi vet att en existenssats $\exists v\varphi(v)$ är sann är att något element satisfierar $\varphi(v)$, men vi vet inte vilket. Vi inför då ett namn på en individ vi antar satisfiera $\varphi(v)$. Denna sats, $\varphi(c)$, är då en provisorisk premiss. Det c vi väljer måste vara en ny konstant, dvs den får inte ingå tidigare i härledningen. Om vi kan härleda ψ med hjälp av den provisoriska premissen $\varphi(c)$ tillsammans med övriga premisser, och c inte ingår i ψ, så kan vi dra slutsatsen att ψ följer från $\exists v\varphi(v)$ och avföra $\varphi(c)$ från premissmängden. Se exempel 63 nedan för exemplifiering.

Refl uttrycker att likhetsrelationen är reflexiv. t är en sluten term. Formeln får införas på en rad vilken som helst, är inte en premiss och beror inte på några andra premisser.
I Subst är t och u slutna termer, och ψ uppstår ur φ genom att förekomster av t ersätts med u.

Observera att samtliga deduktionsregler arbetar mot huvudoperatorn i en sats!

Definition 24: En *härledning* är en ändlig svit av konsekutivt numrerade rader, som var och en innehåller en premissnummermängd och en sats.

Vi ger först ett exempel och visar sedan hur premissnummermängder konstrueras för de nya deduktionsreglerna.

Exempel 60: Visa att $\forall x\varphi(x) \wedge \forall x\psi(x) \vdash \forall x(\varphi(x) \wedge \psi(x))$.

Lösning:	{1}	1	$\forall x\varphi(x) \wedge \forall x\psi(x)$	Premiss
	{1}	2	$\forall x\varphi(x)$	1, $\wedge$E
	{1}	3	$\forall x\psi(x)$	1, $\wedge$E
	{1}	4	$\varphi(c)$	2, $\forall$E, c ny konstant
	{1}	5	$\psi(c)$	3, $\forall$E
	{1}	6	$\varphi(c) \wedge \psi(c)$	4, 5, $\wedge$I
	{1}	7	$\forall x(\varphi(x) \wedge \psi(x))$	6, $\forall$I

Alltså $\forall x\varphi(x) \wedge \forall x\psi(x) \vdash \forall x(\varphi(x) \wedge \psi(x))$.

Premissnummermängder konstrueras enligt följande.

1. Reglerna för de satslogiska konnektiven fungerar som tidigare.

2. En sats får införas på ny rad, om den följer av en sats som förekommer på en rad med lägre radnummer i sekvensen med någon av reglerna $\forall$E eller $\exists$I. Som premissnummermängd tas premissnummermängden för raden på vilken $\varphi(c)$ respektive $\varphi(t)$ förekommer (se nedan exempel 62).

3. Satsen $\forall v\varphi(v)$ får införas på en ny rad med $\forall$I, om $\varphi(c)$ förekommer på en rad med lägre radnummer, och konstanten c inte förekommer i $\varphi(v)$ eller i någon sats som förekommer på en rad vars radnummer är element i den premissnummermängd som hör till den rad på vilken $\varphi(c)$ finns. Som premissnummermängd tas den åberopade radens (se nedan exempel 61). Villkoren på c är synnerligen viktiga, eftersom de skall garantera att c är en godtycklig konstant.

4. En härledning med $\exists$E har följande struktur.

X	i	$\exists v\varphi(v)$	
{j}	j	$\varphi(c)$	Provisorisk premiss
Y	k	ψ	
$(X\cup Y)$ - {j}	l	ψ	i, j, k, $\exists$E

Här skall $i < j < k < l$. När premissen på rad j införs skall man välja en ny konstant, eftersom det skall vara en godtycklig sådan. Konstanten c får inte heller ingå i ψ.

Detta innebär att, om $\exists v\varphi(v)$ förekommer på en rad med radnummer i, och $\varphi(c)$ förekommer på en rad j, där $j > i$, och ψ förekommer på en rad k, med $k > j$, och c inte förekommer på en rad vars radnummer är element i premissnummermängden på rad k, utom i $\varphi(c)$, så får ψ införas på ny rad med $\exists$E. Som premissnummer tas alla premissnummer på raderna i och k utom j (se exempel 63).

5. Satsen $t = t$ får införas på ny rad med Refl. Som premissnummermängd tas $\varnothing$ (se exempel 76).

6. Vid användning av Subst väljs unionen av de premissnummermängder, vars satser vi använder (se exempel 77).

Definitioner av begreppen *härledning*, *härledning från premissmängd* Γ, *slutsats*, *bevis* och *teorem* är desamma som i satslogik, och vi hänvisar därför till definitionerna 2:17 och 2:18.

Exempel 61: Visa att $\forall x\varphi(x) \dashv\vdash \forall y\varphi(y)$.

Lösning:

{1}	1	$\forall x\varphi(x)$	Premiss
{1}	2	$\varphi(c)$	1, $\forall$E, där c ej förekommer i $\varphi(x)$
{1}	3	$\forall y\varphi(y)$	2, $\forall$I

$\therefore\ \forall x\varphi(x) \vdash \forall y\varphi(y)$.

{1}	1	$\forall y\varphi(y)$	Premiss
{1}	2	$\varphi(c)$	1, $\forall$E, där c ej förekommer i $\varphi(y)$
{1}	3	$\forall x\varphi(x)$	2, $\forall$I

$\therefore\ \forall y\varphi(y) \vdash \forall x\varphi(x)$.

I båda ovanstående härledningar måste konstanten c väljas så att c är en ny konstant, dvs inte ingår i premissen på rad 1.

Exempel 62: Visa att $\forall x\varphi(x) \vdash \exists x\varphi(x)$.

Lösning:

{1}	1	$\forall x\varphi(x)$	Premiss
{1}	2	$\varphi(c)$	1, $\forall$E
{1}	3	$\exists x\varphi(x)$	2, $\exists$I

$\therefore\quad \forall x\varphi(x) \vdash \exists x\varphi(x)$.

Exempel 63: Visa att $\forall x(\varphi(x) \rightarrow \psi) \vdash \exists x\varphi(x) \rightarrow \psi$ där ψ är en sats.

Lösning:

{1}	1	$\forall x(\varphi(x) \rightarrow \psi)$	Premiss
{2}	2	$\exists x\varphi(x)$	Prov prem (för $\rightarrow$I)
{3}	3	$\varphi(c)$	Prov prem (för $\exists$E), c ny konstant
{1}	4	$\varphi(c) \rightarrow \psi$	1, $\forall$E
{1, 3}	5	ψ	3, 4 $\rightarrow$E
{1, 2}	6	ψ	2, 3, 5, $\exists$E
{1}	7	$\exists x\varphi(x) \rightarrow \psi$	2, 6, $\rightarrow$I

$\therefore\quad \forall x(\varphi(x) \rightarrow \psi) \vdash \exists x\varphi(x) \rightarrow \psi$.
Tänk noga genom hur $\exists$E använts enligt mönstret

{2}	2	$\exists x\varphi(x)$	
{3}	3	$\varphi(c)$	Prov prem
{1, 3}	5	ψ	
({2}$\cup${1, 3}) - {3}	6	ψ	2, 3, 5 $\exists$E

Villkoret att x inte är fri i ψ, att ψ är en sats, behövs på rad 4-6, eftersom detta villkor garanterar att c inte förekommer i ψ efter instantieringen på rad 4. På rad 3 inför vi en provisorisk premiss där vi antar att c är ett namn på en individ som satisfierar $\varphi(x)$. Vi väljer en ny konstant, dvs en konstant som inte förekommit tidigare i härledningen. Satsen ψ på rad 5 beror på premisserna på rad 1 och rad 3. Regeln $\exists$E säger då att vi kan ersätta premissen på rad 3 med premissen på rad 2 i premissnummermängden, under förutsättning att c uppfyller de givna villkoren. När vi använder regeln $\rightarrow$I, stryker vi premissen på rad 2, och inför denna sats som försats i en implikation.

Exempel 64: Vi visar i detta exempel en felaktig användning av $\forall$I. Det gäller att man inte kan härleda $\forall xP(x)$ från $\exists xP(x)$ där P är en enställig predikatsymbol. Med följande missbruk av $\forall$I går det.

{1}	1	$\exists xP(x)$	Premiss
{2}	2	$P(c)$	Prov prem (för $\exists$E)
{2}	3	$\forall xP(x)$	2, FEL

Här kan vi inte använda $\forall$I eftersom c ingår i premissen på rad två.
Observera också att det inte gäller att $\exists xP(x) \vdash P(c)$, eftersom $P(c)$ inte följer av $\exists xP(x)$ med hjälp av $\exists$E.

Exempel 65: Visa att $\exists x \exists y \varphi(x, y) \vdash \exists y \exists x \varphi(x, y)$.

Lösning:

{1}	1	$\exists x \exists y \varphi(x, y)$	Premiss
{2}	2	$\exists y \varphi(c_1, y)$	Prov prem (för $\exists$E), c_1 ny konstant
{3}	3	$\varphi(c_1, c_2)$	Prov prem (för $\exists$E), c_2 ny konstant
{3}	4	$\exists x \varphi(x, c_2)$	3, $\exists$I
{3}	5	$\exists y \exists x \varphi(x, y)$	4, $\exists$I
{2}	6	$\exists y \exists x \varphi(x, y)$	2, 3, 5, $\exists$E, c_2 förekommer inte i $\varphi(x, y)$
{1}	7	$\exists y \exists x \varphi(x, y)$	1, 2, 6, $\exists$E, c_1 förekommer inte i $\varphi(x, y)$

I denna härledning väljer vi på rad två en ny konstant c_1, dvs en konstant som inte förekommit tidigare i härledningen, och så även på rad tre. Därefter är $\exists$I oproblematisk. Vi måste sedan med hjälp av $\exists$E bli av med de extra premisserna på raderna tre respektive två. På rad sex gör vi oss av med den extra premissen på rad tre. Den delen av härledningen ser ut på följande sätt:

{2}	2	$\exists y \varphi(c_1, y)$	Prov prem
{3}	3	$\varphi(c_1, c_2)$	Prov prem
{3}	5	$\exists y \exists x \varphi(x, y)$	4, $\exists$I
{2}	6	$\exists y \exists x \varphi(x, y)$	2, 3, 5, $\exists$E

Eftersom c_2 är en ny konstant i rad tre och inte förekommer i satsen på rad fem, kan vi säga att $\exists y \exists x \varphi(x, y)$ följer av $\exists y \varphi(c_1, y)$ i stället för av $\varphi(c_1, c_2)$, och ändrar därför i premissnummermängden. Analogt får vi på rad sju att $\exists y \exists x \varphi(x, y)$ följer av $\exists x \exists y \varphi(x, y)$ med hjälp av $\exists$E och ändrar innehållet i premissnummermängden.

Exempel 66: Visa att $\{\exists x P(x), \exists x \neg P(x)\} \vdash \exists x \exists y (P(x) \wedge \neg P(y))$, där P är en enställig predikatsymbol.

Lösning:

{1}	1	$\exists x P(x)$	Premiss
{2}	2	$\exists x \neg P(x)$	Premiss
{3}	3	$P(c_1)$	Prov prem (för $\exists$E)
{4}	4	$\neg P(c_2)$	Prov prem (för $\exists$E)
{3, 4}	5	$P(c_1) \wedge \neg P(c_2)$	3, 4, $\wedge$I
{3, 4}	6	$\exists y (P(c_1) \wedge \neg P(y))$	5, $\exists$I
{3, 4}	7	$\exists x \exists y (P(x) \wedge \neg P(y))$	6, $\exists$I
{2, 3}	8	$\exists x \exists y (P(x) \wedge \neg P(y))$	2, 4, 7, $\exists$E
{1, 2}	9	$\exists x \exists y (P(x) \wedge \neg P(y))$	1, 3, 8, $\exists$E

Därmed har vi $\{\exists x P(x), \exists x \neg P(x)\} \vdash \exists x \exists y (P(x) \wedge \neg P(y))$.

Notera att vi inte behöver påpeka här att c_1 respektive c_2 är nya konstanter. Det syns ju, eftersom vi i detta exempel arbetar med en formel i det predikatlogiska språket och inte i metaspråket.

Vi vill kunna använda oss av några härledda deduktionsregler för att förenkla härledningar. Enligt exempel 61 gäller det att

$$\frac{\forall x \varphi(x)}{\forall y \varphi(y)}$$

Här väljer vi normalt att låta y vara en ny variabel. Det är dock viktigt att vi väljer y så att y är fri för x i $\varphi(x)$, dvs så att y inte kommer inom räckvidden av en kvantifikator $\forall y$ eller $\exists y$ i $\varphi(x)$. Dylika kan finnas i $\varphi(x)$. Väljer vi en variabel som inte finns i $\varphi(x)$ drabbas vi inte av detta problem. Vi får alltså byta en bunden allkvantifierad variabel. Att vi även får byta bundna existenskvantifierade variabler framgår av följande exempel.

Exempel 67: Visa att $\exists x\varphi(x) \vdash \exists y\varphi(y)$, där y är en variabel som inte förekommer i $\varphi(x)$.

Lösning:

{1}	1	$\exists x\varphi(x)$	Premiss
{2}	2	$\varphi(c)$	Prov prem ($\exists$E), c ny konstant
{2}	3	$\exists y\varphi(y)$	2, $\exists$I
{1}	4	$\exists y\varphi(y)$	1, 2, 3, $\exists$E

Vi har då att $\dfrac{\exists x\varphi(x)}{\exists y\varphi(y)}$

Båda dessa regler betecknar vi med BBV, Byte av Bunden Variabel. Vi inser också att härledningarna av båda dessa regler inte beror på val av variabel, bara vi ser till att välja ny variabel vid bytet.

Vi vill också har några härledda deduktionsregler för sambandet mellan kvantifikatorer och negation. Det bör också påpekas att det sätt man kommer på lösningen av ett matematiskt eller logiskt problem inte är det sätt på vilket man presenterar lösningen av problemet. För att kunna genomföra en härledning i ett logiskt system, måste man ha en strategi för hur denna härledning skall kunna gå till. I exemplen nedan inleds lösningen med en strategidiskussion. Observera emellertid att denna inte ingår i presentationen av själva lösningen av problemet.

Exempel 68: Visa att $\exists x\neg\varphi(x) \vdash \neg\forall x\varphi(x)$.

Lösning: I denna uppgift skall vi bevisa en negation. Därför antar vi $\forall x\varphi(x)$ (rad två). Ur detta antagande skall vi sedan härleda en motsägelse. Lyckas vi med detta får vi dra slutsatsen att $\neg\forall x\varphi(x)$. Härledningen kan se ut enligt nedan.

{1}	1	$\exists x\neg\varphi(x)$	Premiss
{2}	2	$\forall x\varphi(x)$	Prov prem (för $\neg$I)
{3}	3	$\neg\varphi(c)$	Prov prem (för $\exists$E), c ny konstant
{2}	4	$\varphi(c)$	2, $\forall$E
{2, 3}	5	$\perp$	3, 4, $\perp$I
{1, 2}	6	$\perp$	1, 3, 5, $\exists$E
{1}	7	$\neg\forall x\varphi(x)$	2, 6, $\neg$I

I härledningen har vi att utnyttja existenssatsen vi har i premissen. Då låter vi c vara en ny konstant, och inför $\neg\varphi(c)$ (rad tre) som extra premiss. Från denna extra premiss härleder vi sedan en motsägelse (rad fem), och eftersom c väljs så att c inte ingår i φ och inte ingår i $\perp$, kan vi med hjälp av $\exists$E ersätta premissen på rad tre med premissen på rad 1 i premissnummermängden på rad sex.

Det gäller alltså att $\dfrac{\exists x\neg\varphi(x)}{\neg\forall x\varphi(x)}$.

Exempel 69: Visa att $\forall x\neg\varphi(x) \vdash \neg\exists x\varphi(x)$.

Lösning: Strategin här är analog med den i föregående exempel.

Lösning:

{1}	1	$\forall x \neg\varphi(x)$	Premiss
{2}	2	$\exists x \varphi(x)$	Prov prem (för $\neg$I)
{3}	3	$\varphi(c)$	Prov prem (för $\exists$E), c ny konstant
{1}	4	$\neg\varphi(c)$	1, $\forall$E
{1, 3}	5	$\perp$	3, 4, $\perp$I
{1, 2}	6	$\perp$	2, 3, 5, $\exists$E
{1}	7	$\neg\exists x\varphi(x)$	2, 6, $\neg$I

Därmed är det klart att $\dfrac{\forall x\neg\varphi(x)}{\neg\exists x\varphi(x).}$

Exempel 70: Visa att $\neg\exists x\varphi(x) \vdash \forall x\neg\varphi(x)$.

Lösning: Målet här är att vi ska kunna sluta oss till $\forall x\neg\varphi(x)$. Detta kan vi göra om vi på en tidigare rad har formeln $\neg\varphi(c)$ och c är en godtycklig konstant. För att härleda $\neg\varphi(c)$ är den normala strategin att anta $\varphi(c)$ för att härleda en motsägelse.

{1}	1	$\neg\exists x\varphi(x)$	Premiss
{2}	2	$\varphi(c)$	Prov prem (för $\neg$I), c ny konstant
{2}	3	$\exists x\varphi(x)$	2, $\exists$I
{1, 2}	4	$\perp$	1, 3, $\perp$I
{1}	5	$\neg\varphi(c)$	2, 4, $\neg$I
{1}	6	$\forall x\neg\varphi(x)$	5, $\forall$I, Observera att c inte ingår i den sats vars nummer finns i premiss-mängden till rad 5.

Då är det klart att $\dfrac{\neg\exists x\varphi(x)}{\forall x\neg\varphi(x).}$

Exempel 71: Visa att $\neg\forall x\varphi(x) \vdash \exists x\neg\varphi(x)$.

Lösning: I detta problem är det nära till hands att försöka använda strategin från förra exemplet. Dock blir följande fel:

{1}	1	$\neg\forall x\varphi(x)$	Premiss
{2}	2	$\varphi(c)$	Prov prem
{2}	3	$\forall x\varphi(x)$	2 FEL, eftersom c ingår i en premiss och därför inte är godtycklig!

Vi genomför i stället ett motsägelsebevis och antar för motsägelse att $\neg\exists x\neg\varphi(x)$. I härledningen åker vi sedan snålskjuts på föregående exempel.

{1}	1	$\neg\forall x\varphi(x)$	Premiss
{2}	2	$\neg\exists x\neg\varphi(x)$	Prov prem (för $\neg$E)
{2}	3	$\forall x\neg\neg\varphi(x)$	2, föreg ex
{2}	4	$\neg\neg\varphi(c)$	3, $\forall$E, c ny konstant
{2}	5	$\varphi(c)$	4, DN

{2}	6	$\forall x\varphi(x)$	2, $\forall$I
{1, 2}	7	$\perp$	1, 6, $\perp$I
{1}	8	$\exists x\neg\varphi(x)$	2, 7, $\neg$E

Då är den fjärde regeln slutligen bevisad, och det är alltså klart att även

$$\frac{\neg\forall x\varphi(x)}{\exists x\neg\varphi(x).}$$

Samtliga dessa fyra härledda deduktionsregler betecknar vi med Q.

Därmed är samtliga härledda deduktionsregler presenterade, och vi sammanfattar dem i en tabell.

Härledda deduktionsregler, sammanfattning

Regel				Namn	Förkortning
$\dfrac{\neg(\varphi \lor \psi)}{\neg\varphi \land \neg\psi}$	$\dfrac{\neg\varphi \land \neg\psi}{\neg(\varphi \lor \psi)}$	$\dfrac{\neg(\varphi \land \psi)}{\neg\varphi \lor \neg\psi}$	$\dfrac{\neg\varphi \lor \neg\psi}{\neg(\varphi \land \psi)}$	De Morgans Lagar	DM
$\begin{array}{c}\varphi \lor \psi \\ \underline{\neg\varphi} \\ \psi\end{array}$	$\begin{array}{c}\varphi \lor \psi \\ \underline{\neg\psi} \\ \varphi\end{array}$			Disjunktiv Syllogism	DS
$\begin{array}{c}\varphi \to \psi \\ \underline{\psi \to \theta} \\ \varphi \to \theta\end{array}$				Hypotetisk Syllogism	HS
$\dfrac{\neg\neg\varphi}{\varphi}$	$\dfrac{\varphi}{\neg\neg\varphi}$			Dubbel Negation	DN
$\begin{array}{c}\varphi \to \psi \\ \underline{\neg\psi} \\ \neg\varphi\end{array}$				Modus Tollens	MT
$\dfrac{\neg(\varphi \to \psi)}{\varphi \land \neg\psi}$				Negerad Implikation	NI
$\dfrac{\exists x\neg\varphi(x)}{\neg\forall x\varphi(x)}$	$\dfrac{\forall x\neg\varphi(x)}{\neg\exists x\varphi(x)}$	$\dfrac{\neg\exists x\varphi(x)}{\forall x\neg\varphi(x)}$	$\dfrac{\neg\forall x\varphi(x)}{\exists x\neg\varphi(x)}$	Kvantifi-katorregler	Q

Innan vi exemplifierar reglerna för identitet skall vi visa ytterligare några exempel på härledningar.

Exempel 72: Visa att $\forall x(\varphi(x) \to \psi(x)) \vdash \forall x\varphi(x) \to \forall x\psi(x)$.

Lösning: I denna uppgift skall vi härleda en implikation. Vi antar därför försatsen $\forall x\varphi(x)$ som en extra premiss, härleder $\forall x\psi(x)$ och stryker denna extra premiss ur premissmängden när vi skriver in $\forall x\varphi(x) \to \forall x\psi(x)$ i härledningen.

{1}	1	$\forall x(\varphi(x) \to \psi(x))$	Premiss
{2]	2	$\forall x\varphi(x)$	Prov prem (för $\to$I)
{2}	3	$\varphi(c)$	2, $\forall$E, c ny konstant
{1}	4	$\varphi(c) \to \psi(c)$	1, $\forall$E
{1, 2}	5	$\psi(c)$	3, 4, $\to$E
{1, 2}	6	$\forall x\psi(x)$	5, $\forall$I, c förekommer inte i satserna på rad 1 eller 2
{1}	7	$\forall x\varphi(x) \to \forall x\psi(x)$	2, 6, $\to$I

Alltså gäller $\forall x(\varphi(x) \to \psi(x)) \vdash \forall x\varphi(x) \to \forall x\psi(x)$.

Exempel 73: Visa att $\exists x\varphi(x) \to \psi \vdash \forall x(\varphi(x) \to \psi)$, där ψ är en sats.

Lösning: Vi skall härleda en allkvantifierad implikation, och behöver då fram satsen $\varphi(c) \to \psi$. Under förutsättning att c är godtycklig, kan vi allkvantifiera denna. För att producera implikationen antar vi försatsen $\varphi(c)$ och härleder ψ. Observera att c inte längre ingår i en premiss när vi infört $\varphi(c) \to \psi$.

{1}	1	$\exists x\varphi(x) \to \psi$	Premiss
{2}	2	$\varphi(c)$	Prov prem (för $\to$I), c ny konst
{2}	3	$\exists x\varphi(x)$	2, $\exists$I
{1, 2}	4	ψ	1, 3, $\to$E
{1}	5	$\varphi(c) \to \psi$	2, 4, $\to$I
{1}	6	$\forall x(\varphi(x) \to \psi)$	5, $\forall$I, obs att c inte ingår i satsen på rad 1

Då är det klart att $\exists x\varphi(x) \to \psi \vdash \forall x(\varphi(x) \to \psi)$.

Exempel 74: Visa att $\exists x(Q(x) \wedge \forall y(P(y) \to R(y, x))) \vdash \forall x(P(x) \to \exists y(Q(y) \wedge R(x, y)))$.

Lösning: Även i detta exempel skall vi härleda en allkvantifierad implikation, och strategin blir densamma som i föregående uppgift. Vi inför $P(c_2)$ som extra premiss (rad fem). Vi behöver emellertid utnyttja premissens existenssats, och inför därför $Q(c_1) \wedge \forall y(P(y) \to R(y, c_1))$ (rad två) som extra premiss. Idén är sedan att först bryta ner denna sats, för att sedan bygga upp den formel $P(c_2) \to \exists y(Q(y) \wedge R(c_2, y))$ vi vill ha.

{1}	1	$\exists x(Q(x) \wedge \forall y(P(y) \to R(y, x)))$	Premiss
{2}	2	$Q(c_1) \wedge \forall y(P(y) \to R(y, c_1))$	Prov prem (för $\exists$E)
{2}	3	$Q(c_1)$	2, $\wedge$E
{2}	4	$\forall y(P(y) \to R(y, c_1))$	2, $\wedge$E
{5}	5	$P(c_2)$	Prov prem (för $\to$I)
{2}	6	$P(c_2) \to R(c_2, c_1)$	4, $\forall$E
{2, 5}	7	$R(c_2, c_1)$	5, 6, $\to$E
{2, 5}	8	$Q(c_1) \wedge R(c_2, c_1)$	3, 7, $\wedge$I
{2, 5}	9	$\exists y(Q(y) \wedge R(c_2, y))$	8, $\exists$I
{2}	10	$P(c_2) \to \exists y(Q(y) \wedge R(c_2, y))$	5, 9, $\to$I
{2}	11	$\forall x(P(x) \to \exists y(Q(y) \wedge R(x, y)))$	10, $\forall$I
{1}	12	$\forall x(P(x) \to \exists y(Q(y) \wedge R(x, y)))$	1, 2, 11, $\exists$E

Därmed är det klart att

$\exists x(Q(x) \wedge \forall y(P(y) \to R(y, x))) \vdash \forall x(P(x) \to \exists y(Q(y) \wedge R(x, y)))$.

Vi behöver inte påpeka att c_1 och c_2 är godtyckliga eftersom det framgår direkt i formlerna.

Exempel 75: Visa att $\vdash \exists x \forall y \varphi(x, y) \rightarrow \forall y \exists x \varphi(x, y)$.

Lösning:

{1}	1	$\exists x \forall y \varphi(x, y)$	Prov prem (för $\rightarrow$I)
{2}	2	$\forall y \varphi(c_1, y)$	Prov prem (för $\exists$E), c_1 ny konstant
{2}	3	$\varphi(c_1, c_2)$	2, $\forall$E, c_2 ny konstant
{2}	4	$\exists x \varphi(x, c_2)$	3, $\exists$I
{2}	5	$\forall y \exists x \varphi(x, y)$	4, $\forall$I
{1}	6	$\forall y \exists x \varphi(x, y)$	1, 2, 5, $\exists$E
$\varnothing$	7	$\exists x \forall y \varphi(x, y) \rightarrow \forall y \exists x \varphi(x, y)$	1, 6, $\rightarrow$I

Därmed är det klart att vi härlett den önskade formeln från en tom premissmängd.

Observera att det inte går att härleda $\exists x \forall y \varphi(x, y)$ från $\forall y \exists x \varphi(x, y)$. Vad är det för fel på följande försök?

{1}	1	$\forall y \exists x \varphi(x, y)$	Premiss
{1}	2	$\exists x \varphi(x, c_1)$	1, $\forall$E
{3}	3	$\varphi(c_2, c_1)$	Prov prem
{3}	4	$\forall y \varphi(c_2, y)$	FEL, eftersom c_1 förekommer i en premiss i premissmängden till rad 3.

Vi ger avslutningsvis några exempel på användningen av reglerna för identitetsrelationen.

Exempel 76: Visa att $t = u \vdash u = t$, för slutna termer u och t.

Lösning:

{1}	1	$t = u$	Premiss
$\varnothing$	2	$t = t$	Refl
{1}	3	$u = t$	1, 2, Subst

I rad tre har den första förekomsten av t i satsen $t = t$ ersatts med u. På rad två använder vi Refl för att införa satsen $t = t$, som inte beror på någon premiss. Därav en tom premissmängd.

Exempel 77: Visa att $\vdash \forall x \forall y(x = y \rightarrow (P(x) \leftrightarrow P(y)))$.

Lösning:

{1}	1	$c_1 = c_2$	Prov prem
{2}	2	$P(c_1)$	Prov prem
{1, 2}	3	$P(c_2)$	1, 2, Subst
{1}	4	$P(c_1) \rightarrow P(c_2)$	2, 3, $\rightarrow$I
{5}	5	$P(c_2)$	Prov prem
{1, 5}	6	$P(c_1)$	1, 5, Subst
{1}	7	$P(c_2) \rightarrow P(c_1)$	5, 6, $\rightarrow$I
{1}	8	$P(c_1) \leftrightarrow P(c_2)$	4, 7, $\leftrightarrow$I
$\varnothing$	9	$c_1 = c_2 \rightarrow P(c_1) \leftrightarrow P(c_2)$	1, 8, $\rightarrow$I
$\varnothing$	10	$\forall y(c_1 = y \rightarrow (P(c_1) \leftrightarrow P(y))$	9, $\forall$I
$\varnothing$	11	$\forall x \forall y(x = y \rightarrow (P(x) \leftrightarrow P(y))$	10. $\forall$I

c_1 och c_2 ingår inte i några premisser som satsen på rad 9 beror på, eftersom ifrågavarande sats inte beror på några premisser alls.

4 GÖDELS FULLSTÄNDIGHETSSATS

Målet i detta kapitel är att visa Gödels fullständighetssats. Detta resultat bevisades ursprungligen av Kurt Gödel 1930, och säger att en sats φ är ett teorem i en predikatlogisk kalkyl om och endast om φ är en predikatlogisk sanning. Det är emellertid opraktiskt att bevisa detta resultat om det formella system för naturlig deduktion i predikatlogik vi tidigare studerat. Detta beror på att detta system har en relativt omfattande vokabulär, och detta gör att några av bevisen skulle bli väl långa. Vi skall därför beskriva ett formellt system med en något mindre omfattande vokabulär. I detta system är det mödosammare att utföra härledningar, men det är enklare att bevisa påståenden om detta system. De båda systemen är ekvivalenta i den meningen att exakt samma formler kan deduceras i de båda kalkylerna.

Det formella systemet K

Vi presenterar här vokabulär, formationsregler, axiom och deduktionsregler för det system för vilket vi skall bevisa fullständighetssatsen. Avsnittet nedan innehåller mycket upprepningar från tidigare, och finns till enbart för att framställningen skall vara någorlunda fullständig.

Vokabulär

Logiska symboler:	Konnektiver:	$\neg, \rightarrow$
	Kvantikatorer:	$\forall$
	Variabelsymboler:	$x_1, x_2, x_3, \ldots$

Vi kommer dessutom att använda oss av parenteser och kommatecken för att underlätta läsningen av formler.

Icke-logiska symboler:	Konstanter:	$c_1, c_2, c_3, \ldots$
	Funktionssymboler:	f_i^n, där $i, n \geq 1$, n anger ställighet och i är ett index
	Predikatsymboler:	A_i^n, där $i, n \geq 1$, n anger ställighet och i är ett index

Som tidigare specificeras ett predikatlogiskt språk L genom att ange vilka icke-logiska symboler som ingår i L. De logiska symbolerna ingår i varje predikatlogiskt språk. Minst en predikatsymbol måste ingå i ett lexikon, eftersom vi inte annars kan bilda några formler. Detta beror på att vi inte infört någon symbol för identitet.

Formationsregler

Termer är tänkta att peka ut individer, och vi börjar med att rekursivt definiera begreppet *term*.

Definition 1: (i) Alla konstanter är termer.
(ii) Alla variabler är termer.
(iii) Om $t_1, \ldots, t_n$ är termer och f_i^n är en funktionssymbol, så är $f_i^n(t_1, \ldots, t_n)$ en term.

Definition 2: En term som inte innehåller några variabler kallas *sluten*.

Målet är nu att definiera vilka uttryck över vår vokabulär som är välbildade formler (vbf), och vi börjar med att definiera begreppet *atomär formel*.

Definition 3: Om $t_1, \dots, t_n$ är termer och A_i^n är en predikatsymbol, så är $A_i^n(t_1, \dots, t_n)$ en *atomär formel*.

Vi definierar sedan begreppet *välbildad formel* rekursivt.

Definition 4:
(i) Alla atomära formler är vbf.
(ii) Om φ är en vbf, så är $\neg\varphi$ en vbf.
(iii) Om φ och ψ är vbf, så är $(\varphi \to \psi)$ en vbf.
(iv) Om φ är en vbf, så är $\forall x_i\varphi$ en vbf för $i = 1, 2, 3, \dots$

Observera att definitionerna 3 och 4 ovan är kortare än de tidigare givna motsvarande definitionerna. Detta beror förstås på den begränsade vokabulären. Observera också att $\{\neg, \to\}$ är en fullständig konnektivmängd. Övriga konnektiver kan införas som definitioner, eller förkortningar, enligt följande.

$\varphi \vee \psi$ är en förkortning av $\neg\varphi \to \psi$.
$\varphi \wedge \psi$ är en förkortning av $\neg(\varphi \to \neg\psi)$.
$\varphi \leftrightarrow \psi$ är en förkortning av $(\varphi \to \psi) \wedge (\psi \to \varphi)$, dvs $\neg((\varphi \to \psi) \to \neg(\psi \to \varphi))$.

På motsvarande sätt kan existenskvantifikatorn införas som en förkortning genom

$\exists x_i\varphi$ är en förkortning av $\neg\forall x_i\neg\varphi$.

Innebörden av beteckningen $\varphi(v_1, v_2, \dots, v_n)$ är i detta avsnitt att eventuella fria variablerna i formeln φ kan finnas bland variablerna $v_1, v_2, \dots, v_n$. Detta skiljer sig från tidigare användning av beteckningen, eftersom vi tidigare sagt att de eventuella fria variablerna i $\varphi(v_1, v_2, \dots, v_n)$ finns bland $v_1, v_2, \dots, v_n$. Som vanligt är v_i, φ och ψ symboler i ett metaspråk. De ingår ju inte i vårt formella språk. Om formeln φ innehåller precis de fria variablerna $v_1, v_2, \dots, v_n$, så uppfattar vi $\varphi(v_1, v_2, \dots, v_n)$ som en förkortning av satsen $\forall v_1\forall v_2 \dots \forall v_n\varphi(v_1, v_2, \dots, v_n)$.

Axiom

Axiomen i en teori är av två typer, logiska respektive äkta.

De *logiska axiomen* i K är följande där φ, ψ och θ är godtyckliga vbf.

(i) $\varphi \to (\psi \to \varphi)$
(ii) $(\varphi \to (\psi \to \theta)) \to ((\varphi \to \psi) \to (\varphi \to \theta))$
(iii) $(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)$
(iv) $\forall x_i\varphi(x_i) \to \varphi(t)$, där t är en sluten term.
(v) $\forall x_i(\varphi \to \psi) \to (\varphi \to \forall x_i\psi)$, där φ inte innehåller några fria förekomster av x_i.

De *äkta axiomen* specificeras inte, eftersom de varierar från teori till teori. Med *teori* förstår vi en mängd av satser, som är tänkt att vara teorins axiom. Observera att axiomen (i) till (v) egentligen är axiomscheman där vi får olika axiom genom att ersätta φ, ψ respektive θ med vbf.

Exempel 1: $A_1^2(x_1,c_3) \to (\forall x_2 A_3^1(x_2) \to A_1^2(x_1,c_3))$ är en instans av (i), och denna formel är en förkortning av $\forall x_1(A_1^2(x_1,c_3) \to (\forall x_2 A_3^1(x_2) \to A_1^2(x_1,c_3)))$.

Notera att vi i exempel 1 egentligen har brutit mot formationsreglerna ovan. Detta kommer vi att fortsätta med, om det inte försvårar läsbarheten av formler.

Saknas äkta axiom säger vi att vi har en första ordningens predikatlogisk *kalkyl*. K är alltså en första ordningens predikatlogisk kalkyl. Vi kommer att använda K som beteckning både för den teori, som har samtliga instanser av (i) till (v) som axiom, och för det formella system vi just specificerar. I samtliga teorier nedan ingår axiomen i K.

Deduktionsregler

Vårt formella system har två deduktionsregler.

(i)	Modus ponens (MP):	ψ följer av φ och $\varphi \to \psi$, där φ och ψ är vbf.
(ii)	Generalisering (Gen)	$\forall x_i\varphi(x_i)$ följer av $\varphi(c)$, där c är en godtycklig konstant (se $\forall$I).

Begreppen *härledning*, *bevis* och *teorem* definieras som tidigare med tillägget att ett axiom får införas på valfri rad. Ett axiom är ingen premiss. Vi skriver $S \vdash \varphi$, om φ är ett teorem i teorin S, dvs om φ kan härledas från axiomen i S. I stället för $K \vdash \varphi$, skriver vi vanligen $\vdash \varphi$. Om φ kan härledas från teorin S utökad med premissmängden Γ, så skriver vi $S + \Gamma \vdash \varphi$, och om φ kan härledas från teorin S utökad med $\{\psi\}$, så skriver vi $S + \psi \vdash \varphi$. Vi låter dessutom $Th(S) = \{\varphi : S \vdash \varphi\}$, dvs Th(S) är mängden av alla teorem som kan härledas från S. I en del framställningar används ordet ”teori” om mängden Th(S).

Exempel 2: Vi ger ett exempel på en första ordningens teori. Låt språket för teorin innehålla en predikatsymbol A_1^2, men varken funktionssymboler eller konstanter. Vi skriver $x_i < x_j$ i stället för $A_1^2(x_i,x_j)$. Teorin innehåller två äkta axiom.

(i)	$\forall x_1 \neg(x_1 < x_1)$	(irreflexivitet)
(ii)	$\forall x_1\forall x_2\forall x_3(x_1 < x_2 \to (x_2 < x_3 \to x_1 < x_3))$	(transitivitet)

Vi har då konstruerat teorin för partiell ordning.

Exempel 3: Ytterligare ett exempel på en teori är följande, där vi ger en axiomatisering av aritmetiken. Språket för denna teori innehåller en konstant c, som är tänkt att tolkas som talet noll. Det finns tre funktionsbokstäver f_1^1, f_1^2 och f_2^2, som är tänkta att tolkas som efterföljarfunktionen ($S(x) = x + 1$), addition respektive multiplikation. Vi har slutligen en predikatsymbol A_1^2, som är tänkt att tolkas som identitetssymbol. Av läsbarhetsskäl använder vi 0, S, +, $\cdot$ respektive = för ovannämnda symboler. Det är emellertid viktigt att inse att vi därmed indikerar tänkta tolkningar, och att symbolerna kan ha andra tolkningar. De äkta axiomen är följande, och vi avstår från att skriva ut allkvantifikatorerna.

(i) $x_1 = x_2 \rightarrow (x_1 = x_3 \rightarrow x_2 = x_3)$
(ii) $x_1 = x_2 \rightarrow S(x_1) = S(x_2)$
(iii) $0 \neq S(x_1)$
(iv) $S(x_1) = S(x_2) \rightarrow x_1 = x_2$
(v) $x_1 + 0 = x_1$
(vi) $x_1 + S(x_2) = S(x_1 + x_2)$
(vii) $x_1 \cdot 0 = 0$
(viii) $x_1 \cdot S(x_2) = (x_1 \cdot x_2) + x_1$
(ix) För varje välbildad formel $\varphi(x)$, så
$\varphi(0) \rightarrow (\forall x(\varphi(x) \rightarrow \varphi(S(x)) \rightarrow \forall x\varphi(x))$

Axiomen (i) till (viii) är specifika vbf, medan axiom (ix) är ett axiomschema. Detta schema är det starkaste möjliga sättet att formulera induktionsaxiomet i ett första ordningens språk. De två första axiomen ger vissa egenskaper hos $=$, axiomen (iii) och (iv) ger egenskaper hos S, axiomen (v) och (vi) ger en rekursiv definition av addition och axiomen (vii) och (viii) ger en rekursiv definition av multiplikation. Denna teori är mycket viktig och brukar betecknas Peanos aritmetik (PA). I den struktur som har de naturliga talen som domän, och som tolkar de ickelogiska symbolerna som de är tänkta enligt ovan, så gäller det att samtliga axiom i PA är sanna. Vi betecknar denna struktur med $\underline{N}$, och det gäller alltså att $\underline{N} \models PA$. Det gäller också att samtliga satser i $Th(PA)$ är sanna i denna struktur.

För att illustrera det mödosamma med att bevisa teorem i K tittar vi på följande exempel.

Exempel 4: Visa att $\vdash \neg\varphi \rightarrow (\varphi \rightarrow \psi)$, där φ och ψ är två vbf.

Lösning:

1	$\neg\varphi \rightarrow (\neg\psi \rightarrow \neg\varphi)$	Ax (i)
2	$(\neg\psi \rightarrow \neg\varphi) \rightarrow (\varphi \rightarrow \psi)$	Ax (iii)
3	$((\neg\psi \rightarrow \neg\varphi) \rightarrow (\varphi \rightarrow \psi)) \rightarrow$ $\rightarrow (\neg\varphi \rightarrow ((\neg\psi \rightarrow \neg\varphi) \rightarrow (\varphi \rightarrow \psi)))$	Ax (i)
4	$\neg\varphi \rightarrow ((\neg\psi \rightarrow \neg\varphi) \rightarrow (\varphi \rightarrow \psi))$	2, 3 MP
5	$(\neg\varphi \rightarrow ((\neg\psi \rightarrow \neg\varphi) \rightarrow (\varphi \rightarrow \psi))) \rightarrow$ $\rightarrow ((\neg\varphi \rightarrow (\neg\psi \rightarrow \neg\varphi)) \rightarrow (\neg\varphi \rightarrow (\varphi \rightarrow \psi)))$	Ax (ii)
6	$(\neg\varphi \rightarrow (\neg\psi \rightarrow \neg\varphi)) \rightarrow (\neg\varphi \rightarrow (\varphi \rightarrow \psi))$	4, 5 MP
7	$\neg\varphi \rightarrow (\varphi \rightarrow \psi)$	1, 6 MP

Ovanstående bevis hade varit avsevärt mycket enklare om vi hade haft tillgång till den härledda deduktionsregeln Hypotetisk Syllogism. Denna kan bevisas i K på följande sätt.

Exempel 5: Visa att $\{\varphi \rightarrow \psi, \psi \rightarrow \theta\} \vdash \varphi \rightarrow \theta$, där φ, ψ och θ är tre vbf.

Lösning:

1	$\varphi \rightarrow \psi$	Prem
2	$\psi \rightarrow \theta$	Prem
3	$(\varphi \rightarrow (\psi \rightarrow \theta)) \rightarrow ((\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \theta))$	Ax (ii)
4	$(\psi \rightarrow \theta) \rightarrow (\varphi \rightarrow (\psi \rightarrow \theta))$	Ax (i)
5	$\varphi \rightarrow (\psi \rightarrow \theta)$	2, 4 MP
6	$(\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \theta)$	3, 5 MP
7	$\varphi \rightarrow \theta$	1, 6 MP

Med tillgång till denna deduktionsregel, blir härledningen i ex 4 avsevärt enklare. Vi utnyttjar den helt enkelt på formlerna på rad 1 och 2 för att direkt erhålla det önskade.

Genom att lägga till härledda deduktionsregler till K, och genom att flitigt utnyttja förkortningar, kan man få K att bli ett smidigt system att arbeta i.

Deduktionsteoremet gäller i K, och vi klarar oss med nedanstående svagare variant av detta teorem.

Deduktionsteoremet: Om φ är en sats och Γ en mängd av premisser, så gäller $\Gamma + \varphi \vdash \psi$ om och endast om $\Gamma \vdash \varphi \to \psi$.

Exempel 6: Visa att $\vdash \neg\neg\varphi \to \varphi$, där vi förutsätter att φ är en sats.

Lösning:

1	$\neg\neg\varphi$	Prem
2	$\neg\neg\varphi \to (\neg\neg\neg\neg\varphi \to \neg\neg\varphi)$	Ax (i)
3	$\neg\neg\neg\neg\varphi \to \neg\neg\varphi$	1, 2 MP
4	$(\neg\neg\neg\neg\varphi \to \neg\neg\varphi) \to (\neg\varphi \to \neg\neg\neg\varphi)$	Ax (iii)
5	$\neg\varphi \to \neg\neg\neg\varphi$	3, 4 MP
6	$(\neg\varphi \to \neg\neg\neg\varphi) \to (\neg\neg\varphi \to \varphi)$	Ax (iii)
7	$\neg\neg\varphi \to \varphi$	5, 6 MP
8	φ	1, 7 MP

Det är då klart att $\neg\neg\varphi \vdash \varphi$, och deduktionsteoremet ger att $\vdash \neg\neg\varphi \to \varphi$.

Exempel 7: Visa att $\vdash (\varphi \to \neg\varphi) \to \neg\varphi$, där vi förutsätter att φ är en sats.

Lösning:

1	$\varphi \to \neg\varphi$	Prem
2	$\neg\neg\varphi \to (\neg\neg(\varphi \to \neg\varphi) \to \neg\neg\varphi)$	Ax (i)
3	$(\neg\neg(\varphi \to \neg\varphi) \to \neg\neg\varphi)) \to (\neg\varphi \to \neg(\varphi \to \neg\varphi))$	Ax (iii)
4	$\neg\neg\varphi \to (\neg\varphi \to \neg(\varphi \to \neg\varphi))$	2, 3 HS
5	$(\neg\neg\varphi \to (\neg\varphi \to \neg(\varphi \to \neg\varphi))) \to$ $\to ((\neg\neg\varphi \to \neg\varphi) \to (\neg\neg\varphi \to \neg(\varphi \to \neg\varphi)))$	Ax (ii)
6	$(\neg\neg\varphi \to \neg\varphi) \to (\neg\neg\varphi \to \neg(\varphi \to \neg\varphi))$	4, 5 MP
7	$\neg\neg\varphi \to \varphi$	Ex 6
8	$\neg\neg\varphi \to \neg\varphi$	1, 7 HS
9	$\neg\neg\varphi \to \neg(\varphi \to \neg\varphi)$	6, 8 MP
10	$(\neg\neg\varphi \to \neg(\varphi \to \neg\varphi)) \to ((\varphi \to \neg\varphi)\to \neg\varphi)$	Ax (iii)
11	$(\varphi \to \neg\varphi)\to \neg\varphi$	9, 10 MP
12	$\neg\varphi$	1, 11 MP

Det är då klart att $\varphi \to \neg\varphi \vdash \neg\varphi$. Eftersom φ är en sats, så är $\varphi \to \neg\varphi$ också en sats, och deduktionsteoremet ger då att $\vdash (\varphi \to \neg\varphi) \to \neg\varphi$.

Några metateorem

Låt $\psi(v_1, \dots, v_n)$ vara en vbf där de fria variablerna är precis de angivna. Då kallas satsen $\forall v_1 \dots \forall v_n\psi(v_1, \dots, v_n)$ för (det slutna) *höljet* till ψ. Observera att för godtycklig struktur $\underline{M}$ och godtycklig formel ψ så gäller att $\underline{M} \models \psi(v_1, \dots, v_n)$ om och endast om höljet till $\psi(v_1, \dots, v_n)$ är sann i $\underline{M}$, dvs om och endast om $\underline{M} \models \forall v_1 \dots \forall v_n\psi(v_1, \dots, v_n)$. Eftersom vi låter $\psi(v_1, \dots, v_n)$ vara en förkortning av $\forall v_1 \dots \forall v_n\psi(v_1, \dots, v_n)$, gäller det dessutom att $\Gamma \vdash \psi(v_1, \dots, v_n)$ om och endast om $\Gamma \vdash \forall v_1 \dots \forall v_n\psi(v_1, \dots, v_n)$. Detta innebär att vi ofta kan välja att diskutera satser i stället för godtyckliga vbf i bevisen nedan.

Teorem 1: Den predikatlogiska kalkylen K är sund, dvs för varje sats φ så gäller att om $\vdash \varphi$ så är $\models \varphi$, dvs alla teorem är logiskt sanna.

Bevis: Vi har att bevisa

(i) att varje axiom är en predikatlogisk sanning, och
(ii) att MP och Gen bevarar predikatlogisk sanning, dvs om premisserna är predikatlogiskt sanna, så är slutsatsen logiskt sann.

(i) Vi illustrerar med axiom (iv) och överlåter verifieringen av övriga axiom till läsaren.
Vi har att visa att $\models \forall x_i\varphi(x_i) \rightarrow \varphi(t)$, där t är en sluten term, dvs variabelfri.
Vi antar att de fria variablerna i $\forall x_i\varphi(x_i) \rightarrow \varphi(t)$ finns bland $x_1, x_2, \ldots, x_n$ där vi också antar att $n < i$. Om så inte är fallet, kan vi byta *bundna* variabler så att dessa villkor är uppfyllda och att inga bundna variabler finns bland $x_1, x_2, \ldots, x_n$. Vi har då att visa att

$\models \forall x_1\forall x_2 \ldots \forall x_n(\forall x_i\varphi(x_1, x_2, \ldots, x_n, x_i) \rightarrow \varphi(x_1, x_2, \ldots, x_n, t))$.

Antag för motsägelse att det finns en namnfullständig struktur $\underline{M}$ sådan att

$\underline{M} \not\models \forall x_1\forall x_2 \ldots \forall x_n(\forall x_i\varphi(x_1, x_2, \ldots, x_n, x_i) \rightarrow \varphi(x_1, x_2, \ldots, x_n, t))$.

Då finns element $a_1, a_2, \ldots, a_n \in M$ så att

$\underline{M} \not\models \forall x_i\varphi(c_{a_1}, c_{a_2}, \ldots, c_{a_n}, x_i) \rightarrow \varphi(c_{a_1}, c_{a_2}, \ldots, c_{a_n}, t))$,

och då har vi att

(1) $\underline{M} \models \forall x_i\varphi(c_{a_1}, c_{a_2}, \ldots, c_{a_n}, x_i)$ och
(2) $\underline{M} \not\models \varphi(c_{a_1}, c_{a_2}, \ldots, c_{a_n}, t))$.

(1) ger att

$\underline{M} \models \varphi(c_{a_1}, c_{a_2}, \ldots, c_{a_n}, t))$, som motsäger (2), och då är det klart att $\models \forall x_i\varphi(x_i) \rightarrow \varphi(t)$.

(ii) Vi visar att MP bevarar logisk sanning.

Antag att $\models \varphi$ och $\models \varphi \rightarrow \psi$, och för motsägelse att $\not\models \psi$. Då finns en struktur $\underline{M}$ så att $\underline{M} \not\models \psi$. Men det gäller också att $\underline{M} \models \varphi$ och $\underline{M} \models \varphi \rightarrow \psi$, som ger att $\underline{M} \models \psi$ och vi har en motsägelse. Därmed är det klart att MP bevarar predikatlogisk sanning.

Eftersom det då gäller att varje axiom är predikatlogiskt sant och att MP och Gen bevarar logisk sanning, så måste varje teorem vara predikatlogiskt sant.

Ovanstående är den ”lätta” delen av Gödels fullständighetssats.

Definition 5: En teori S är konsistent om och endast om det inte finns någon sats φ sådan att både $S \vdash \varphi$ och $S \vdash \neg\varphi$.

Teorem 2: K är konsistent.

Bevis*:* Antag för motsägelse att K inte är konsistent. Då finns en sats φ så att $\vdash \varphi$ och $\vdash \neg\varphi$. Sundheten ger då att $\models \varphi$ och $\models \neg\varphi$, och vi har en motsägelse. Alltså är K konsistent.

Följande teorem ger en karakterisering av konsistenta teorier, men vi behöver först ytterligare en definition.

Definition 6: En teori S är en *extension* (*utvidgning*) av en teori T om och endast om det för varje φ gäller att om $T \vdash \varphi$, så $S \vdash \varphi$. Vi skriver $T \dashv S$, om S är en extension av T, och säger även att T är en *delteori* till S.

Observera att K är delteori till varje teori S.

Teorem 3: En extension S till K är konsistent om och endast om det finns en sats som inte är bevisbar i S.

Bevis: Observera att $K \dashv S$. Vi använder i båda riktningarna nedan den kontrapositiva formuleringen. Kontrapositionen till formeln $\varphi \to \psi$ är formeln $\neg\psi \to \neg\varphi$.

$\Rightarrow$/ Antag att alla satser är bevisbara i S. Vi ska visa att S är inkonsistent. Låt φ vara en godtycklig sats. Då gäller $S \vdash \varphi$ och $S \vdash \neg\varphi$, och därmed är det klart att S är inkonsistent.

$\Leftarrow$/ Antag att S är inkonsistent. Då finns φ så att $S \vdash \varphi$ och $S \vdash \neg\varphi$. Låt ψ vara en godtycklig sats. Eftersom $\vdash \neg\varphi \to (\varphi \to \psi)$ enligt exempel 4, och $K \dashv S$, så har vi att $S \vdash \neg\varphi \to (\varphi \to \psi)$. Då ger två tillämpningar av MP att $S \vdash \psi$, och därmed är det klart.

Vi ska nu påbörja beviset av den svåra delen av Gödels fullständighetssats, och inleder med några förberedande hjälpsatser.

Lemma 1: Antag att S är en konsistent extension av K, och att satsen $\neg\varphi$ inte är teorem i S. Då är teorin $S' = S + \varphi$ konsistent.

Bevis: Antag för motsägelse att S' är inkonsistent. Då finns ψ så att $S' \vdash \psi$ och $S' \vdash \neg\psi$. Eftersom $K \dashv S \dashv S'$, och $\vdash \neg\psi \to (\psi \to \neg\varphi)$, så $S' \vdash \neg\psi \to (\psi \to \neg\varphi)$. Då ger MP två gånger att $S' \vdash \neg\varphi$.
Eftersom $S' = S + \varphi$, så har vi att $S + \varphi \vdash \neg\varphi$, och deduktionsteoremet ger att $S \vdash \varphi \to \neg\varphi$. Det gäller också enligt exempel 7 att $\vdash (\varphi \to \neg\varphi) \to \neg\varphi$, varför $S \vdash (\varphi \to \neg\varphi) \to \neg\varphi$. Vi får då att $S \vdash \neg\varphi$, som motsäger förutsättningen att $\neg\varphi$ inte är bevisbar i S. Det är då klart att $S + \varphi$ är konsistent.

Definition 7: En teori S är *fullständig* om och endast om det för varje sats φ gäller att $S \vdash \varphi$ eller $S \vdash \neg\varphi$. Om S dessutom är konsistent, sägs S vara *maximalkonsistent*.

Innan vi går vidare med nästa lemma, skall vi diskutera kardinalitet. Om detta kan man också läsa t ex i Rosen kap 1.7. Två mängder A och B har *samma kardinalitet* om och endast om det finns en bijektion mellan A och B. Vi skriver $|A| = |B|$, om A och B har samma kardinalitet. Uttrycket $|A|$ läser vi kardinaliteten för mängden A. Om A är ändlig och innehåller n element, så är $|A| = n$. Det minsta oändliga kardinaltalet betecknas $\aleph_0$ (alef-noll), och är kardinaltalet för **N**, dvs $|\mathbf{N}| = \aleph_0$. En mängd med kardinalitet $\aleph_0$ är *uppräknelig*. Eftersom det finns en bijektion mellan **N** och en uppräknelig mängd, så kan denna alltid räknas upp i ett första, ett andra , ett tredje osv element. En mängd med kardinalitet större än $\aleph_0$ sägs vara *överuppräknelig*. Dylika mängder kan inte räknas upp.

Exempel 8: **Z** = {heltal} och **Q** = {rationella tal} är uppräkneliga medan P(**N**) = potensmängden till **N** och **R** = {reella tal} är överuppräkneliga. (Se [Rosen] kap 1.7.)

Exempel 9: Visa att mängden ändliga strängar över en uppräknelig vokabulär är uppräknelig.

Lösning: Låt vokabulären vara $V = \{v_1, v_2, v_3, \ldots\}$. Observera att V är uppräknelig. Vi associerar sedan ett tal med varje symbol i vokabulären enligt

$$g(v_k) = 2k + 1.$$

Varje element i vokabulären är då associerat med ett udda tal.
Därefter associerar vi varje ändlig sträng $u = u_1u_2 \ldots u_n$, där $u_i \in V$, med talet

$$2^{g(u_1)} \cdot 3^{g(u_2)} \cdot 5^{g(u_3)} \cdot \ldots \cdot p_n^{g(u_n)},$$

där p_i är det i:te primtalet och $p_1 = 2$. Tomma strängen associeras med talet 0. Varje ändlig sträng är då associerat med ett unikt jämnt naturligt tal, och vi kan räkna upp strängarna i ”storleksordning” med de associerade talen.
Vi kan också effektivt avgöra, via aritmetikens fundamentalsats (Rosen kap 2.3), om ett givet tal är associerat med en sträng och i så fall med vilken sträng.
Observera att om vi låter vokabulären vara predikatlogikens, så har vi nu visat att mängden termer, mängden satser etc är uppräkneliga.

Lemma 2: (Lindenbaums lemma) Om S är en konsistent teori, så har S en maximalkonsistent extension.

Bevis: Låt $\varphi_1, \varphi_2, \varphi_3, \ldots$ vara en uppräkning av samtliga satser i S:s språk. En dylik finns enligt ovanstående exempel. Vi bildar sedan en följd av extensioner $J_1, J_2, J_3, \ldots$ till S på följande sätt.

Låt $J_1 = S$

Låt $J_{n+1} = \begin{cases} J_n + \varphi_n, & \text{om } J_n \text{ inte bevisar } \neg\varphi_n \\ J_n, & \text{om } J_n \text{ bevisar } \neg\varphi_n \end{cases}$ för $n \geq 1$.

Påstående 1: J_i är konsistent för alla i.

Vi visar detta med induktion, där grundsteget redan är klart, eftersom $J_1 = S$ är konsistent enligt förutsättning.
Antag för induktion att J_n är konsistent. Vi har att visa att J_{n+1} är konsistent.
Om det inte är så att $J_n \vdash \neg\varphi_n$, så är $J_{n+1} = J_n + \varphi_n$ konsistent enligt lemma 1.
Om $J_n \vdash \neg\varphi_n$, så är $J_{n+1} = J_n$, och J_{n+1} är konsistent enligt induktions-antagandet.
Induktionsaxiomet ger då att J_i är konsistent för alla i. Därmed är påstående 1 bevisat.

Vi bildar sedan $J = \bigcup_{i=1}^{\infty} J_i$. Observera att $S \dashv J_1 \dashv \ldots \dashv J_n \dashv \ldots \dashv J$.

Påstående 2: J är konsistent.

Antag för motsägelse att J inte är konsistent. Då finns en sats ψ sådan att $J \vdash \psi$ och $J \vdash \neg\psi$. Eftersom alla bevis är ändliga, så används högst ett ändligt antal axiom från J i dessa båda bevis. Då finns ett n så att samtliga axiom som används i dessa två bevis redan ingår i J_n. Då kan bevisen för ψ och $\neg\psi$ genomföras i J_n, dvs $J_n \vdash \psi$ och $J_n \vdash \neg\psi$. Men detta motsäger att J_n är konsistent. Alltså är J konsistent.

Påstående 3: J är fullständig.

Vi har då att visa att det för godtycklig sats ψ gäller att $J \vdash \psi$ eller $J \vdash \neg\psi$.
Låt därför ψ vara en godtycklig sats. Då är
$\psi = \varphi_n$ för något n i den inledande uppräkningen.
Om det inte är så att $J_n \vdash \neg\varphi_n$, så är $J_{n+1} = J_n + \varphi_n$, och därmed gäller att $J_{n+1} \vdash \varphi_n$, som ger att $J \vdash \varphi_n$, varför $J \vdash \psi$ eller $J \vdash \neg\psi$.
Om $J_n \vdash \neg\varphi_n$, så gäller förstås att $J \vdash \psi$ eller $J \vdash \neg\psi$.
Alltså är J fullständig.

Därmed är det klart att J är en maximalkonsistent extension av S.

Inför nästa lemma behöver vi ytterligare en definition. Observera att en term är sluten om den inte innehåller några variabler.

Definition 8: En teori S är *existensfullständig* om och endast om det för varje välbildad formel $\varphi(x)$ med exakt en fri variabel finns en sluten term t sådan att $S \vdash \exists x \neg\varphi(x) \rightarrow \neg\varphi(t)$.

I det följande kommer vi ibland att använda oss av definierade logiska symboler, som formellt inte ingår i K, och av härledda deduktionsregler, som vi inte bevisat, men vi vet i princip hur det skall gå till, och eftersom vi vet hur ett system för naturlig deduktion fungerar och eftersom vi vet hur vi kan utvidga K till ett dylikt, så tar vi oss dessa friheter.

Lemma 3: Varje konsistent teori S har en konsistent, existensfullständig extension T sådan att T:s språk innehåller uppräkneligt många slutna termer.

Bevis: Utöka språket för S med en uppräknelig mängd $\{b_1, b_2, b_3, \ldots\}$ av *nya* konstanter. Kalla denna nya teori för S_0. Axiomen i S_0 är de i S, samt nya logiska axiom som innehåller de nya konstanterna.

Påstående 1: S_0 är konsistent.

Antag för motsägelse att S_0 inte är konsistent. Då finns φ så att $S_0 \vdash \varphi \wedge \neg\varphi$ för någon sats φ. Ersätt varje konstant b_i, som förekommer i beviset av $\varphi \wedge \neg\varphi$ i S_0, med en konstant från K:s vokabulär och som inte förekommer i beviset. Dessa substitutioner bevarar axiom och korrektheten av användning av deduktionsregler. Vi har därför ett bevis i S av $\varphi' \wedge \neg\varphi'$, där φ' är den vbf som dykt upp efter substitutionerna. Eftersom S är konsistent enligt förutsättning, så har vi en motsägelse. Alltså är S_0 konsistent.

Låt nu $\varphi_1(x_{i_1}), \varphi_2(x_{i_2}), \ldots$ vara en uppräkning av alla vbf med exakt en fri variabel. Välj en sekvens $b_{j_1}, b_{j_2}, \ldots$ bland de nya konstanterna så att b_{j_k} inte ingår i någon av formlerna $\varphi_1(x_{i_1}), \ldots, \varphi_k(x_{i_k})$, och så att $b_{j_k} \neq b_{j_i}$ för $i < k$. Betrakta formeln

$$F_k: \quad \exists x_{i_k} \neg\varphi_k(x_{i_k}) \to \neg\varphi_k(b_{j_k}), \text{ för } k \geq 1.$$

Bilda nu en svit av teorier

$$S_n = S_{n-1} + F_n, \text{ för } n \geq 1,$$

och låt $T = \bigcup_{n=0}^{\infty} S_n$.

Uppenbart är T existensfullständig, och det återstår att visa att T är konsistent. Vi visar först att S_n är konsistent för alla n.

Påstående 2: S_n är konsistent för alla n.

Grundsteget, S_0 konsistent, är redan klart.

Antag för induktion att S_{n-1} är konsistent, och för motsägelse att S_n är inkonsistent. Då gäller, enligt karakteriseringen av konsistenta teorier i teorem 3, att S_n bevisar alla formler. Det gäller alltså att $S_n \vdash \neg F_n$. Då gäller $S_{n-1} + F_n \vdash \neg F_n$, och med deduktionsteoremet att $S_{n-1} \vdash F_n \to \neg F_n$. Tillsammans med exempel 5 ger detta att $S_{n-1} \vdash \neg F_n$, dvs

$$S_{n-1} \vdash \neg(\exists x_{i_n} \neg\varphi_n(x_{i_n}) \to \neg\varphi_n(b_{j_n})).$$

I fortsättningen av beviset använder vi nu en kraftfullare deduktionsapparat än vad K har att erbjuda. Vi får att

$$(+) \qquad S_{n-1} \vdash \exists x_{i_n} \neg\varphi_n(x_{i_n}), \text{ och}$$
$$S_{n-1} \vdash \varphi_n(b_{j_n}).$$

Eftersom b_{j_n} inte förekommer i $F_1, \ldots, F_{n-1}$, innebär detta att Gen ger att

$$S_{n-1} \vdash \forall x_p \varphi_n(x_p), \text{ dvs}$$
$$S_{n-1} \vdash \neg\exists x_p \neg\varphi_n(x_p), \text{ som motsäger (+).}$$

Då är det klart att S_n är konsistent, och induktionsaxiomet ger då att S_n är konsistent för alla n.

Slutligen får vi på samma sätt som i beviset för Lindenbaums lemma att

$$T = \bigcup_{n=0}^{\infty} S_n$$

är konsistent.

I det lemma som skall bevisas nedan används en viktig teknik för att skapa strukturer. Idén är att låta de slutna termerna i ett språk vara element i tolkningens domän. Vi får då en så kallad *termmodell*.

Lemma 4: Låt J vara en maximalkonsistent, existensfullständig teori. Då finns en struktur $\underline{M}$ sådan att $\underline{M} \models J$ och $M = \{t : t$ är en sluten term i J:s språk$\}$.

Bevis: För varje konstant a_i, låter vi $a_i^{\underline{M}} = a_i$. För varje funktionssymbol f_k^n, låter vi för varje uppsättning slutna termer $t_1, \dots, t_n$,

$$f_k^n(t_1, \dots, t_n)^{\underline{M}} = f_k^n(t_1, \dots, t_n).$$

Slutligen låter vi, för varje predikatsymbol A_k^n och varje uppsättning slutna termer $t_1, \dots, t_n$,

$$(A_k^n)^{\underline{M}} = \{(t_1, \dots, t_n) : J \vdash A_k^n(t_1, \dots, t_n)\}, \text{ dvs}$$

$$(t_1, \dots, t_n) \in (A_k^n)^{\underline{M}} \text{ om och endast om } J \vdash A_k^n(t_1, \dots, t_n).$$

Det är nu tillräckligt att visa att

(+) $\underline{M} \models \varphi$ om och endast om $J \vdash \varphi$ för varje sats φ.

Beviset av (+) sker med induktion över antalet konnektiver och kvantifikatorer i satsan φ.

Grundsteg: φ innehåller varken konnektiv eller kvantifikatorer. Då är φ på formen $A_k^n(t_1, \dots, t_n)$, där termerna är variabelfria. Då följer (+) direkt av definitionen $(A_k^n)^{\underline{M}}$ då den ger att $\underline{M} \models \varphi$ om och endast om $\underline{M} \models A_k^n(t_1, \dots, t_n)$ om och endast om $(t_1, \dots, t_n) \in (A_k^n)^{\underline{M}}$ om och endast om $J \vdash A_k^n(t_1, \dots, t_n)$ om och endast om $J \vdash \varphi$.

Induktionssteg: Antag för induktion att (+) gäller för alla satser med färre än r konnektiv och kvantifikatorer. Vi ska visa att (+) gäller för alla satser med r konnektiv (jämför exempel 2:20).

Fall 1: φ är $\neg\psi$

Om $\underline{M} \models \varphi$, så $\underline{M} \not\models \psi$, och induktionsantagandet ger att J inte bevisar ψ. Eftersom J är fullständig och ψ en sats, så har vi att $J \vdash \neg\psi$, dvs $J \vdash \varphi$. Alltså gäller det att om $\underline{M} \models \varphi$, så $J \vdash \varphi$.

Om $J \vdash \varphi$, dvs $J \vdash \neg\psi$, så gäller, eftersom J är konsistent att J inte bevisar ψ. Induktionsantagandet ger då att $\underline{M} \not\models \psi$, varför $\underline{M} \models \varphi$. Alltså gäller det att om $J \vdash \varphi$, så $\underline{M} \models \varphi$.

Alltså gäller det att $\underline{M} \models \varphi$ om och endast om $J \vdash \varphi$.

Fall 2: φ är $\psi \to \theta$

Eftersom φ är en sats, måste även ψ och θ vara satser.

Om $\underline{M} \not\models \varphi$, så $\underline{M} \models \psi$ och $\underline{M} \not\models \theta$. Då ger induktionsantagandet att $J \vdash \psi$ och att J inte bevisar θ, och fullständigheten hos J ger att $J \vdash \neg\theta$. Vi får $J \vdash \psi \wedge \neg\theta$, där $\psi \wedge \neg\theta$ är en förkortning av $\neg(\psi \to \theta)$, dvs $J \vdash \neg(\psi \to \theta)$. Då ger J:s konsistens att J inte bevisar $\psi \to \theta$. Därmed är det klart att

$$J \vdash \psi \to \theta \text{ medför att } \underline{M} \models \psi \to \theta, \text{ dvs}$$

$J \vdash \varphi$ medför att $\underline{M} \models \varphi$.

Om J inte bevisar φ, så ger fullständigheten att $J \vdash \neg(\psi \rightarrow \theta)$, dvs $J \vdash \psi$ och $J \vdash \neg\theta$.. Konsistensen ger att J inte bevisar θ. Induktionsantagandet ger att $\underline{M} \models \psi$ och $\underline{M} \not\models \theta$, dvs $\underline{M} \not\models \psi \rightarrow \theta$ och vi har att $\underline{M} \not\models \varphi$. Det är då klart att $\underline{M} \models \varphi$ medför att $J \vdash \varphi$.

Alltså gäller det att $\underline{M} \models \varphi$ om och endast om $J \vdash \varphi$.

Fall 3: φ är $\forall x_m \psi$

a/ Om ψ är en sats så gäller

$\underline{M} \models \varphi$ om och endast om $\underline{M} \models \psi$ om och endast om $J \vdash \psi$ om och endast om $J \vdash \varphi$.

b/ Om ψ inte är en sats så innehåller ψ exakt en fri variabel x_m, så ψ är $\theta(x_m)$, och φ är $\forall x_m \theta(x_m)$.

Antag nu att $\underline{M} \models \varphi$, och antag för motsägelse att J inte bevisar φ. På grund av fullständigheten, så $J \vdash \neg\varphi$, dvs $J \vdash \neg\forall x_m \theta(x_m)$, som ger att $J \vdash \exists x_m \neg \theta(x_m)$. Eftersom J är existensfullständig, så $J \vdash \neg \theta(t)$ för någon sluten term t. Eftersom $\underline{M} \models \forall x_m \theta(x_m)$, och $\models \forall x_m \theta(x_m) \rightarrow \theta(t)$, gäller det att $\underline{M} \models \theta(t)$, och induktionsantagandet ger $J \vdash \theta(t)$. Detta motsäger J:s konsistens.
Alltså har vi att $\underline{M} \models \varphi$ medför att $J \vdash \varphi$.

Antag sedan att $J \vdash \varphi$ och för motsägelse att $\underline{M} \not\models \varphi$, dvs
(1) $J \vdash \forall x_m \theta(x_m)$ och (2) $\underline{M} \not\models \forall x_m \theta(x_m)$.
(2) ger att $\underline{M} \not\models \theta(t)$ för någon term $t \in M$, och (1) ger att $J \vdash \theta(t)$, eftersom $\vdash \forall x_m \theta(x_m) \rightarrow \theta(t)$ för en sluten term t. Induktionsantagandet ger då att $\underline{M} \models \theta(t)$, och vi har en motsägelse.
Alltså har vi att $J \vdash \varphi$ medför att $\underline{M} \models \varphi$., och därmed är det klart att $\underline{M} \models \varphi$ om och endast om $J \vdash \varphi$.

Därmed är induktionen klar.

Vi är nu klara att bevisa följande centrala teorem. Observera att en struktur är uppräknelig, om den har en uppräknelig domän.

Teorem 4: För varje konsistent teori S, finns en uppräknelig struktur $\underline{M}$ sådan att $\underline{M} \models S$.

Bevis: Låt S vara en konsistent teori. Då finns en konsistent, existensfullständig extension S' av S sådan att språket för S' innehåller uppräkneligt många slutna termer (lemma 3). Enligt lemma 2 så har S' en maximalkonsistent extension T sådan att T och S' har samma språk. Enligt lemma 4 finns en struktur $\underline{M}$ sådan att $\underline{M} \models T$. Eftersom $S \dashv S' \dashv T$, gäller då att $\underline{M} \models S$.

Korollarium 1: Om $\models \varphi$, så $\vdash \varphi$.

Bevis: Antag att φ är en predikatlogiskt sann sats, och antag för motsägelse att inte $\vdash \varphi$. Då är $K + \neg\varphi$ konsistent enligt lemma 1. Enligt teorem 4, finns en uppräknelig struktur $\underline{M}$ sådan att $\underline{M} \models K + \neg\varphi$. Då gäller $\underline{M} \models \neg\varphi$, och eftersom $\models \varphi$, så gäller även $\underline{M} \models \varphi$, dvs satsen φ är både sann och falsk i $\underline{M}$. Vi har en motsägelse, och därmed är det klart att $\vdash \varphi$.

Korollarium 2: (Gödels fullständighetssats) $\models \varphi$ om och endast om $\vdash \varphi$.

Bevis: Följer av teorem 1 och korollarium 1.

Korollarium 3: Låt φ vara en sats och Γ en mängd av satser i K:s vokabulär. Då gäller $\Gamma \models \varphi$ om och endast om $\Gamma \vdash \varphi$.

Bevis: $\Rightarrow$/ Antag att $\Gamma \models \varphi$ och för motsägelse att det inte är så att $\Gamma \vdash \varphi$.

Då är $\Gamma + \neg\varphi$ konsistent, och teorem 4 ger att det finns en struktur $\underline{M}$ sådan att $\underline{M} \models \Gamma + \neg\varphi$, dvs $\underline{M} \models \Gamma$ och $\underline{M} \models \neg\varphi$.

Men detta motsäger att $\Gamma \models \varphi$, som ju säger att för varje struktur sådan att $\underline{M} \models \Gamma$, så gäller $\underline{M} \models \varphi$.

Alltså gäller det att $\Gamma \models \varphi$ medför $\Gamma \vdash \varphi$.

$\Leftarrow$/ Antag $\Gamma \vdash \varphi$, och låt $\underline{M}$ vara en godtycklig struktur sådan att $\underline{M} \models \Gamma$. Vi ska visa att $\underline{M} \models \varphi$.

I beviset av φ från Γ utnyttjas högst ett ändligt antal av premisserna i Γ, säg att dessa är $\psi_1, \psi_2, \dots, \psi_n$.
Då gäller det att $\{\psi_1, \psi_2, \dots, \psi_n\} \vdash \varphi$, som ger att $\vdash (\psi_1 \wedge \psi_2 \wedge \dots \wedge \psi_n) \rightarrow \varphi$.
Teorem 1 (sundheten) ger då att

$\models (\psi_1 \wedge \psi_2 \wedge \dots \wedge \psi_n) \rightarrow \varphi$, dvs $\{\psi_1, \psi_2, \dots, \psi_n\} \models \varphi$.

Eftersom $\underline{M} \models \Gamma$, så måste $\underline{M} \models \{\psi_1, \psi_2, \dots, \psi_n\}$ och därmed gäller det att $\underline{M} \models \varphi$.

Det är då klart att $\Gamma \vdash \varphi$ medför $\Gamma \models \varphi$.

Definition 9: En teori S är *avgörbar* om och endast om det finns en mekanisk metod med hjälp av vilken man kan avgöra huruvida en given sats är teorem i S eller ej.

Observera att detta inte är en särskilt precis definition eftersom begreppet *mekanisk metod* är vagt och inte definierat. Man kan emellertid precisera detta begrepp, t ex med hjälp av begreppet *turingmaskin* och på så sätt erhålla en precis definition av begreppet avgörbarhet.

Teorem 5: Det formella systemet K är ej avgörbart.

Beviset av detta resultat ligger ej inom ramen för vad vi kan åstadkomma i detta kompendium.

Den första ordningens predikatlogik vi definierat i detta avsnitt kan utvidgas till en första ordningens predikatlogik med identitet. Vi reserverar en av predikatsymbolerna, säg A_1^2, för det vi tänkt som likhetstecken och vi utökar mängden axiom med ett axiom och ett axiomschema, som är tänkta att karakterisera identitet. De axiom vi lägger till är

(vi) $\forall x_1 A_1^2(x_1, x_1)$

(vii) $\forall x_1 \forall x_2 (x_1 = x_2 \rightarrow (\varphi(x_1, x_1) \rightarrow \varphi(x_1, x_2))$

I (vii) är $\varphi(x_1, x_1)$ är en godtycklig vbf, $\varphi(x_1, x_2)$ uppstår ur $\varphi(x_1, x_1)$ genom att ersätta en eller flera, dock inte nödvändigtvis alla, fria förekomster av x_1 med x_2 under förutsättning att x_2 är fri för x_1 i $\varphi(x_1, x_1)$.

Även i detta fall gäller lemma 4, men i beviset krävs en annan konstruktion av en lämplig struktur.

Vi avslutar med ytterligare en tillämpning av fullständighetssatsen. Den följande satsen har viktiga tillämpningar vad gäller konstruktioner av så kallade icke-standardmodeller.

Teorem 6: Låt S vara en teori. Då gäller det att det finns en struktur $\underline{M}$ sådan att $\underline{M} \models S$ om och endast om det för varje ändlig delmängd T till S finns en struktur $\underline{M}'$ sådan att $\underline{M}' \models T$.

Bevis: $\Rightarrow$/ Om det finns en struktur $\underline{M}$ sådan att $\underline{M} \models S$, så måste $\underline{M}$ vara sådan att $\underline{M} \models T$ för varje T sådant att $T \subseteq S$.

$\Leftarrow$/ Antag att det för varje ändlig delmängd T till S finns en struktur $\underline{M}'$ sådan att $\underline{M}' \models T$.

Vi ska visa att det finns en struktur $\underline{M}$ sådan att $\underline{M} \models S$. Antag för motsägelse att det inte finns en struktur $\underline{M}$ sådan att $\underline{M} \models S$.

Då ger teorem 4 att S är inkonsistent. Därmed finns en sats φ så att $S \vdash \varphi$ och $S \vdash \neg\varphi$. Eftersom alla bevis är ändliga, så finns en ändlig delmängd T till S sådan att T innehåller samtliga axiom som utnyttjas i de båda bevisen.

Då gäller det att $T \vdash \varphi$ och $T \vdash \neg\varphi$. Fullständighetssatsen (kor 3) ger att $T \models \varphi$ och $T \models \neg\varphi$. Eftersom T är en ändlig delmängd till S, så gäller enligt förutsättning att det finns en struktur $\underline{M}'$ sådan att $\underline{M}' \models T$. I denna struktur gäller då att $\underline{M}' \models \varphi$ och $\underline{M}' \models \neg\varphi$.

Vi har en motsägelse, och därmed är det klart att det finns en struktur $\underline{M}$ sådan att $\underline{M} \models S$.

Exempel 10: Detta exempel är knutet till exempel 3. Utöka det språk L som är givet i exempel 3 med en ny konstant c_1, och bilda teorin

$$T = Th(PA) \cup \{0 < c_1, S(0) < c_1, S(S(0)) < c_1, \ldots\}$$

dvs T har som axiom alla teorem i PA utökat med alla satser på formen $\underline{n} < c_1$. Vi använder $x < y$ som en förkortning av $\exists z(z \neq 0 \wedge x + z = y)$, och $\underline{n}$ är en förkortning av $S(S(\ldots(S(0))\ldots))$ där vi har n st användningar av S.

Då gäller det att $PA \dashv T$, och att till varje ändlig delmängd U till T finns en struktur $\underline{M}'$ sådan att $\underline{M}' \models U$.

Kompakthetssatsen ger då att det finns en struktur $\underline{M}$ sådan att $\underline{M} \models T$. Då gäller det att $c_1^{\underline{M}}$ inte är ett naturligt tal.

Eftersom $\underline{M}$ är en struktur för $L \cup \{c_1\}$, så finns en struktur för L där den enda ändring vi behöver göra i $\underline{M}$ är att eliminera tolkningen av konstanten c_1.

Beteckna denna struktur med $\underline{M}''$. Då gäller att $\underline{M}$ och $\underline{M}''$ har samma domän!

Vi har alltså konstruerat en struktur sådan att samtliga axiom i PA är sanna, och där domänen innehåller element som inte är naturliga tal. Strukturen kallas en *ickestandardmodell* till PA, och de element i domänen som inte är naturliga tal kallas *ickestandardtal*.

Litteratur

Barwise J, Etchemendy J	1990	*The Language of first-order logic*, CSLI lecture notes
Barwise J, Etchemendy J	1996	*Tarski´s world*, CSLI lecture notes
Bennet C m fl	1998	*En introduktion till första ordningens språk*, Inst för filosofi, Göteborgs universitet
Boolos G S, Jeffrey RC	1989	*Computability and Logic* (3:e uppl), Cambridge University Press
Drake F R, Singh D	1996	*Intermediate Set Theory*, Wiley
Galton A	1990	*Logic For Information Technology*, Wiley
Goldblatt R	1992	*Logics of Time and Computation* (2:a uppl), CSLI lecture notes
Hansen K B	1997	*Grundläggande logik* (3:e uppl), Studentlitteratur
Hofstadter	1985	*Gödel, Escher, Bach*, Bromberg
Hopcroft J E, Ullman J D	1979	*Introduction to Automata Theory, Languages and Computation*, Addison Wesley
Mates B	1965	*Elementary Logic*, Oxford University Press
Mendelson E	1987	*Introduction to Mathematical Logic* (3:e uppl), Wadsworth and Brooks
Newton-Smith W H	1985	*Logic, An Introductory Course*, Routledge & Kegan Paul
Rosen K H	1999	*Discrete Mathematics and Its Applications* (4:e uppl), McGraw-Hill