MODELING REASONING

IN PREDICATE LOGIC

Automated Theorem Proving (ATP)

- deals with the development of computer programs which show that some statement (the *conjecture*) is a *logical/syntactic consequence* of a set of statements (the *axioms* and the *hypotheses*).

These automated systems were used in a lot of domains such as:

- mathematics (*EQP, Otter, Geometry Expert*),
- software generation (*KIDS, AMPHION*),
- software verification (*KIV, PVS*),
- hardware verification (*ACL2, HOL, ANALYTICA)*.

Dedicated (educational) automated theorem provers:

- based on semantic tableaux method: 3TAP, pTAP, leanTAP, Cassandra;
- based on resolution method: OTTER, PCPROVE, AMPHION, Jape;
- based on semantic trees + Herbrand theorem: HERBY;
- based on model elimination calulus: SETHEO;

Example1

H_1. All hummingbirds are richly colored.	$H_1 : (\forall x)(hb(x) \to rc(x))$
H_2. No large birds live on honey.	$H_2 : \neg(\exists x)(\neg sb(x) \wedge lh(x)) \equiv$ $\equiv (\forall x)(\neg sb(x) \to \neg lh(x))$
H_3. Birds that do not live on honey are dull in color.	$H_3 : (\forall x)(\neg lh(x) \to \neg rc(x))$
H_4. *Piky* is a hummingbird.	$H_4 : hb(Piky)$
<u>*Conclusion:*</u> C. All hummingbirds are small.	$C : (\forall x)(hb(x) \to sb(x))$
$H_1, H_2, H_3 \vdash^{?} C$.	

Example2

H_1. Every child loves *Santa*.	$H_1 : (\forall x)(child(x) \rightarrow loves(x, Santa))$
H_2. Everyone who loves *Santa* loves any reindeer.	$H_2 : (\forall x)(\forall y)(loves(x, Santa) \wedge reindeer(y) \rightarrow loves(x, y))$
H_3. *Rudolph* is a reindeer, and *Rudolph* has a red nose.	$H_3 : reindeer(Rudolph) \wedge red\_nose(Rudolph)$
H_4. Anything which has a red nose is weird or is a clown.	$H_4 : (\forall z)(red\_nose(z) \rightarrow weird(z) \vee clown(z))$
H_5. No reindeer is a clown.	$H_5 : (\forall s)(reindeer(s) \rightarrow \neg clown(s))$
H_6. *Scrooge* does not love anything which is weird.	$H_6 : (\forall t)(weird(t) \rightarrow \neg loves(Scrooge, t))$
C. *Scrooge* is not a child.	$C : \neg child(Scrooge)$
$H_1, H_2, H_3, H_4, H_5, H_6 \overset{?}{\vdash} C$.	where: x, u, y, z, s, t are variables, $Rudolph, Santa, Scrooge$ are constants, $child, reindeer, red\_nose, weird, clown$ - unary predicates $loves$ is a binary predicate

Example2 (contd.)

We apply a refutation proof method: general predicate resolution.

T. $U1, U2, \ldots, Un \vdash V$ **if and only if** $\{U1^c, U2^c, \ldots, Un^c, (\neg V)^c\} \vdash^{\mathrm{Pr}}_{\mathrm{Res}} \square$.

The clausal normal forms corresponding to the hypotheses and the negation of the conclusion are as follows:

$H_1^{\,C} : \neg child(x) \vee loves(x, Santa) = C_1$

$H_2^{\,C} : \neg loves(x, Santa) \vee \neg reindeer(y) \vee loves(x, y) = C_2$

$H_3^{\,C} : reindeer(Rudolph) \wedge red\_nose(Rudolph) = C_3 \wedge C_4$

$H_4^{\,C} : \neg red\_nose(z) \vee weird(z) \vee clown(z) = C_5$

$H_5^{\,C} : \neg reindeer(s) \vee \neg clown(s) = C_6$

$H_6^{\,C} : \neg weird(t) \vee \neg loves(Scrooge, t) = C_7$

$(\neg C)^C : child(Scrooge) = C_8$

To the set of clauses $S = \{C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8\}$, **general predicate resolution is applied.**

Example2 (contd.)

To the set of clauses $S=\{C_1,C_2,C_3,C_4,C_5,C_6,C_7,C_8\}$ **, general predicate resolution is applied.**

In the resolution process, the following resolvents are obtained:

$$C_9=\mathrm{Res}^{\mathrm{Pr}}_{[x\leftarrow Scrooge]}(C_8,C_1)=loves(Scrooge,Santa)$$

$$C_{10}=\mathrm{Res}^{\mathrm{Pr}}_{[s\leftarrow Rudolph]}(C_3,C_6)=\neg clown(Rudolph)$$

$$C_{11}=\mathrm{Res}^{\mathrm{Pr}}_{[z\leftarrow Rudolph]}(C_4,C_5)=weird(Rudolph)\vee clown(Rudolph)$$

$$C_{12}=\mathrm{Res}^{\mathrm{Pr}}(C_{10},C_{11})=weird\,(Rudolph)$$

$$C_{13}=\mathrm{Res}^{\mathrm{Pr}}_{[t\leftarrow Rudolph]}(C_{12},C_7)=\neg loves(Scrooge,Rudolph)$$

$$C_{14}=\mathrm{Res}^{\mathrm{Pr}}_{[y\leftarrow Rudolph]}(C_2,C_3)=\neg loves(x,Santa)\vee loves(x,Rudolph)$$

$$C_{15}=\mathrm{Res}^{\mathrm{Pr}}_{[x\leftarrow Scrooge]}(C_{13},C_{14})=\neg loves(Scrooge,Santa)$$

$$C_{16}=\mathrm{Res}^{\mathrm{Pr}}(C_9,C_{15})=\square$$

The most general unifier generated during the resolution process is the substitution:

$$[x\leftarrow Scrooge,y\leftarrow Rudolph,z\leftarrow Rudolph,s\leftarrow Rudolph,t\leftarrow Rudolph]$$

$S\vdash^{\mathrm{Pr}}_{\mathrm{Res}}\square$, **therefore** S **is an inconsistent set and the deduction**

$H_1,H_2,H_3,H_4,H_5,H_6\vdash C$ **holds.**

Example3

Hypotheses: and Conclusion	Formalization
Hypotheses: H_1: Any Computer Science student likes ***logic*** and likes any programming language. H_2: Someone who likes ***logic*** is a Computer Science student or a Philosophy student. H_3: ***Java*** is a programming language. H_4: ***John*** doesn't like ***Java*** but he likes ***logic***. **Conclusion:** C: John is a Philosophy student but he is not a Computer Science student. $H_1, H_2, H_3, H_4 \overset{?}{\vdash} C$	$H_1 : (\forall x)(\forall y)(CS(x) \wedge pl(y) \rightarrow likes(x, logic) \wedge likes(x, y))$ $H_2 : (\forall z)(likes(z, logic) \rightarrow CS(z) \vee P(z))$ $H_3 : pl(Java)$ $H_4 : likes(John, \text{logic}) \wedge \neg likes(John, Java)$ $C : \neg CS(John) \wedge P(John)$ where: - x, y, z are variables; - $logic, Java, John$ are constants, - CS and P are unary predicate symbols with the meanings: $CS(x)$: ' x is a Computer Science student' $P(x)$: ' x is a Philosophy student' - pl is a unary predicate, $pl(x)$: ' x is a programming language' - $likes$ is a binary predicate, $likes(x, y)$: ' x likes y

Example3 – Normal forms

$H_1, H_2, H_3, H_4 \vdash C$ **if and only if** $\{H_1{}^c, H_2{}^c, H_3{}^c, H_4{}^c, (\neg C)^c\} \vdash^{\Pr}_{\mathrm{Res}} \square$.

We transform the hypotheses and the negation of the conclusion into clausal normal forms:

$H_1 : (\forall x)(\forall y)(CS(x) \wedge pl(y) \rightarrow likes(x, logic) \wedge likes(x, y))$	$(H_1)^c = \neg CS(x) \vee \neg pl(y) \vee likes(x, logic) \wedge likes(x, y) \equiv$ $\equiv (\neg CS(x) \vee \neg pl(y) \vee likes(x, logic)) \wedge (\neg CS(x) \vee \neg pl(y) \vee likes(x, y)) = C_1 \wedge C_2$ $C_1 = \neg CS(x) \vee \neg pl(y) \vee likes(x, logic)$, $C_2 = \neg CS(x) \vee \neg pl(y) \vee likes(x, y)$
$H_2 : (\forall z)(likes(z, logic) \rightarrow CS(z) \vee P(z))$	$(H_2)^c = \neg likes(z, logic) \vee CS(z) \vee P(z) = C_3$
$H_3 : pl(Java)$	$(H_3)^c = pl(Java) = C_4$
$H_4 : likes(John, \mathrm{logic}) \wedge \neg likes(John, Java)$	$(H_4)^c = likes(John, logic) \wedge \neg likes(John, Java) = C_5 \wedge C_6$ $C_5 = likes(John, logic)$, $C_6 = \neg likes(John, Java)$
$C : \neg CS(John) \wedge P(John)$	$(\neg C)^c = CS(John) \vee \neg P(John) = C_7$

Example3 – Resolution process

We apply linear resolution to the set $S=\{C_1,C_2,C_3,C_4,C_5,C_6,C_7\}$ with C_3 as the top clause.

$C_1 = \neg CS(x) \vee \neg pl(y) \vee likes(x, logic)$,

$C_2 = \neg CS(x) \vee \neg pl(y) \vee likes(x, y)$

$C_3 = \neg likes(z, logic) \vee CS(z) \vee P(z)$

$C_4 = pl(Java)$

$C_5 = likes(John, logic)$,

$C_6 = \neg likes(John, Java)$

$(\neg C)^c = CS(John) \vee \neg P(John) = C_7$

The following resolvents are derived:

$C_8 = \mathrm{Res}^{\mathrm{Pr}}_{[z \leftarrow John]}(C_3, C_5) = CS(John) \vee P(John)$

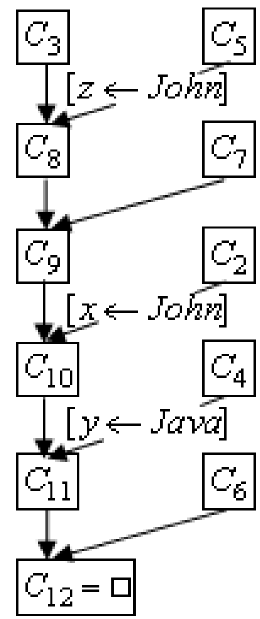
$C_9 = \mathrm{Res}(C_8, C_7) = CS(John)$

$C_{10} = \mathrm{Res}^{\mathrm{Pr}}_{[x \leftarrow John]}(C_9, C_2) = \neg pl(y) \vee likes(John, y)$

$C_{11} = \mathrm{Res}^{\mathrm{Pr}}_{[y \leftarrow Java]}(C_{10}, C_4) = likes(John, Java)$

$C_{12} = \mathrm{Res}(C_{11}, C_6) = \square$

The linear refutation process is represented graphically as follows:



$S \vdash^{lin}_{\mathrm{Res}} \square$, **therefore S is an inconsistent set and based on the hypotheses we conclude that:**

'John is a Philosophy student but he is not a Computer Science student'.

Example 4

Modeling reasoning in geometry using predicate logic

The domain is the set of all the lines in a plane.

We use variables: x, y, z to denote arbitrary objects (lines) and constants: d, d_1, d_2 to denote constant objects (lines).

Hypotheses:

H_1: **If x is perpendicular to y then x intersects y.**

H_2: **If x is parallel to y then x doesn't intersect y.**

H_3: **If x is perpendicular to y and z is perpendicular to y then x is parallel to z.**

H_4: **d_1 is perpendicular to d.**

H_5: **d is perpendicular to d_2.**

Conclusion.

C: **d_2 does not intersect d_1.**

Check whether the conclusion C is derivable from the set of hypotheses $\{H_1, H_2, H_3, H_4, H_5\}$ using a syntactic proof method.

Example 4 – Predicate formulas

$H_1 : (\forall x_1)(\forall x_2)(perpendicular(x_1, x_2) \rightarrow intersects(x_1, x_2))$

$H_2 : (\forall x_3)(\forall x_4)(parallel(x_3, x_4) \rightarrow \neg intersects(x_3, x_4))$

$H_3 : (\forall x_5)(\forall x_6)(\forall x_7)(perpendicular(x_5, x_6) \wedge perpendicular(x_7, x_6) \rightarrow parallel(x_5, x_7))$

$H_4 : perpendicular(d_1, d)$

$H_5 : perpendicular(d, d_2)$

$C : \neg intersects(d_2, d_1)$

We have to add predicate formulas which express the properties of *symmetry* for the geometric relations: $parallel, perpendicular, intersects.$

$P_1 : (\forall x_8)(\forall x_9)(parallel(x_8, x_9) \rightarrow parallel(x_9, x_8))$

$P_2 : (\forall x_{10})(\forall x_{11})(perpendicular(x_{10}, x_{11}) \rightarrow perpendicular(x_{11}, x_{10}))$

$P_3 : (\forall x_{12})(\forall x_{13})(intersects(x_{12}, x_{13}) \rightarrow intersects(x_{13}, x_{12}))$

The properties of *reflexivity* for $intersects$**, and *transitivity* for** $parallel$ **:**

$P_5 : (\forall x_{15}) intersects(x_{15}, x_{15})$

$P_6 : (\forall x_{16})(\forall x_{17})(\forall x_{17})(parallel(x_{16}, x_{17}) \wedge parallel(x_{17}, x_{18}) \rightarrow parallel(x_{16}, x_{18}))$

Example 4 – Normal forms

The clausal normal forms of the hypotheses: H_1, H_2, H_3, H_4, H_5, the negation of the conclusion C and the properties: $P_1, P_2, P_3, P_4, P_5, P_6$ are as follows:

$H_1^C : \neg perpendicular(x_1, x_2) \vee intersects(x_1, x_2) : C_1$

$H_2^C : \neg parallel(x_3, x_4) \vee \neg intersects(x_3, x_4) : C_2$

$H_3^C : \neg perpendicular(x_5, x_6) \vee \neg perpendicular(x_7, x_6) \vee parallel(x_5, x_7) : C_3$

$H_4^C : perpendicular(d_1, d) : C_4$

$H_5^C : perpendicular(d, d_2) : C_5$

$(\neg C)^C : intersects(d_2, d_1) : C_6$

$P_1^C : \neg parallel(x_8, x_9) \vee parallel(x_9, x_8) : C_7$

$P_2^C : \neg perpendicular(x_{10}, x_{11}) \vee perpendicular(x_{11}, x_{10}) : C_8$

$P_3^C : \neg intersects(x_{12}, x_{13}) \vee intersects(x_{13}, x_{12}) : C_9$

$P_5^C : intersects(x_{15}, x_{15}) : C_{11}$

$P_6^C : \neg parallel(x_{16}, x_{17}) \vee \neg parallel(x_{17}, x_{18}) \vee parallel(x_{16}, x_{18}) : C_{12}$

We apply general resolution to the set of clauses: $S = \{C_1, C_2, \ldots, C_{12}\}$.

Example 4 – Resolution process

We apply general resolution to the set of clauses: $S=\{C_1, C_2, \ldots, C_{12}\}$.

The following resolvents are obtained during the resolution process:

$C_{13} = \text{Res}^{\text{Pr}}_{[x_{10}\leftarrow d, x_{11}\leftarrow d_2]}(C_5, C_8) = perpendicular(d_2, d)$

$C_{14} = \text{Res}^{\text{Pr}}_{[x_5\leftarrow d_1, x_6\leftarrow d]}(C_3, C_4) = \neg perpendicular(x_7, d) \vee parallel(d_1, x_7)$

$C_{15} = \text{Res}^{\text{Pr}}_{[x_7\leftarrow d_2]}(C_{13}, C_{14}) = parallel(d_1, d_2)$

$C_{16} = \text{Res}^{\text{Pr}}_{[x_3\leftarrow d_1, x_4\leftarrow d_2]}(C_{15}, C_2) = \neg intersects(d_1, d_2)$

$C_{17} = \text{Res}^{\text{Pr}}_{[x_{13}\leftarrow d_1, x_{12}\leftarrow d_2]}(C_{16}, C_9) = \neg intersects(d_2, d_1)$

$C_{18} = \text{Res}(C_{17}, C_6) = \square$

The empty clause was derived from the set S of clauses, so S is inconsistent and the deduction $H_1, H_2, H_3, H_4, H_5 \vdash C$ holds.

The conclusion 'd_2 ***does not intersect*** d_1' is valid, based on the validity of the hypotheses.

Example 5 - Mathematical reasoning modeling in algebra

Prove:

"If every element of a group G is its own inverse, then G is an Abelian group".

We introduce the axioms, H_1, H_2 which define the group, the hypothesis, H_3: "every element of the group is its own inverse", and the conclusion C: "the group is an Abelian group".

Mathematical language:

$H_1 : (\forall x)(\forall y)(\forall z)[(x * y) * z = x * (y * z)]$ - associativity

$H_2 : (\forall x)[x * e = e * x = x]$ - e - neutral element

$H_3 : (\forall x)[x * x = e]$ - every element is its own inverse

$C : (\forall x)(\forall y)[x * y = y * x]$ - conclusion: G is an Abelian group

Example 5 – Predicate formulas and clauses

Mathematical language	*First-order logic language*
➢ associativity: $H_1:(\forall x)(\forall y)(\forall z)[(x*y)*z=x*(y*z)]$ ➢ e - neutral element $H_2:(\forall x)[x*e=e*x=x]$ ➢ every element is its own inverse $H_3:(\forall x)[x*x=e]$ Conclusion: ➢ G is an Abelian group $C:(\forall x)(\forall y)[x*y=y*x]$	We use the ternary predicate symbol P, with the meaning: $P(x,y,z)$:" $x*y=z$" , ‘e’ is a constant. The formulas U_1 and U_2 correspond to H_1: $U_1:(\forall x)(\forall y)(\forall z)(\forall u)(\forall v)(\forall w)[P(x,y,u)\wedge P(u,z,w)\wedge P(y,z,v)\to P(x,v,w)]$ $U_2:(\forall x)(\forall y)(\forall z)(\forall u)(\forall v)(\forall w)[P(y,z,v)\wedge P(x,v,w)\wedge P(x,y,u)\to P(u,z,w)]$ $U_1^C:\neg P(x,y,u)\vee\neg P(u,z,w)\vee\neg P(y,z,v)\vee P(x,v,w)=C_1$ $U_2^C:\neg P(y,z,v)\vee\neg P(x,v,w)\vee\neg P(x,y,u)\vee P(u,z,w)=C_2$ The formulas U_3 and U_4 correspond to H_2: $U_3:(\forall x)P(x,e,x)$, $U_3^C:P(s,e,s)=C_3$ $U_4:(\forall x)P(e,x,x)$, $U_4^C:P(e,r,r)=C_4$ The formula U_5 corresponds to H_3: $U_5:(\forall x)P(x,x,e)$, $U_5^C:P(l,l,e)=C_5$ The formula U_6 corresponds to the conclusion: $U_6:(\forall x)(\forall y)(\exists t)(P(x,y,t)\to P(y,x,t))$ $\neg U_6:\neg((\forall x)(\forall y)(\exists t)(P(x,y,t)\to P(y,x,t)))\equiv$ $\equiv(\exists x)(\exists y)(\forall t)(P(x,y,t)\wedge\neg P(y,x,t))$ $(\neg U_6)^C:P(a,b,t)\wedge\neg P(b,a,t)=C_6\wedge C_7$ a,b – Skolem constants In the clauses we renamed some of the free variables.
Checking whether $H_1,H_2,H_3\vdash C$ was reduced to checking whether $U_1,U_2,U_3,U_4,U_5\vdash U_6$. We check the inconsistency of the set of clauses: $S=\{C_1,C_2,C_3,C_4,C_5,C_6,C_7\}$.	

Example 5 – Resolution process

Predicate resolution is applied to the set $S=\{C_1,C_2,C_3,C_4,C_5,C_6,C_7\}$. **The literals resolved upon from the parent clauses are underlined.**	**Unifiers**
$C_1=\underline{\neg P(x,y,u)}\vee\neg P(u,z,w)\vee\neg P(y,z,v)\vee P(x,v,w)$, $C_5=\underline{P(l,l,e)}$ • $\mathbf{C_8}=\mathrm{Res}^{\mathrm{Pr}}_{\theta_1}(C_1,C_5)=\neg P(e,z,w)\vee\neg P(l,z,v)\vee P(l,v,w)$	$\theta_1=[x\leftarrow l,y\leftarrow l,u\leftarrow e]=mgu(P(x,y,u),P(l,l,e))$
$C_2=\neg P(y,z,v)\vee\neg P(x,v,w)\vee\underline{\neg P(x,y,u)}\vee P(u,z,w)$, $C_6=\underline{P(a,b,t)}$ • $\mathbf{C_9}=\mathrm{Res}^{\mathrm{Pr}}_{\theta_2}(C_2,C_6)=\neg P(b,z,v)\vee\neg P(a,v,w)\vee P(t,z,w)$	$\theta_2=[x\leftarrow a,y\leftarrow b,u\leftarrow t]=mgu(P(a,b,t),P(x,y,u)))$
$C_4=\underline{P(e,r,r)}$, $C_8=\underline{\neg P(e,z,w)}\vee\neg P(l,z,v)\vee P(l,v,w)$ • $\mathbf{C_{10}}=\mathrm{Res}^{\mathrm{Pr}}_{\theta_3}(C_4,C_8)=\neg P(l,r,v)\vee P(l,v,r)$	$\theta_3=[z\leftarrow r,w\leftarrow r]=mgu(P(e,z,w),P(e,r,r))$
$C_5=\underline{P(l,l,e)}$, $C_9=\underline{\neg P(b,z,v)}\vee\neg P(a,v,w)\vee P(t,z,w)$ • $\mathbf{C_{11}}=\mathrm{Res}^{\mathrm{Pr}}_{\theta_4}(C_5,C_9)=\neg P(a,e,w)\vee P(t,b,w)$	$\theta_4=[l\leftarrow b,z\leftarrow b,v\leftarrow e]=mgu(P(l,l,e),P(b,z,v))$
$C_3=\underline{P(s,e,s)}$, $C_{11}=\underline{\neg P(a,e,w)}\vee P(t,b,w)$ • $\mathbf{C_{12}}=\mathrm{Res}^{\mathrm{Pr}}_{\theta_5}(C_3,C_{11})=P(t,b,a)$,	$\theta_5=[s\leftarrow a,w\leftarrow a]=mgu(P(s,e,s),P(a,e,w))$

Example 5 – Resolution process (contd.)

$C_{10} = \underline{\neg P(l,r,v)} \vee P(l,v,r)$, $C_{12} = \underline{P(t,b,a)}$ • $\boldsymbol{C_{13}} = \mathrm{Res}^{\mathrm{Pr}}_{\theta_6}(C_{10}, C_{12}) = P(t,a,b)$	$\theta_6 = [l \leftarrow t, r \leftarrow b, v \leftarrow a] = mgu(P(t,b,a), P(l,r,v))$
$C_2 = \underline{\neg P(y,z,v)} \vee \neg P(x,v,w) \vee \neg P(x,y,u) \vee P(u,z,w)$, $C_5 = \underline{P(l,l,e)}$ • $\boldsymbol{C_{14}} = \mathrm{Res}^{\mathrm{Pr}}_{\theta_7}(C_2, C_5) = \neg P(x,e,w) \vee \neg P(x,l,u) \vee P(u,l,w)$	$\theta_7 = [y \leftarrow l, z \leftarrow l, v \leftarrow e] = mgu(P(l,l,e), P(y,z,v))$
$C_3 = \underline{P(s,e,s)}$, $C_{14} = \underline{\neg P(x,e,w)} \vee \neg P(x,l,u) \vee P(u,l,w)$ • $\boldsymbol{C_{15}} = \mathrm{Res}^{\mathrm{Pr}}_{\theta_8}(C_3, C_{14}) = \neg P(s,l,u) \vee P(u,l,s)$	$\theta_8 = [x \leftarrow s, w \leftarrow s] = mgu(P(s,e,s), P(x,e,w))$
$C_{13} = \underline{P(t,a,b)}$, $C_{15} = \underline{\neg P(s,l,u)} \vee P(u,l,s)$ • $\boldsymbol{C_{16}} = \mathrm{Res}^{\mathrm{Pr}}_{\theta_9}(C_{13}, C_{15}) = P(b,a,t)$	$\theta_9 = [s \leftarrow t, l \leftarrow a, u \leftarrow b] = mgu(P(t,a,b), P(s,l,u))$
$C_7 = \underline{\neg P(b,a,t)}$, $C_{16} = \underline{P(b,a,t)}$ • $\boldsymbol{C_{17}} = \mathrm{Res}^{\mathrm{Pr}}(C_{16}, C_7) = \square$, **so the set** S **is inconsistent and** $U_1, U_2, U_3, U_4, U_5 \vdash U_6$.	
We conclude that the statement “***If every element of a group*** G ***is its own inverse, then*** G ***is an Abelian group***” is valid.	

Example 5 (contd.)

The resolution process is represented graphically by the following binary tree.