Logika és informatikai alkalmazásai példatár

Németh L. Zoltán, Iván Szabolcs

2014

A tananyag a TÁMOP-4.1.2.A/1-11/1-2011-0104 "A felsőfokú informatikai oktatás minőségének fejlesztése, modernizációja" c. projekt keretében a Pannon Egyetem és a Szegedi Tudományegyetem együttműködésében készült.

Logika és informatikai alkalmazásai példatár

Németh L. Zoltán, Iván Szabolcs

Bevezetés

Ez a példatár elsősorban a Szegedi Tudományegyetem ,,Logika és informatiakai alkalmazásai" c. kurzus gyakorlataihoz készült. Célunk, hogy egységes, az előadás menetét és jelölésmódját követő tananyagot adjunk a hallgatók kezébe.
Mindkettőnk sokéves gyakorlati tapasztalatain túl a feladatok kiválasztásánál az alábbi munkákból is merítettünk:

- Szendrei Ágnes: Diszkrét matematika, Polygon, Szeged, 2004.
- Ésik Zoltán: Logika és informatikai alkalmazásai, előadásvázlatzok, `http://www.inf.u-szeged.hu/tanszekek/szamitastudomanyalapjai/logika.pdf`
- Fülöp Zoltán: Gyakorló feladatok a "Logika a számítástudományban" tárgyhoz I. "Ítéletkalkulus", II. "Predikátumkalkulus", kézirat.
- Kalmárné Németh Márta, Katonáné Horváth Eszter, Kámán Tamás: Diszkrét matematikai feladatok. Szeged, Polygon, 2003.
- Lengyel Zoltán: Logikai feladatgyűjtemény, Debrecen Egyetem, Informatikai Kar, 2005, `http://www.inf.unideb.hu/~lengyelz/docs/logika.pdf`
- Pásztorné Varga Katalin, Várterész Magda, Kósa Márk, Róbert Édelkraut, A matematikai logika alkalmazásszemléletű tárgyalása, Panem Könyvkiadó Kft., Budapest, 2003.
- Serény György: Matematikai logika jegyzet, 1. rész: Propozicionális logika, 2. rész: Predikátum logika `www.math.bme.hu/~sereny/`
- Raymond Smullyan: Mi a címe ennek a könyvnek?, TypoTeX, Budapest, 2005.
- I. A. Lavrov – L. L. Makszimova: Halmazelméleti, matematikai logikai és algoritmuselméleti feladatok, Műszaki Könyvkiadó, Budapest, 1987

1. fejezet

A predikátumkalkulus szintaxisa

Elméleti összefoglaló

Minden logikai rendszer definiálásakor két dolgot kell megadnunk. Ezek az alábbiak.

a) **Szintaxis:** Melyek a szabályos formulák? Ennek nem tulajdonítható jelentés, csak formális szabályok.

b) **Szemantika:** Mikor igaz egy formula egy adott modellben? Ez határozza meg a formulák jelentését, az igazság fogalmát, és hogy mikor tekintünk egy logikai következtetést helytállónak.

Mindenek előtt fontos leszögeznünk, hogy logika tanulmányaink kezdetén két alapvető logikai rendszert különböztetünk meg. Az egyik a *zérusrendű logika*, más néven *ítéletkalkulus*, a másik az *elsőrendű logika*, más néven *predikátumkalkulus*. Annak ellenére, hogy a zérusrendű logika azonosságait és bizonyos módszereit az elsőrendű logikára is alkalmazni fogjuk, fontos, hogy a köztük levő különbséget tisztán lássuk. A továbbiakban először a diszkrét matematika tárgyból szerzett ismeretekre támaszkodva a zérusrendű logikát ismételjük át röviden, majd az elsőrendű logika szintaxisának ismertetésére térünk rá.
Az ítéletkalulusban, ahogy a neve is mutatja, csak ítétletváltozók vannak, azaz olyan változók, melyek csak a 0 (logikai hamis) és 1 (logikai igaz) értéket vehetik fel. Az ítéletváltozókat az ábécé végéről, általában p, q, r, ... betűkkel (esetenként nagybetűkkel : P, Q, R, ...) jelöljük. Formulákat a változókból zárójelekkel és az alábbi, logikai műveletekkel képezhetünk: $\neg$ („nem", vagy negáció), $\vee$ („vagy" vagy diszjunkció), $\wedge$ („és" vagy konjunkció), $\rightarrow$ („akkor" vagy implikáció), $\leftrightarrow$ („akkor és csak akkor" vagy ekvivalencia). Ezen kívül használjuk még a $\uparrow$ (azonosan igaz), és a $\downarrow$ (azonosan hamis) műveleteket is. Néhol találkozunk még az $\oplus$ („kizáró vagy" vagy „XOR" vagy anti-ekvivalencia), $|$ („nem és" vagy „NAND"), $||$

(,,nem vagy", vagy ,,NOR") jelölésekkel. Ezek jelentését itt most nem ismertetjük, később sor fog rá kerülni, de aki hiányosságot érez e téren, könnyen utánanézhet például Szendrei Ágnes Diszkrét matematika című könyvében. A zérusrendű logikában csak ezeknek a műveleteknek az összefüggéseiről tudunk beszélni, mint például, hogy logikai törvények a De-Morgan azonosságok, azaz mindig igazak az alábbi formulák:

$$\neg(p \wedge q) \leftrightarrow (\neg p \vee \neg q),$$
$$\neg(p \vee q) \leftrightarrow (\neg p \wedge \neg q).$$

Ennél nagyobb kifejezőerővel rendelkezik az elsőrendű logika, melyben a változók tetszőleges objektumhalmazból felvehetik értéküket. A továbbiakban ezzel foglalkozunk.

Az elsőrendű logika szintaxisa

Bár ritkán definiáljuk explicit módon, minden logikai rendszer definiálása a *jelkészlet* megadásával kezdődik, mely definiálja, hogy mely szimbólumok szerepelhetnek a formulákban. A továbbiakban végig az alábbi jelkészlettel dolgozunk:

- változók (*Var*): x, y, z, ...
- függvényszimbólumok (*Fgv*):
  - konstansok: c, d, ...
  - többváltozósak: f, g, h, ...
- predikátumszimbólumok (*Pred*): p, q, r, ...
- logikai jelek (*Log*): $\uparrow$, $\downarrow$, $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, $\forall$, $\exists$
- egyéb jelek (*Aux*): zárójelek: (), [], {}, vessző.

A különböző típusú zárójelpárok egyenértékűek, használatukkal, csak az összetettebb formulák olvashatóságát szeretnénk esetenként megkönnyíteni.

Feltesszük, hogy a *Var*, *Fgv* és *Pred* halmazok (utóbbiak minden véges aritásra is) megszámlálhatóan végtelen sok szimbólumot tartalmaznak, ezt a fenti betűk indexelt változatainak használatával érhetjük el. Például x_1, x_2, ... valamennyien változók. A fenti jelkészletet $\mathcal{L} = (Var, Fgv, Pred, Log, Aux)$ típusú nyelvnek is nevezzük.

Ezután az elsőrendű logika szintaxisát négy kategorikus egység segítségével definiálhatjuk. Ezek: I. Változók, II. Termek, III. Atomi formulák és IV. formulák. Ezek definíciója rendre az alábbi.

I. Változók

A változók *Var* halmaza a jelkészlettel együtt adott, nem szorul definiálásra.

II. Termek

- Minden változó term.
- Minden konstans term.
- Termekből a függvényszimbólumok alkalmazásával újabb termek képezhetők. Természetesen a függvényszimbólum aritásának (a változói számának) a tiszteletben tartásával az alábbi szabály szerint

$$\boxed{\frac{t_1, \ldots, t_n}{f(t_1, \ldots, t_n)}, \text{ ahol } f \text{ egy } n \text{ rangú fgv. szimbólum, } n \geq 0.}$$

Itt és a továbbiakban az ilyen törtvonallal írt szabályok mindig úgy értelmezendők, hogy amennyiben a vonal feletti $t_1, \ldots t_n$ objektumról már tudjuk, hogy a halmazba tartoznak (esetünkben termek), akkor a vonal alatti objektum (most $f(t_1, \ldots, t_n)$) is a halmazba tartozik, (azaz esetünkben term). Például $f(x, g(g(c)))$ term, ha x változó, c konstans, f kétváltozós, g pedig egyváltozós függvényszimbólum.
Megjegyzés. *Alaptermek* más szóval ground termek az olyan termek, melyekben változó nem fordul elő, azaz csak konstansokból és függvény jelekből épülnek fel. Például $g(f(c, c))$ alapterm az előbbi feltételek mellett.

III. Atomi formulák

- A predikátumszimbólumokba termek beírásával kapott kifejezés.

$$\boxed{\frac{t_1, \ldots, t_n}{p(t_1, \ldots, t_n)}, \ t_1, \ldots, t_n \text{ term, } p \ n \text{ rangú pred. szimb., } n \geq 0.}$$

Például $p(x, g(c), c)$ atomi formula, ha p háromváltozós predikátumszimbólum, g egyváltozós függvényszimbólum, x változó, c pedig konstans. Speciálisan e szabály szerint a konstans (azaz 0 változó) predikátumszimbólumok önmagukban atomi formulák.
Fontos, hogy ez a szabály nem induktív, minden atomi formula készítéséhez pontosan egyszer alkalmazzuk.

IV. Formulák

- Minden atomi formula formula.

- A konstans műveleti jelek: $\uparrow$ és $\downarrow$ önmagukban formulák.
- Formulákból a $\neg$, $\vee$, $\wedge$, $\rightarrow$, $\leftrightarrow$, $\forall$, $\exists$ logikai műveletekkel újabb formulák képezhetők az alábbi szabályok szerint:

$$\boxed{\frac{F}{(\neg F)}}, \boxed{\frac{F, G}{(F \vee G)}}, \boxed{\frac{F, G}{(F \wedge G)}}, \boxed{\frac{F, G}{(F \rightarrow G)}}, \boxed{\frac{F, G}{(F \leftrightarrow G)}}$$

$$\boxed{\frac{F}{(\forall x F)}}, \boxed{\frac{F}{(\exists x F)}}.$$

Természetesen a kvantorok alkalmazásakor x változó. Például $(\forall y(p(x, g(c), c) \vee (\neg p(c, c, c))))$ formula, a fenti feltételek mellett, ha y változó.
Az elsőrendű logika szintaxisának definícióját szemlélteti az **1. animáció**.
A konstans műveleteken és az atomi formulákon kívül minden formula a fenti szabályokkal állítható elő. Általában több szabályt is alkalmazhatunk egymás után. Bármely szabály alkalmazásakor a vonal feletti formulákat, melyekből építhetünk, a végső formula *részformuláinak* nevezzük. Az utoljára alkalmazott szabály feltételei a végső formula **közvetlen részformulái** (vagy közvetlen részformulája a $\neg$ és a kvantoros szabályok esetében).
Megjegyzés. Az utolsó két szabály alkalmazásához nem szükséges, hogy a kvantor y változója elő is forduljon a részformulában, amire a kvantort alkalmazzuk. Persze, ha kvantor változója a kvantifikált részben nem fordul elő szabadon, akkor a kvantornak nincs hatása, szintaktikailag azonban nem hibás a formula.
A műveletek precedencia sorrendje.
Formálisan fenti szabályok alkalmazásakor minden részeredményként kapott formulát zárójelbe kell tenni, például $((p(x) \vee (\forall y q(y))) \vee r(x, f(y)))$. Így azonban még az egyszerű formulák is nehezen olvashatók, ezért megállapodunk a zárójelek elhagyhatóságának alábbi szabályaiban.

- Mindig elhagyható a **legkülső** zárójel.
- Elhagyható az ***unér*** (egyváltozós) műveleti jelek ($\neg$, $\forall$, $\exists$) **körüli** zárójel, azaz ezek a jelek erősebben kötnek bármely binér műveletnél.
- Elhagyhatók a belső zárójelek három vagy több argumentumú **azonos asszociatív művelet** ($\vee$, $\wedge$ és $\leftrightarrow$) esetén, például $(F \wedge G) \wedge H$ és $F \wedge (G \wedge H)$ helyett $F \wedge G \wedge H$ írható.
- Elhagyhatók azok a zárójelek, melyek a műveletek alábbi, balról-jobbra vett **precedencia sorrendjéből** következnek:

$$\boxed{\wedge \quad \vee \quad \rightarrow \quad \leftrightarrow},$$

azaz $\wedge$ köt a legerősebben, $\leftrightarrow$ a leggyengébben.

- Bár $\to$ nem asszociatív művelet, megegyezünk, hogy az egymás után **jobbról zárójelezett implikációk körüli** zárójelek kitevésétől eltekintünk, azaz $F \to G \to H$ alatt mindig $F \to (G \to H)$-t értünk. Azonban $(F \to G) \to H$ esetén a zárójelet kötelező kitenni.

Kötött és szabad változók. Első körben nem egy változó, hanem annak egy konkrét előfordulásáról mondjuk meg, hogy kötött vagy szabad-e, például a $\forall x p(x) \to Q(x)$ formulában az x változó –közvetlenül kvantor utáni előfordulását nem számítva– első előfordulása kötött, a második pedig szabad. Definíció szerint x *egy előfordulása kötött* az F formulában, ha x ezen előfordulása egy $\forall xG$ vagy $\exists xG$ alakú részformulába esik. Az x *egy előfordulása szabad*, ha nem kötött. Végül x *szabad változó* vagy *paraméter* az F formulában, ha x-nek van szabad előfordulása F-ben. Másképp fogalmazva F-ben minden kvantor minden előfordulásához egyértelműen hozzárendelhető az a $\forall xG$ vagy $\exists xG$ alakú részformula (ahol természetesen x a kvantor változója), mely az adott kvantort F-be bevezeti. Ezt a G-t nevezzük az adott kvantor *hatáskörének*. Minden kvantor *köti* az általa kvantifikált változó összes szabad előfordulását, mely a hatáskörébe esik. Azok a változó-előfordulások, melyek egyetlen kvantor hatáskörébe se esnek bele, a *szabad változó-előfordulások*.

Feladatok

1.1. Feladat. Legyen a változók halmaza $Var = \{x, y, z, \ldots\}$, a függvényszimbólumok halmaza $Fgv = \{f, g, h, c\}$, ahol f rangja 1, g rangja 2, h rangja 3, c rangja 0 (azaz c konstans szimbólum). A predikátumszimbólumok halmaza legyen $Pred = \{p, q, r\}$, ahol p rangja 1, q-é 2, r-é 0. Írjunk fel ezen jelkészlet felett 5-5 darab

a) termet;

b) alaptermet;

c) atomi formulát;

d) formulát (de nem atomi formulát).

1.1. Feladat megoldása.

a) Term minden olyan kifejezés, melyet a változókból és a függvényszimbólumokból építhetünk a függvényszimbólumok aritásának figyelembevételével. Ide tartoznak önmagukban a konstansok is. Például x, c, $f(x)$, $h(x, x, y)$, $g(f(x), a)$, ...

b) Alaptermek a változómentes termek, például: c, $f(c)$, $g(c,c)$, $f(g(c,f(c)))$, $f(h(c,f(c),g(c,c)))$, ...

c) Atomi formulákat úgy készíthetünk, hogy a predikátumszimbólumokat az aritásuknak megfelelő számú termre alkalmazzuk. Például $p(c)$, $p(f(x))$, $q(c,g(c,y))$, $q(x,f(g(c,c)))$, r,...

d) Az atomi formulákból formulákat logikai műveletekkel készíthetünk. A definíció szerint minden lépés során a képzett részformulát zárójelbe kell tennünk. Például $(p(x) \to p(c))$, $(\forall x p(x))$, $(\exists x(\forall y(p(f(x)) \leftrightarrow (\neg q(y,x)))))$, ... Később, megállapodás alapján, a nem feltétlenül szükséges zárójeleket el fogjuk hagyni. Ide tartozik még a két konstans logikai művelettel képzett formula is: $\downarrow$ (azonosan hamis) és $\uparrow$ (azonosan igaz).

1.2. Feladat. Legyen a változók halmaza $Var = \{x_1, x_2, \ldots\}$, a függvényszimbólumok halmaza $Fgv = \{f, g, h, c\}$, ahol f rangja 1, g rangja 2, h rangja 3, c rangja 0 (azaz c konstans szimbólum). A predikátumszimbólumok halmaza legyen $Pred = \{p, q, r\}$, ahol p rangja 1, q-é 2, r-é 0. Állapítsuk meg, hogy termek-e az alábbi szavak a fentebb definiált jelkészlet felett?

a) $f(g(x_1, x_2))$;

b) $f(g(x_3), h(x_1, x_2, x_3))$;

c) $g(f(f(c)), h(x_2, x_2, x_2))$;

d) c;

e) r;

f) $\exists x_2 g(f(x_1), x_2)$;

g) $f(x_1) + g(x_1, x_2)$;

h) $g(x_1, q(r, r), f(x_2))$.

1.2. Feladat megoldása.

a) Igen.

b) Nem, mert f egyváltozós, g pedig 2.

c) Igen.

d) Igen.

e) Nem, mert r predikátumszimbólum.

f) Nem, mert kvantor a formulák képzéséhez kell.

g) Nem, mert + nem szerepel a jelkészletben.

h) Nem, mert $q(r, r)$ nem term, még csak nem is formula.

1.3. Feladat. Az előző feladat jelöléseit használva elsőrendű formulák-e az alábbiak?

a) $q(f(f(x_1)), c)$;

b) $p(c) \to \forall x_3(p(x_1) \wedge r)$;

c) $f(g(x_1, x_2))$;

d) $Qx_1p(x_1)$;

e) $\exists! x_1 p(g(x_1, x_1))$;

f) $R \wedge \forall x_1 x_2 q(x_1, x_2)$;

g) $\forall x_1(\exists x_2 p(x_1) \wedge q(x_1, x_2))$;

h) $\neg p(x_1) \to \forall c p(g(c, x_1))$;

i) $\exists n(p(x_1) \vee p(x_2) \vee \ldots \vee p(x_n) \vee \neg p(x_{n-1}))$.

1.3. Feladat megoldása.

a) Igen, ez atomi formula.

b) Igen, de nem atomi formula.

c) Nem, ez term.

d) Nem, Q nem kvantor, nincs a jelkészletben. Igaz, néha bizonyítások egyszerűsítésének érdekében a $\forall$ és a $\exists$ kvantor valamelyikének helyettesítésére szoktuk használni, de ekkor is odaírandó, hogy $Q \in \{\forall, \exists\}$.

e) Nem, a $\exists!$ rövidítést formulákban nem engedtük meg, nincs „!” a jelkészletben. Matematikai bizonyításokban $\exists!$ a „létezik pontosan egy” rövidítése szokott lenni.

f) Nem, az x_2 változó elől hiányzik egy kvantor.

g) Igen. Ugyan mivel a $\exists x_2$ kvantor csak a $p(x_1)$ részformulára hat, x_2 szabad változó, és ennek a kvantifikációnak a formula értékére nincs hatása, szemantika szempontjából felesleges. Szintaktikailag azonban a formula helyes.

h) Nem, mert c nem változó, hanem konstans.

i) Nem, n nem változó, és ... nem szerepel a jelkészletben.

1.4. Feladat. Itt és a továbbiakban a következő jelöléseket használjuk: *predikátumszimbólumok*: $p, q, r, \ldots$, *függvényszimbólumok*: $f, g, h, \ldots$, *változók*: $x, y, z, \ldots$, *konstansok* c, d, e, valamint ezek indexelt változatai. Mindig feltesszük, hogy a predikátum és függvényszimbólumok olyan aritásúak, ahogy a formulában szerepelnek. Soroljuk fel az alábbi formulák összes részformuláját! Melyek közülük a közvetlen részformulák?

a) $\exists x(\forall y(q(x) \to p(x,y)))$

b) $(q(z) \to \neg\exists z\forall x p(z,x)) \to \neg\forall y q(y,x)$

c) $q(g(y,x), f(y))$

d) $\neg[(\exists x(p(x) \to q(x,y)) \wedge (q(y,x) \to r(x))) \to \forall x(\neg p(x))]$.

1.4. Feladat megoldása. A formulák önmaguknak mindig részformulái, ezen kívül részformulák még az alábbiak:

a)
- $q(x)$, $p(x,y)$,
- $q(x) \to p(x,y)$,
- közvetlen részformula: $(\forall y(q(x) \to p(x,y))$.

b)
- $q(z)$, $\forall y q(y,x)$,
- $p(z,x)$, $\forall x p(z,x)$, $\exists z\forall x p(z,x)$, $\neg\exists z\forall x p(z,x)$, $q(y,x)$,
- közvetlen részformulák: $(q(z) \to \neg\exists z\forall x p(z,x))$, $\neg\forall y q(y,x)$.

c) A formula atomi formula, ezért egyetlen részformulája önmaga.

d)
- atomi formulák: $p(x)$, $q(x,y)$, $q(y,x)$, $r(x)$,
- $p(x) \to q(x,y)$, $\exists x(p(x) \to q(x,y))$, $q(y,x) \to r(x)$
- $\exists x(p(x) \to q(x,y)) \wedge (q(y,x) \to r(x))$
- $\neg p(x)$ és $\forall x(\neg p(x))$
- közvetlen részformula: $(\exists x(p(x) \to q(x,y)) \wedge (q(y,x) \to r(x))) \to \forall x(\neg p(x))$.

1.5. Feladat. Jelöljük be az egyes kvantorok hatáskörét! Mely változók szabad változók a formulában?

a) $\forall x(\exists y q(f(x), h(y,x,z)) \to p(x))$;

b) $\forall x(p(x) \vee \neg \exists x q(x, g(x,x))) \wedge \exists x p(f(f(x)))$;

c) $\exists x(p(x) \vee \forall y \neg q(g(x,y), y) \wedge \exists x p(x))$;

d) $\exists x \forall y p(x) \vee \neg p(x)$.

1.5. Feladat megoldása. Minden kvantor esetén a közvetlenül a kvantor után következő téglalap jelöli az adott kvantor hatáskörét:

a) $\forall x \boxed{(\exists y \boxed{q(f(x), h(y,x,z))} \to p(x))}$.

b) $\forall x \boxed{(p(x) \vee \neg \exists x \boxed{q(x, g(x,x))})} \wedge \exists x \boxed{p(f(f(x)))}$.

c) $\exists x \boxed{(p(x) \vee \forall y \boxed{\neg q(g(x,y), y)} \wedge \exists x \boxed{p(x)})}$.

d) $\exists x \boxed{\forall y \boxed{p(x)}} \vee \neg p(x)$, így x szabad változó (paraméter).

1.6. Feladat. Jelöljük be az alábbi formulákban, hogy mely kvantor melyik változót köti, és határozzuk meg a formula paramétereinek (azaz a benne szabadon (is) előforduló változóknak) halmazát!

a) $\exists x \forall y q(x,y) \vee p(x)$;

b) $\forall x q(z) \leftrightarrow \forall y \exists y q(x,y) \wedge q(y,x)$;

c) $(\forall x p(x,y) \to \forall y r(x,y)) \wedge p(c)$;

d) $\neg\exists z(q(z,z) \wedge r(f(y,z)))$;

e) $\forall x(\forall y p(x,y,z) \to q(x,y))$;

f) $\forall y \exists z(p(x,y,z) \to \exists x \forall x q(z,x))$;

g) $\exists x \forall y(p(x) \vee q(x,f(y))) \to \forall y q(x,y)$.

1.6. Feladat megoldása. Csak az egyik részfeladat megoldását ismertetjük:

g) $\exists x \boxed{\forall y \boxed{(p(x) \vee q(x,f(y)))}} \to \forall y \boxed{q(x,y)}$.

Az első $\forall y$ az y változó első előfordulását köti. Az első $\exists x$ az x változó első és második előfordulását köti. A második $\forall y$ az y változó második előfordulását köti. Végül x utolsó előfordulása szabad, így x az egyetlen paraméter.

1.7. Feladat. Tegyük fel, hogy a FÉL(x,y) predikátumszimbólumok jelentése az, hogy „x fél y-tól”. Az objektumok halmaza az erdő állatai, három konstans szimbólum pedig: „**Medve**”, „**Róka**” és „**Nyuszika**”. Fejezzük ki formulákkal ebben a rendszerben az alábbi állításokat! (Az egyenlőség predikátumszimbólumot is használhatjuk.)

a) A Nyuszika mindenkitől fél.

b) A Medve senkitől sem fél.

c) Aki a Rókától fél, az a Medvétől is fél.

d) Senki sem fél önmagától. [Azaz a félelem irreflexív.]

e) Senki sem fél önmagától, csak a Nyuszika.

f) Nem igaz, hogy a Medvétől mindenki fél, de a Medve senkitől sem fél.

g) Ha a Medve fél a Rókától, akkor a Róka fél a Nyuszikától.

h) Ha a Medvétől pontosan azok az állatok félnek, akik nem félnek a Nyuszikától, akkor van valaki, akitől a Medve nem fél. Igaz ez az állítás?]

i) Mindenki fél attól, akitől félelmének oka fél. [Azaz a félelem tranzitív.]

j) Bármely két különböző állat egyike fél a másiktól. [Azaz a félelem trichotóm.]

1.7. Feladat megoldása.

a) A Nyuszika mindenkitől fél: $\forall x\,\text{FÉL}(\textbf{Nyuszika}, x)$.

b) A Medve senkitől sem fél:
$\neg\exists x\,\text{FÉL}(\textbf{Medve}, x)$, vagy szintén helyes $\forall x\neg\,\text{FÉL}(\textbf{Medve}, x)$.

c) Aki a Rókától fél, az a Medvétől is fél:
$\forall x\,\text{FÉL}(x, \textbf{Róka}) \to \text{FÉL}(x, \textbf{Medve})$.

d) Senki sem fél önmagától: $\neg\exists x\,\text{FÉL}(x, x)$.

e) Senki sem fél önmagától, csak a Nyuszika: $\forall x(\text{FÉL}(x, x) \to (x = \textbf{Nyuszika})$, vagy $\forall x(\neg\,\text{FÉL}(x, x) \vee (x = \textbf{Nyuszika}))$.

f) Nem igaz, hogy a Medvétől mindenki fél, de a Medve senkitől sem fél:
$\neg[\forall x\,\text{FÉL}(x, \textbf{Medve}) \wedge \neg\exists x\,\text{FÉL}(\textbf{Medve}, x)]$.

g) Ha a Medve fél a Rókától, akkor a Róka fél a Nyuszikától:
$\text{FÉL}(\textbf{Medve}, \textbf{Róka}) \to \text{FÉL}(\textbf{Róka}, \textbf{Nyuszika})$.

h) Ha a Medvétől pontosan azok az állatok félnek, akik nem félnek a Nyuszikától, akkor van valaki, akitől a Medve nem fél:
$\forall x[\text{FÉL}(x, \textbf{Medve}) \leftrightarrow \neg\,\text{FÉL}(x, \textbf{Nyuszika})] \to \exists y\neg\,\text{FÉL}(\textbf{Medve}, y)]$. Az állítás biztosan igaz, mert vagy a Medve nem fél önmagától, vagy, ha fél önmagától, akkor a feltétel miatt nem szabad félnie a Nyuszikától.

i) A félelem tranzitív: $\forall x\forall y\forall z[(\text{FÉL}(x, y) \wedge \text{FÉL}(y, z)) \to \text{FÉL}(x, z)]$.

j) A félelem trichotóm: $\forall x\forall y[\text{FÉL}(x, y) \vee (x = y) \vee \text{FÉL}(y, x)]$.

1.8. Feladat. Tegyük fel, hogy az előző feladat feltételezésein túl még van egy f egyváltozós függvényünk, mely minden állathoz *annak legjobb barátját* rendeli, azaz $f(x)$ az x legjobb barátját jelöli. Tegyük fel, hogy ez minden erdőbeli állat esetén egyértelműen létezik. Legyen továbbá a BAR a „*bemegy a barlangba*" egyváltozós predikátum, azaz $\text{BAR}(x)$, igaz, ha x bemegy a barlangba. Fejezzük ki formulákkal ebben a rendszerben az alábbi állításokat! (Az egyenlőség predikátumszimbólum továbbra is használható.)

a) A Medve legjobb barátja a Nyuszika.

b) A Medve legjobb barátja nem fél a Rókától.

c) Senki sem fél a legjobb barátjától.

d) Mindenki fél attól, akinek a Medve a legjobb barátja.

e) Aki fél a Medve legjobb barátjától, az nem megy be a barlangba.

f) Aki bemegy a barlangba, az nem fél a Medve legjobb barátjától.

g) Ha a Nyuszika a Medve legjobb barátja, akkor mindenki fél a Nyuszikától, kivéve önmagát és a Medvét.

h) Az erdőben él legalább három különböző állat.

1.8. Feladat megoldása.

a) A Medve legjobb barátja a Nyuszika: $f(\mathbf{Medve}) = \mathbf{Nyuszika}$.

b) A Medve legjobb barátja nem fél a Rókától: $\neg\,\mathrm{FÉL}(f(\mathbf{Medve}), \mathbf{Róka})$.

c) Senki sem fél a legjobb barátjától: $\forall x \neg\,\mathrm{FÉL}(x, f(x)) \equiv \neg\exists x\,\mathrm{FÉL}(x, f(x))$.

d) Mindenki fél attól, akinek a Medve a legjobb barátja: $\forall x \forall y[(f(y) = \mathbf{Medve}) \to \mathrm{FÉL}(x, y)]$.

e) Aki fél a Medve legjobb barátjától, az nem megy be a barlangba: $\forall x[\mathrm{FÉL}(x, f(\mathbf{Medve})) \to \neg\,\mathrm{BAR}(x)]$.

f) Aki bemegy a barlangba, az nem fél a Medve legjobb barátjától: $\forall x[\mathrm{BAR}(x) \to \neg\,\mathrm{FÉL}(x, f(\mathbf{Medve}))]$. Vegyük észre, hogy ez az előzővel ekvivalens (egyenértékű) állítás a kontrapozíció elve alapján.

g) Ha a Nyuszika a Medve legjobb barátja, akkor mindenki fél a Nyuszikától, kivéve önmagát és a Medvét: $(\mathbf{Nyuszika} = f(\mathbf{Medve})) \to \forall x(\mathrm{FÉL}(x, \mathbf{Nyuszika}) \vee (x = \mathbf{Nyuszika}) \vee (x = \mathbf{Medve}))$.

h) Az erdőben él legalább három különböző állat: $\exists x \exists y \exists z[\neg(x = y) \wedge \neg(x = z) \wedge \neg(y = z)]$.

2. fejezet

A predikátumkalkulus szemantikája: struktúrák, kielégíthetőség, tautológiák.

Elméleti összefoglaló

Az elsőrendű logika szintaxisa azt definiálta, hogy melyek a szabályos formulák. Egy szintaktikailag helyes formula azonban még nem jelent semmit. A szemantika alapfeladata, hogy formulákhoz értelmet, jelentést társítson. Ehhez mindenekelőtt a modell, más szóval struktúra megadása szükséges, mely azt a világot, valóságot adja meg, melyben a formulát kiértékeljük. Csak a modell megadása után tudjuk majd a formula jelentését, vagyis a definiált modellben a formula logikai igaz/hamis értékét meghatározni. A struktúra fogalma az alábbi:
Elsőrendű modell definíciója. Legyen $\mathcal{L} = (Var, Fgv, Pred, Log, Aux)$ egy elsőrendű logikai nyelv.

- Az $\mathcal{A} = (A, I, \varphi)$ hármas $\mathcal{L}$-típusú *elsőrendű struktúra* vagy *elsőrendű modell*, melyben
- A tetszőleges nemüres halmaz, az *alaphalmaz* vagy *univerzum* vagy a struktúra *tartóhalmaza*;
- I az *interpretáció*, mely
  - minden $f \in Fgv$ függvény *szimbólumhoz*, melynek rangja $n \geq 0$ egy $I(f) : A^n \to A$ *valódi függvényt* rendel, és
  - minden $p \in Pred$ predikátum *szimbólumhoz*, melynek rangja $n \geq 0$ egy $I(p) : A^n \to \{0, 1\}$ *valódi predikátumot* rendel;

- $\varphi : Var \to A$ pedig a *változó hozzárendelés* vagy *változó kiértékelés*, mely minden x változónak egy A-beli $\varphi(x)$ értéket ad.

Ha nincsenek szabad változók egy formula kiértékelésekor, akkor a harmadik, φ komponens elhagyható a modellből.
A továbbiakban a könnyebb olvashatósága érdekében, amennyiben egy konkrét $\mathcal{A} = (A, I, \varphi)$ modellről van szó, a modellben a predikátum- és függvényszimbólumok interpretációját az adott jel felé tett $\tilde{}$ (hullám) szimbólummal jelöljük. Így például a p predikátumszimbólum valamint az f és 0 függvényszimbólumok interpretációja, rendre $\tilde{p} := I(p)$, $\tilde{f} := I(f)$ és $\tilde{0} := I(0)$.
Fontos, hogy a modellben az I interpretáció a függvény és predikátumszimbólumoknak tetszőleges értéket adhat. Ez alól egyetlen kivétel van. Ha az egyenlőség (=) szerepel a predikátumszimbólumok között, akkor annak interpretációja mindenképpen az objektumhalmazon értelmezett egyenlőség reláció kell, hogy legyen. Ebben az esetben *egyenlőséges logikáról* beszélünk.
Mint említettük, a formulák kiértékelése azt jelenti, hogy minden ($\mathcal{L}$-feletti) F formulához és minden ($\mathcal{L}$-feletti) $\mathcal{A}$ modellhez hozzárendeljük az F formula igaz vagy hamis logikai értékét az $\mathcal{A}$ struktúrában. Ennek jele: $\mathcal{A}(F)$.
De mielőtt ezt megtennénk, először a tetszőleges t termre kell $\mathcal{A}(t)$-t, vagyis a t term $\mathcal{A} = (A, I, \varphi)$ struktúrában felvett értékét meghatároznunk. Ez a t term felépítése szerinti indukcióval az alábbi módon történik:

- ha $t = x$, vagyis t változó, akkor legyen

$$\mathcal{A}(t) = \varphi(x);$$

- különben $t = f(t_1, \ldots, t_n)$ alakú, valamely f függvényszimbólumra és t-nél egyszerűbb felépítésű $t_1, t_2, \ldots, t_n$ termekre; ekkor legyen

$$\mathcal{A}(t) = \tilde{f}(\mathcal{A}(t_1), \mathcal{A}(t_2), \ldots, \mathcal{A}(t_n)).$$

Ezután már $\mathcal{A}(F)$-et, azaz az F formula $\mathcal{A} = (A, I, \varphi)$ modellben felvett értékét is meghatározhatjuk. Ez a termek kiértékeléséhez hasonlóan az F formula felépítése szerint történik:

- ha $F =\uparrow$, akkor $\mathcal{A}(F) = 1$, (vagyis az azonosan igaz formula értéke 1);
- ha $F =\downarrow$, akkor $\mathcal{A}(F) = 0$, (vagyis az azonosan hamis formula értéke 0);
- ha $F = p(t_1, \ldots, t_n)$ atomi formula, akkor

$$\mathcal{A}(F) = \tilde{p}(\mathcal{A}(t_1), \mathcal{A}(t_2), \ldots, \mathcal{A}(t_n)),$$

- ha $F = \neg F_1$, akkor

$$\mathcal{A}(F) = \begin{cases} 1, & \text{ha } \mathcal{A}(F_1) = 0 \\ 0 & \text{különben, vagyis ha } \mathcal{A}(F_1) = 1; \end{cases}$$

- ha $F = F_1 \vee F_2$, akkor

$$\mathcal{A}(F) = \begin{cases} 1, & \text{ha } \mathcal{A}(F_1) = 1 \textbf{ vagy } \mathcal{A}(F_2) = 1, \\ 0, & \text{különben;} \end{cases}$$

- ha $F = F_1 \wedge F_2$, akkor

$$\mathcal{A}(F) = \begin{cases} 1, & \text{ha } \mathcal{A}(F_1) = 1 \textbf{ és } \mathcal{A}(F_2) = 1, \\ 0, & \text{különben;} \end{cases}$$

- ha $F = F_1 \to F_2$, akkor

$$\mathcal{A}(F) = \begin{cases} 1, & \text{ha } \mathcal{A}(F_1) = 0 \text{ vagy } \mathcal{A}(F_2) = 1, \\ 0, & \text{különben;} \end{cases}$$

- ha $F = F_1 \leftrightarrow F_2$, akkor

$$\mathcal{A}(F) = \begin{cases} 1, & \text{ha } \mathcal{A}(F_1) = \mathcal{A}(F_2) = 0 \text{ vagy } \mathcal{A}(F_1) = \mathcal{A}(F_2) = 1, \\ 0, & \text{különben.} \end{cases}$$

A kvantorok szemantikájának megadásához szükségünk van tetszőleges $\mathcal{A} = (A, I, \varphi)$ modell olyan módosítására, mely egy x változó értékét egy tetszőleges, de előre rögzített $a \in A$ elembe viszi. Ez a következő:
$\mathcal{A}_{[x \mapsto a]} = (A, I, \varphi')$, ahol bármely y változóra

$$\varphi'(y) = \begin{cases} a, & \text{ha } y = x, \\ \varphi(y), & \text{különben.} \end{cases}$$

Azaz, φ' ugyanaz, mint φ, kivéve, hogy $\varphi'(x) = a$.
Ennek segítségével a kvantorokat így értelmezzük:

- ha $F = \exists x G$, akkor

$$\mathcal{A}(F) = \begin{cases} 1, & \text{ha } \textbf{létezik } a \in A, \text{ melyre } \mathcal{A}_{[x \mapsto a]}(G) = 1; \\ 0, & \text{különben.} \end{cases}$$

- ha $F = \forall x G$, akkor

$$\mathcal{A}(F) = \begin{cases} 1, & \text{ha } \textbf{bármely } a \in A\text{-ra } \mathcal{A}_{[x \mapsto a]}(G) = 1; \\ 0, & \text{különben.} \end{cases}$$

Ezután már a fenti képletek segítségével minden F formulára, annak felépítése szerinti indukcióval haladva, meg tudjuk határozni $\mathcal{A}(F)$-et. Amennyiben $\mathcal{A}(F) = 1$, azt mondjuk, hogy az F formula *igaz az $\mathcal{A}$ modellben*, másképp fogalmazva $\mathcal{A}$ *modellje F-nek*.
Nyilvánvaló, hogy az (elsőrendű) formulák értéke függ attól, hogy melyik modellben tekintjük őket. Ugyanaz a formula más-más modellben más-másképpen értékelődhet ki.
Például a

$$\forall x \exists y (x + y < x)$$

formula igaz, ha a modellben az alaphalmaz a valós számok halmaza($\mathbb{R}$), hiszen y lehet negatív. Viszont hamis, ha a modellben az alaphalmaz a természetes számok halmaza ($\mathbb{N}$) hiszen ekkor y nem lehet negatív. Persze ez csak akkor igaz, ha a fenti két modellben a "+" függvényszimbólum a szokásos összeadás függvény és a "<" predikátumszimbólum a szokásos rendezés. Amennyiben a "+" szimbólum interpretációja a szorzás és a "<" interpretációja a kisebb vagy egyenlő reláció, akkor a formula igaz mindkét modellben.
Az ítéletkalkulust, a predikátumkalkulus speciális eseteként úgy kaphatjuk meg, hogy kikötjük, hogy minden predikátumszimbólum 0-változós legyen. Ekkor az elsőrendű változók, függvényszimbólumok és a kvantorok feleslegessé válnak, elhagyhatjuk őket. A modell pedig a predikátumszimbólumok konstans 0, vagy 1 értékének meghatározására redukálódik. Ezért a (0-változós) predikátumszimbólumokat *ítéletváltozóknak* is hívjuk, a modellt pedig *változóhozzárendelésként* vagy *kiértékelésként* is említjük. Az ítéletkalkulus egy modellje theát egy $\mathcal{A} : Var \to \{0, 1\}$ leképezésként adható meg, ahol most $Var = \{p, q, r, \ldots\}$ az ítéletváltozók halmaza.
Fontos szemantikai alapfogalmak még az alábbiak. Egy formulát *kielégíthetőnek* nevezünk, ha van modellje. Egy formula *tautológia* (más néven *azonosan igaz formula* vagy *logikai törvény*), ha minden modellben igaz. Amennyiben egy F formula tautológia, a $\models F$ jelölést is hasznájuk. Egy formula *kielégíthetetlen*, ha egyetlen modellben sem igaz.
Formulák egy Σ halmazát akkor nevezzük *kielégíthetőnek*, ha létezik olyan modell, mely egyszerre a halmaz minden elemét igazzá teszi, azaz van olyan $\mathcal{A}$ modell, hogy minden $F \in \Sigma$-ra $\mathcal{A} \models F$ teljesül. Például a $\{\forall x \neg p(x), \exists x p(x)\}$ halmaz kielégíthetetlen, annak ellenére, hogy elemei külön-külön kielégíthetők. Kielégíthetőség szempontjából gyakran célszerű, ha a Σ halmazra úgy tekintünk,

mint axiómák halmazára, melyek mindegyikének teljesülnie kell a vizsgált modellekben.

Feladatok

2.1. Feladat. Az alábbi $\mathcal{A} = (A, I)$ struktúrák közül melyik modellje a $\exists x \exists y \exists z (p(x,y) \wedge p(z,y) \wedge p(x,z) \wedge \neg p(z,x))$ formulának?

a) $A = \{0, 1, 2, \ldots\} (= \mathbb{N}_0)$, minden $m, n \in \mathbb{N}_0$-ra $I(p)(m,n) = 1$ akkor és csak akkor, ha $m < n$.

b) $A = \mathbb{N}_0$, minden $m, n \in \mathbb{N}_0$-ra $I(p)(m,n) = 1$ akkor és csak akkor, ha $n = m + 1$.

c) $A = \mathcal{P}(\mathbb{N}_0)$, minden $H_1, H_2 \subseteq \mathbb{N}_0$-ra $I(p)(H_1, H_2) = 1$ akkor és csak akkor, ha $H_1 \subseteq H_2$.

2.1. Feladat megoldása.

a) Modellje. $\mathcal{A}_{[x \mapsto 1, y \mapsto 3, z \mapsto 2]}$ modellje a formula magjának, hiszen ekkor $\varphi(x) = 1 < \varphi(y) = 3$, $\varphi(z) = 2 < \varphi(y) = 3$ és $\varphi(x) = 1 < \varphi(z) = 2$, de nem $\varphi(z) = 2 < \varphi(x) = 1$.

b) Nem modellje. $x + 1 = y, z + 1 = y, x + 1 = z$ egyszerre nem teljesülhet.

c) Modellje. $\mathcal{A}_{[x \mapsto \{1\}, y \mapsto \{1,2,3\}, z \mapsto \{1,2\}]}$ modellje a formula magjának.

2.2. Feladat. Minden formulához adjunk meg egy olyan struktúrát, amely modellje és egy olyat, amely nem modellje a formulának!

a) $F = \forall x \forall y p(x, y, f(z))$;

b) $F = \forall x \forall y ((p(x,y) \wedge p(y,x)) \rightarrow x = y)$;

c) $F = \forall x \exists y (f(y) = x \wedge \neg \exists z (f(z) = x \wedge \neg (y = z)))$.

2.2. Feladat megoldása.

a) Legyen mondjuk $\mathcal{A}_1 = (\mathbb{N}_+, I, \varphi)$, ahol
$I(p) : \mathbb{N}_+^3 \to \{0,1\}, \quad I(p)(a,b,c) = \begin{cases} 1, & \text{ha } a+b \geq c \\ 0, & \text{különben,} \end{cases} \quad (\forall a,b,c \in \mathbb{N}_+\text{-ra})$
$I(f) : \mathbb{N}_+ \to \mathbb{N}_+, \quad I(f)(t) := t+1, \ (\forall t \in \mathbb{N}_+\text{-ra})$
$\varphi(z) = 1$, különben φ tetszőleges.

Ekkor $\forall x \forall y p(x, y, f(z)) =$ „$\forall x \forall y \in \mathbb{N}_+$-ra $x + y \geq 1 + 1$” igaz, azaz $\mathcal{A}_1 \models F$.

De, ha $\mathcal{A}_2 = (\mathbb{N}_+, I, \varphi')$ ugyanaz a modell mint $\mathcal{A}_1$, kivéve, hogy $\varphi'(z) = 13$, akkor „$\forall x \forall y \in \mathbb{N}_+$-ra $x + y \geq 13 + 1$” nem igaz. Ezért $\mathcal{A}_2 \not\models F$, azaz $\mathcal{A}_2$ nem modellje F-nek.

b) A formula a p-hez rendelt reláció antiszimmetrikus tulajdonságát fejezi ki. $\mathcal{A}_1 = (\mathbb{Z}, I, \varphi)$, ahol $I(p) : \mathbb{Z}^2 \to \{0,1\}, \quad I(p)(a,b) = \begin{cases} 1 \text{ ha } a \leq b \\ 0 \text{ különben} \end{cases}$ modellje a formulának, $\forall a, b \in \mathbb{Z}$-re, tetszőleges φ-re.

De $\mathcal{A}_2 = (\mathcal{P}(\mathbb{N}_0), I, \varphi)$, ahol $I(p) : \mathcal{P}(\mathbb{N}_0) \times \mathcal{P}(\mathbb{N}_0) \to \{0,1\}$,

$$I(p)(A,B) = \begin{cases} 1 & \text{ha } A \cap B \neq \emptyset \\ 0 & \text{különben ,} \end{cases} \quad \forall A, B \in \mathcal{P}(\mathbb{N}_0)\text{-ra, } \varphi \text{ tetszőleges.}$$

Nem modellje a formulának.

c) Az
$$F = \forall x \exists y (f(y) = x \wedge \neg \exists z (f(z) = x \wedge \neg (y = z)))$$
formula azt fejezik ki, hogy az f függvény bijektív függvény, azaz szürjektív (= minden elem képpé válik) és injektív (=különböző elemek képe is különböző).

Ezért egy modell akkor és csak akkor elégíti ki az F formulát, ha benne az f függvényt bijektív függvénynek interpretáljuk. Ha például A a sík pontjainak a halmaza, f interpretációja pedig egy adott egyenesre való tükrözés, akkor a formula egy modelljéhez jutunk. Ha azonban f interpretációja egy adott egyenesre való projekció (vetítés), akkor a struktúra nem modellje a formulának.

2.3. Feladat. Egyenlőséges logikában a predikátumszimbólumok között szerepel az $=$ jel, melynek intrepretációja mindig az az objektumok halmazán értelmezett egyenlőség reláció. Adjunk meg olyan egyenlőséges elsőrendű kielégíthető formulát, melynek...

a) ... minden modellje egyelemű.

b) ... minden modellje legfeljebb kételemű.

c) ... minden modellje legalább kételemű.

d) ... minden modellje legalább kételemű, és nincs benne egyenlőség.

e) ... minden modellje legalább háromelemű.

f) ... minden modellje legalább háromelemű, és nincs benne egyenlőség.

2.3. Feladat megoldása.

a) $\forall x \forall y (x = y)$, vagy $\forall x (x = c)$, ahol c konstans.

b) $\forall x \forall y \forall z ((x = y) \vee (x = z) \vee (y = z))$, vagy $\forall x ((x = c) \vee (x = d))$.

c) $\exists x \exists y \neg (x = y)$, vagy $\neg (c = d)$.

d) $\exists x \exists y (p(x) \wedge \neg p(y))$ vagy $p(c) \wedge \neg p(c)$.

e) $\exists x \exists y \exists z (\neg (x = y) \wedge \neg (x = z) \wedge \neg (y = z))$.

f) $\exists x \exists y \exists z (p(x) \wedge \neg p(y) \wedge \neg p(z) \wedge q(y) \wedge \neg q(z))$.

2.4. Feladat. Adjunk meg olyan elsőrendű formulát, melynek pontosan akkor modellje egy $\mathcal{A}$ struktúra, ha benne a p bináris predikátum interpretációja...

a) ... reflexív.

b) ... irreflexív.

c) ... tranzitív.

d) ... szimmetrikus.

e) ... aszimmetrikus.

f) ... antiszimmetrikus.

g) ... ekvivalenciareláció.

h) ... parciális rendezés.

i) ... szigorú parciális rendezés.

j) ... teljes rendezés.

k) ... szigorú teljes rendezés.

2.4. Feladat megoldása. A p binér reláció

a) ... reflexív: $\forall x p(x,x)$.

b) ... irreflexív: $\forall x \neg p(x,x)$.

c) ... tranzitív: $\forall x \forall y \forall z[(p(x,y) \wedge p(y,z)) \rightarrow p(x,z)]$.

d) ... szimmetrikus: $\forall x \forall y[p(x,y) \rightarrow p(y,x)]$.

e) ... aszimmetrikus: $\forall x \forall y \neg[p(x,y) \wedge p(y,x)]$.

f) ... antiszimmetrikus: $\forall x \forall y[(p(x,y) \wedge p(y,x)) \rightarrow (y = x)]$.

g) ... ekvivalenciareláció: reflexív + szimmetrikus + tranzitív: $\forall x p(x,x) \wedge \forall x \forall y[p(x,y) \rightarrow p(y,x)] \wedge \forall x \forall y \forall z[(p(x,y) \wedge p(y,z)) \rightarrow p(x,z)]$.

h) ... parciális rendezés: reflexív + antiszimmetrikus + tranzitív: $\forall x p(x,x) \wedge \forall x \forall y[(p(x,y) \wedge p(y,x)) \rightarrow (y = x)] \wedge \forall x \forall y \forall z[(p(x,y) \wedge p(y,z)) \rightarrow p(x,z)]$.

i) ... szigorú parciális rendezés: irreflexív + tranzitív: $\forall x \neg p(x,x) \wedge \forall x \forall y \forall z[(p(x,y) \wedge p(y,z)) \rightarrow p(x,z)]$;

Vagy az előbbivel ekvivalens módon, szigorú parciális rendezés: aszimmetrikus + tranzitív: $\forall x \forall y[p(x,y) \rightarrow \neg p(y,x)] \wedge \forall x \forall y \forall z[(p(x,y) \wedge p(y,z)) \rightarrow p(x,z)]$;

Könnyen látható, hogy a két definíció egymással ekvivalens. Ugyanis tegyük fel, hogy p tranzitív. Ekkor, ha p irreflexív, vagyis $\forall x \neg p(x,x)$, akkor szükségképpen aszimmetrikus is, hiszen, ha az aszimmetria, vagyis $\forall x \forall y[p(x,y) \rightarrow \neg p(y,x)]$ nem állna fenn, akkor ez azt jelentené, hogy valamely x-re és y-ra, $p(x,y)$ és $p(y,x)$ is igaz lenne, mert az implikáció csak úgy lehet hamis, ha előtagja igaz, utótagja viszont hamis. Ekkor azonban a tranzitivitás miatt $p(x,y)$ és $p(y,x)$-ből azt kapnánk, hogy $p(x,x)$, ami ellentmondana az irreflexivitásnak. Ezért az első definíció teljesülése esetén igaz a második is.

Hasonlóan, amennyiben a p reláció aszimmetrikus, azaz $\forall x \forall y[p(x,y) \rightarrow \neg p(y,x)]$, akkor ebbe a képletbe y helyére x-et helyettesítve azt kapjuk, hogy $\forall x[p(x,x) \rightarrow \neg p(x,x)]$. Ez utóbbi viszont csak akkor lehet igaz, ha $p(x,x)$ mindig hamis, vagyis a p irreflexív reláció. Ezért, ha a második definíció igaz a p relációra, akkor az első is igaz.

j) ... teljes rendezés: parciális rendezés + dichotóm (=bármely két eleme összehasonlítható): $\forall x p(x,x) \wedge \forall x \forall y[(p(x,y) \wedge p(y,x)) \rightarrow (y = x)] \wedge \forall x \forall y \forall z[(p(x,y) \wedge p(y,z)) \rightarrow p(x,z)] \wedge \forall x \forall y(p(x,y) \vee p(y,x))$.

k) ... szigorú teljes rendezés: szigorú parciális rendezés + trichotóm(=bármely két eleme egyenlő vagy összehasonlítható):
$\forall x \neg p(x,x) \wedge \forall x \forall y \forall z[(p(x,y) \wedge p(y,z)) \rightarrow p(x,z)] \wedge \forall x \forall y(p(x,y) \vee (x = y) \vee p(y,x))$.

2.5. Feladat. Adjunk meg olyan elsőrendű formulát, melynek pontosan akkor modellje egy $\mathcal{A}$ struktúra, ha benne a bináris f függvényszimbólum interpretációja...

a) ... kommutatív.

b) ... asszociatív.

c) ... idempotens.

d) ... kancellatív (balról és jobbról).

e) ... rendelkezik (jobb és baloldali) egységelemmel.

f) ... rendelkezik (jobb és baloldali) zéruselemmel.

2.5. Feladat megoldása.

a) ... kommutatív: $\forall x \forall y[f(x,y) = f(y,x)]$.

b) ... asszociatív: $\forall x \forall y \forall z[f(f(x,y),z) = f(x,f(y,z))]$.

c) ... idempotens: $\forall x[f(x,x) = x]$

d) ... kancellatív (balról és jobbról): $\forall x \forall y \forall z[(f(x,y) = f(x,z)) \rightarrow (y = z)] \wedge \forall x \forall y \forall z[(f(y,x) = f(z,x)) \rightarrow (y = z)]$.

e) ... rendelkezik (jobb és baloldali) egységelemmel: $\exists e \forall x[(f(e,x) = x) \wedge (f(x,e) = x)]$.

f) ... rendelkezik (jobb és baloldali) zéruselemmel: $\exists n \forall x[(f(n,x) = n) \wedge (f(x,n) = n)]$.

A tulajdonságokat könnyebb megjegyezni, ha a fentiekkel ellentétben a bináris műveletet infix módon jelöljük, azaz például $\star(x,y)$ helyett $x \star y$-t írunk. Ekkor

a) $\star$ kommutatív: $\forall x \forall y [x \star y = y \star x]$.

b) $\star$ asszociatív: $\forall x \forall y \forall z [(x \star y) \star z = x \star (y \star z)]$.

c) $\star$ idempotens: $\forall x [(x \star x) = x]$

d) $\star$ kancellatív (balról és jobbról): $\forall x \forall y \forall z [(x \star y = x \star z) \to (y = z)] \wedge \forall x \forall y \forall z [(y \star x = z \star x) \to (y = z)]$.

e) $\star$ rendelkezik (jobb és baloldali) egységelemmel: $\exists e \forall x [(e \star x = x) \wedge (x \star e = x)]$.

f) $\star$ rendelkezik (jobb és baloldali) zéruselemmel: $\exists n \forall x [(n \star x = n) \wedge (x \star n = n)]$.

2.6. Feladat. Adjunk meg olyan elsőrendű formulát, melynek pontosan akkor modellje egy $\mathcal{A}$ struktúra, ha benne az unáris f függvényjel interpretációja...

a) ... szürjektív.

b) ... injektív.

c) ... bijektív.

2.6. Feladat megoldása.

a) f szürjektív = minden elem képpé válik: $\forall y \exists x [f(x) = y]$.

b) f injektív = különböző elemek képe is különböző: $\forall x \forall y [\neg(x = y) \to \neg(f(x) = f(y))] \equiv \forall x \forall y [(f(x) = f(y)) \to (x = y)]$.

c) f bijektív = injektív + szürjektív: $\forall x \forall y [(f(x) = f(y)) \to (x = y)] \wedge \forall y \exists x [f(x) = y]$.

2.7. Feladat. Legyen $\mathcal{A} = (A, I)$,

$$F = \forall x \neg p(x, x) \wedge \forall x \forall y \forall z ((p(x, y) \wedge p(y, z)) \to p(x, z))$$

és $\mathcal{A} \models F$.
Tartalmazhat-e az $I(p)$ reláció gráfja az A halmazon kört? A választ indokoljuk! (Az $I(p)$ reláció gráfjában a csúcsok A elemei és tetszőleges a, $b \in A$ esetén a-ból b-be pontosan akkor vezet él, ha $I(p)(a, b)$ igaz.)

2.7. Feladat megoldása. (Vázlat.)
Indirekt módon igazolható. Ha $a_1, a_2, \ldots, a_n = a_1$ kör lenne $I(p)$ gráfjában, akkor a tranzitivitás többszöri felhasználásával azt kapnánk, hogy $I(p)(a_1, a_1)$ is teljesül, ellentmondva F-nek.

2.8. Feladat. Adjunk meg olyan kielégíthető formulát, amelynek minden modellje végtelen!

2.8. Feladat megoldása. Néhány lehetséges megoldás:

- A modellben szerepel egy f egyváltozós függvény, mely injektív, de nem szürjektív, ilyen függvény ugyanis véges halmaz felett nem létezhet. Formulával kifejezve:

$$\forall x \forall y (f(x) = f(y) \to x = y) \wedge \neg \forall y \exists x (f(x) = y)$$

- Az előző feladat F formulájához még adjuk hozzá, hogy $\wedge \forall x \exists y p(x, y)$
- Az előző feladat F formulájához még adjuk hozzá, hogy $\wedge \forall x p(x, f(x))$
- Lásd a 2.22. feladatot.

2.9. Feladat. Igazoljuk csak a szemantika definícióját felhasználva, hogy az alábbi formulák kielégíthetetlenek:

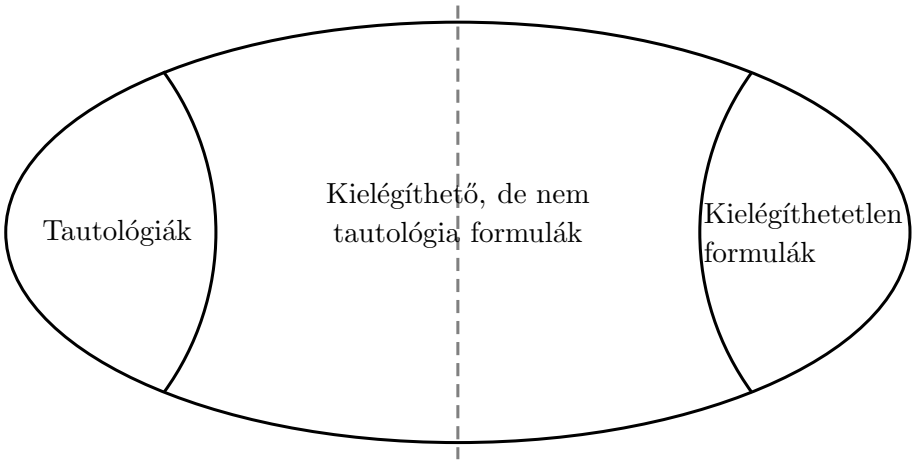
a) $\exists x \forall y (p(x, y) \leftrightarrow \neg p(y, y))$ (Russell-paradoxon);

b) $\exists x p(x) \wedge \forall x \neg p(x)$;

c) $\forall x p(x) \wedge \forall y (p(f(y)) \to q(y)) \wedge \exists z \neg q(z)$.

2.9. Feladat megoldása.

a) Indirekt módon tegyük fel, hogy valamely $\mathcal{A} = (A, I)$ modellre
$\mathcal{A} \models \exists x \forall y (p(x, y) \leftrightarrow \neg p(y, y))$
$\Leftrightarrow \exists a \in A : \mathcal{A}_{[x \mapsto a]} \models \forall y (p(x, y) \leftrightarrow \neg p(y, y))$
$\Leftrightarrow \exists a \in A, \forall b \in A : \mathcal{A}_{[x \mapsto a, y \mapsto b]} \models (p(x, y) \leftrightarrow \neg p(y, y))$
$\Leftrightarrow \exists a \in A, \forall b \in A : \mathcal{A}_{[x \mapsto a, y \mapsto b]}(p(x, y)) \neq \mathcal{A}_{[x \mapsto a, y \mapsto b]}(p(y, y))$
$\Leftrightarrow \exists a \in A, \forall b \in A : I(p)(a, b) \neq I(p)(b, b)$
$\Rightarrow$ Ez $b = a$ választás esetén: $I(p)(a, a) \neq I(p)(a, a)$ ellentmondás.



2.1. ábra. A formulák csoportosítása modelljeik száma szerint

b) Indirekt módon tegyük fel, hogy valamely $\mathcal{A} = (A, I)$ modellre
$\mathcal{A} \models \exists x p(x) \wedge \forall x \neg p(x)$

$\Leftrightarrow \exists a \in A : \mathcal{A}_{[x \mapsto a]} \models p(x)$ és $\forall b \in A : \mathcal{A}_{[x \mapsto b]} \models \neg p(x)$

$\Leftrightarrow \exists a \in A : I(p)(a) = 1$ és $\forall b \in A : I(p)(b) = 0$

De ez $b = a$ választás esetén ellentmondás.

c) Indirekt módon tegyük fel, hogy valamely $\mathcal{A} = (A, I)$ modellre $\mathcal{A} \models \forall x p(x) \wedge \forall y(p(f(y)) \to q(y)) \wedge \exists z \neg q(z)$

$\Leftrightarrow$

$$\forall a \in A : I(p)(a) = 1, \tag{2.1}$$
$$\forall b \in A : I(p)(I(f)(b)) = 1 \text{ esetén } I(q)(b) = 1 \text{ és} \tag{2.2}$$
$$\exists c \in A : I(q)(c) = 0. \tag{2.3}$$

Vegyük észre, hogy (1) miatt (2)-ben az $I(p)(I(f)(b)) = 1$ feltétel mindig teljesül, hiszen $I(f)(b)$ választható a-nak (1)-ben.

Így (1)-ből és (2)-ből kapjuk, hogy

$$\forall b \in A : I(q)(b) = 1, \tag{4}$$

Ez azonban ellentmond (3)-nak.

2.10. Feladat.
Döntsük el, hogy az alábbi formulák melyik kategóriába esnek: A) tautológiák, B) nem tautológiák, de kielégíthetők, C) kielégíthetetlenek. Ezeket a kategóriákat a 2.1. ábra szemlélteti. Vegyük észre, hogy a szaggatott vonal mentén az ábra „szimmetrikus a tagadásra", azaz bármely formula tagadása a formula helyének szaggatott vonalra vett tükörképén található.

a) $\exists x p(x) \vee \neg \exists x p(x)$;

b) $p(x) \wedge \neg p(y)$;

c) $\forall x p(x) \to \neg \exists x p(x)$;

d) $\forall x p(x) \to \neg \exists x \neg p(x)$;

e) $\forall x \exists y r(x, y) \to \exists x \forall y r(x, y)$;

f) $\exists x (p(x) \vee q(x)) \to \exists x p(x) \vee \exists x q(x)$;

g) $\forall x (p(x) \vee q(x)) \to \forall y p(y) \vee \forall z q(z)$;

h) $\neg [\forall x (p(x) \wedge q(x)) \to (\forall x p(x) \wedge \forall x q(x))]$;

i) $\forall x p(x) \to q(y) \leftrightarrow \forall x [p(x) \to q(y)]$;

j) $\exists x \forall y [r(x, y) \leftrightarrow \neg r(y, y)]$;

k) $\forall x \exists y [r(x, y) \leftrightarrow \neg r(y, y)]$;

l) $\forall x p(x) \wedge \forall y (p(f(y)) \to q(y)) \wedge \exists z \neg q(z)$.

2.10. Feladat megoldása. a) A; b) B; c) B; d) A; e) B; f) A; g) B; h) C; i) B; j) C; k) B; l) C.

2.11. Feladat. Igazoljuk, hogy tetszőleges F formulára fennállnak a következők:

a) Ha F kielégíthető, akkor $\exists x F$ is.

b) Ha $\exists x F$ kielégíthető, akkor F is.

c) Ha F tautológia, akkor $\forall x F$ is.

d) Ha $\forall x F$ tautológia, akkor F is.

2.11. Feladat megoldása.

a) F akkor és csak akkor elégíthető ki, ha létezik, olyan $\mathcal{A} = (A, I, \varphi)$ modell, melyre $\mathcal{A} \models F$. Legyen ekkor $a = \varphi(x)$. Így nyilván, $\mathcal{A} = \mathcal{A}_{[x \mapsto a]}$, hiszen ezzel a modellt nem változtattuk meg, mert x értéke változatlanul a. Összefoglalva, $\exists a \in A$, melyre $\mathcal{A}_{[x \mapsto a]} \models F$. Ez pedig pontosan $\exists x F$ kielégíthetőségének definíciója.

b) Ha $\exists x F$ kielégíthető, akkor definíció szerint létezik olyan $a \in A$, melyre $\mathcal{A}_{[x \mapsto a]} \models F$. Így az F-et igazzá tevő $\mathcal{A}_{[x \mapsto a]}$ modell létezése már bizonyítja F kielégíthetőségét.

c) F akkor és csak akkor tautológia, ha bármely $\mathcal{B}$ modellre $\mathcal{B} \models F$. Ez speciálisan az összes $\mathcal{A}_{[x \mapsto a]}$ alakú modellre is igaz. Vagyis bármely $\mathcal{A}$ modell esetén, bármely $a \in A$-ra $\mathcal{A}_{[x \mapsto a]} \models F$. Ez bármely $\mathcal{A}$ modell esetén definíció szerint azt jelenti, hogy $\mathcal{A} \models \forall x F$, vagyis $\forall x F$ is tautológia.

d) Tegyük fel, hogy $\forall x F$ tautológia, azaz bármely $\mathcal{A} = (A, I, \varphi)$ modell és $\forall a \in A$ esetén $\mathcal{A}_{[x \mapsto a]} \models F$. Vegyük észre, hogy tetszőleges $\mathcal{B} = (B, I', \varphi')$ modell előáll $\mathcal{A}_{[x \mapsto a]}$ alakban is, ha az $\mathcal{A} = (A, I, \varphi)$ modellt úgy választjuk meg, hogy $A = B$, $I = I'$, $\varphi = \varphi'$ és $a = \varphi'(x)$ teljesüljön. Így, mivel minden modellre $\mathcal{B} \models F$, valóban F is tautológia.

2.12. Feladat. Kielégíthetők-e az alábbi formulahalmazok?

a) $\{p, q, p \to r, \neg r\}$;

b) $\{p_1 \vee p_2, \neg p_2 \vee \neg p_3, p_3 \vee p_4, \neg p_4 \vee \neg p_5, \ldots\}$.

2.12. Feladat megoldása.

a) Nem. Ha az első kettő és az utolsó formula igaz egy $\mathcal{A}$ modellben, akkor szükségképpen $\mathcal{A}(p) = \mathcal{A}(q) = 1$ és $A(r) = 0$, de ekkor az implikáció definíciója miatt $\mathcal{A} \not\models p \to r$.

b) Igen. Legyen például $\mathcal{A}(p_i) = \begin{cases} 1 & \text{ha i páratlan} \\ 0 & \text{ha i páros} \end{cases}$ (vagy fordítva).

2.13. Feladat. Adjunk példát olyan három elemű Γ formulahalmazra, amely kielégíthetetlen, de minden két elemű részhalmaza kielégíthető! Általánosítsuk a példát n elemű halmazra is!

2.13. Feladat megoldása. Egy lehetséges megoldás:

$$\Gamma = \{p_1 \leftrightarrow p_2, p_2 \leftrightarrow p_3, p_3 \leftrightarrow \neg p_1\}.$$

Általánosítása n elemű formulahalmazra:

$$\Gamma = \{p_1 \leftrightarrow p_2, p_2 \leftrightarrow p_3, \ldots, p_{n-1} \leftrightarrow p_n, p_n \leftrightarrow \neg p_1\}.$$

Könnyen látható, hogy ezek a Γ halmazok a feladat feltételeinek megfelelnek.

2.14. Feladat. Legyen

$$\begin{aligned} F_1 &= \forall x \exists y\ p(x,y), \\ F_2 &= \exists y \forall x\ \neg p(x,y), \\ F_3 &= \forall y \forall x_1 \forall x_2 \Big(\big(p(x_1,y) \wedge p(x_2,y) \big) \rightarrow (x_1 = x_2) \Big). \end{aligned}$$

Igazoljuk, hogy $\{F_1, F_2, F_3\}$ kielégíthető, de nincs véges modellje!

2.14. Feladat megoldása. A három formulának egy közös modellje a nemnegatív egész számok halmaza, $\mathcal{A} = (\mathbb{N}_+, I)$, melyben a $p(x,y)$-t az „y eggyel nagyobb mint x” relációnak értelmezzük, vagyis

$$I(p)(a,b) = 1 \iff b = a + 1, \text{ bármely } a, b \in \mathbb{N}_+ \text{ esetén.}$$

Valóban ebben a modellben az első formula azt állítja, hogy minden nemnegatív egész számnál van eggyel nagyobb. A második szerint létezik olyan y, mégpedig a 0, mely semminél sem eggyel nagyobb. A harmadik szerint pedig, ha x_1 és x_2 rákövetkezője ugyanaz az y, akkor $x_1 = x_2$. Könnyen látható tehát, hogy ez a modell kielégíti mindhárom formulát.
Azt, hogy a három formulának nincs közös véges modellje, indirekt módon bizonyíthatjuk. Tegyük fel, hogy egy $F_1 \wedge F_2 \wedge F_3$-at kielégítő modellben csak véges sok, mondjuk n elem van. F_1 igazsága miatt minden x-hez létezik (legalább egy) olyan y, mellyel x a p reláció szerint relációban áll. Válasszunk minden x-hez egy ilyen y-t és nevezzük azt az x elem p-szerinti rákövetkezőjének. F_3 alapján azt is megállapíthatjuk, hogy minden y csak egyetlen x rákövetkezője lehet, ezért az n elemnek n különböző p szerinti rákövetkezője kell, hogy legyen. De ekkor F_2 már nem lehet igaz, mert F_2 azt állítja, hogy van olyan y, mely nem rákövetkezője egyetlen elemnek sem, nem lehet tehát mind az n elem valaminek a rákövetkezője. Mindhárom formula tehát nem lehet igaz egyetlen véges modellben sem.

2.15. Feladat. Igazoljuk csak a szemantika definícióját felhasználva, hogy az alábbi formulahalmaz kielégíthetetlen:

$$\begin{aligned}\Sigma \;= \{\;\; & \forall x\Big(p(x) \to \big(q(x) \vee r(x)\big)\Big),\\ & \forall y\big(q(y) \to \neg s(y)\big),\\ & \forall y\big(r(y) \to \neg s(y)\big),\\ & \exists z\big(p(f(z)) \wedge s(f(z))\big)\;\}.\end{aligned}$$

2.15. Feladat megoldása. Jelölje a Σ halmaz 4 formuláját a felírás sorrendjében F_1, F_2, F_3 és F_4. Indirekt úton tegyük fel, hogy létezik olyan $\mathcal{A} = (A, I)$ modell, mely e négy formula mindegyikét igazzá teszi. Ekkor $\mathcal{A} \models F_4$ miatt $\exists a \in A$: $\mathcal{A}_{[z\mapsto a]} \models \big(p(f(z) \wedge s(f(z))\big)$. Ez azt jelenti, hogy $\tilde{p}(\tilde{f}(a)) = 1$ és $\tilde{s}(\tilde{f}(a)) = 1$, ahol $\tilde{p}$ a p predikátum, $\tilde{f}$ pedig az f függvényszimbólum interpretációja az $\mathcal{A}$ modellben. Legyen a továbbiakban $b = \tilde{f}(a)$. Tudjuk tehát, hogy

$$\tilde{p}(b) = 1 \text{ és } \tilde{s}(b) = 1, \tag{2.4}$$

Így az F_1 formulából $x = b$ választással azt kapjuk, hogy

$$\tilde{q}(b) = 1 \text{ vagy } \tilde{r}(b) = 1, \tag{2.5}$$

ahol természetesen $\tilde{q}$ és $\tilde{r}$ a nekik megfelelő predikátumszimbólumok interpretációja a modellben. Ha most $\tilde{q}(b) = 1$, akkor az F_2 formula igazsága miatt $\neg\tilde{s}(b) = 1$, vagyis $\tilde{s}(b) = 0$. Ez azonban ellentmond (2.4)-nek.
Hasonlóan, ha (2.5)-ben $\tilde{q}(b) = 1$ helyett $\tilde{r}(b) = 1$ teljesül, akkor F_3-ból kapjuk, hogy $\tilde{s}(b) = 0$, ami szintén ellentmondás.
Mindenképpen ellentmondásra jutottunk, ezért az indirekt feltételezéssel ellentétben a Σ formulahalmaz kielégíthetetlen.

2.16. Feladat. Igaz-e, hogy

a) ... ha $F \to G$ tautológia és F tautológia, akkor G is tautológia;

b) ... ha $F \to G$ kielégíthető és F kielégíthető, akkor G is kielégíthető?

A választ indokoljuk!

2.16. Feladat megoldása.

a) Igaz. Indirekt módon:

Ha G nem tautológia, akkor $\exists\mathcal{A}$ modell, melyre $\mathcal{A}(G) = 0$.

Ekkor $\mathcal{A}(F \to G) = 0$, mert $\mathcal{A}(F) = 1$, hiszen F tautológia, és már tudjuk, hogy $\mathcal{A}(G) = 0$.

De ez ellentmond annak, hogy $F \to G$ tautológia.

b) Nem igaz.

Például $F = p$ és $G = \downarrow$ választással ennek a résznek az állítása nem teljesül: p és $p \to \downarrow$ kielégíthető, ellenben $\downarrow$ nyilván nem az.

A feladat a) részével szemben itt p és $p \to \downarrow$ két különböző modellben igaz. Így nem használhatjuk az implikáció értelmezését, hogy az előtag igazságából az utótag igazságára következtessünk, mint az a) részben, ahol a tautológiák minden modellben igazak.

2.17. Feladat. Legyenek F és G az *ítéletkalkulus* formulái. Tegyük fel, hogy $\models (F \to G)$, továbbá, hogy F-nek és G-nek *nincs közös ítéletváltozója*. Mutassuk meg, hogy ekkor F kielégíthetetlen vagy G tautológia! Mutassuk meg, hogy a bizonyításhoz szükséges feltenni, hogy ne legyen F-nek és G-nek közös ítéletváltozója!

2.17. Feladat megoldása. Tudjuk, hogy $\models F \to G \overset{\text{def.}}{\iff} \forall \mathcal{A} : Var \to \{0,1\}$ kiértékelésre $\mathcal{A}(F \to G) = 1 \Leftrightarrow \forall \mathcal{A}$ kiértékelésre ($\mathcal{A}(F) = 0$ vagy $\mathcal{A}(G) = 1$). Ekkor két eset lehetséges:

I. Ha most minden $\mathcal{A}$ kiértékelésre $\mathcal{A}(F) = 0$, akkor F kielégíthetetlen, és a feladat igazolásával kész vagyunk.

II. Különben van olyan $\mathcal{A}'$ kiértékelés, hogy $\mathcal{A}'(F) \neq 0$. Ekkor meg kell mutatnunk, hogy G tautológia.

Ennek belátásához vegyünk egy tetszőleges $\tilde{\mathcal{A}}$ kiértékelést. Mivel F-nek és G-nek nincs közös ítéletváltozója, készíthetünk egy olyan $\mathcal{B}$ kiértékelést, melyre

$$\mathcal{B}(p_i) = \begin{cases} \mathcal{A}'(p_i), & \text{ha } p_i \in Var(F) \\ \tilde{\mathcal{A}}(p_i), & \text{ha } p_i \in Var(G) \end{cases}$$

Ekkor $\mathcal{B}(F \to G) = 1$, mert $F \to G$ tautológia, de $\mathcal{B}(F) = \mathcal{A}'(F) = 1$, mert $\mathcal{B}$ és $\mathcal{A}'$ az F ítéletváltozóin megegyezik. Ezért az implikáció értelmezése miatt biztos, hogy $\mathcal{B}(G) = 1$ kell hogy teljesüljön.
Mivel $\tilde{\mathcal{A}}$ és $\mathcal{B}$ pedig G ítéletváltozóin azonos, azt kapjuk, hogy $\tilde{\mathcal{A}}(G) = \mathcal{B}(G) = 1$. Mivel ez tetszőleges $\tilde{\mathcal{A}}$ kiértékelésre teljesül, beláttuk, hogy G tautológia.

Végül megmutatjuk, hogy a ,,nincs közös ítéletváltozójuk" feltétel valóban szükséges. Például, $F = p \wedge q, G = p$ esetén $\models F \to G$ teljesül, de F kielégíthető és G nem tautológia.

2.18. Feladat. Az alábbi formulák közül melyek tautológiák? Állításunkat igazoljuk!

a) $\neg(p(x) \wedge p(y)) \to (\neg p(x) \vee \neg p(y))$.

b) $\forall x \forall y \forall z(p(x,x) \wedge (p(x,z) \to (p(x,y) \vee p(y,z)))) \to \exists y \forall z p(y,z)$.

c) $\exists x \forall y((p(x,y) \wedge \neg p(y,x)) \to (p(x,x) \leftrightarrow p(y,y)))$.

2.18. Feladat megoldása.

a) Tautológia.

$\neg(p(x) \wedge p(y)) \to (\neg p(x) \vee \neg p(y)) \equiv (\neg p(x) \vee \neg p(y)) \to (\neg p(x) \vee \neg p(y))$ a negációra és konjunkcióra vonatkozó De-Morgan azonosság alapján.

b) Nem tautológia.

Vegyük például az egész számok halmazát, és legyen p interpretációja a kisebb vagy egyenlő reláció. Ekkor nyilvánvaló, hogy $\forall x p(x,x)$ igaz. Nem nehéz látni, hogy $\forall x \forall y \forall z(p(x,z) \to (p(x,y) \vee p(y,z))$ is igaz, hiszen, ha $x \leq z$, akkor, amennyiben $z \leq y$, akkor $x \leq y$, ha viszont $z > y$, akkor $y \leq z$ teljesül. Mindezek ellenére nem igaz, hogy $\exists y \forall z p(y,z)$, mert $\mathbb{Z}$-nek nincs legkisebb eleme.

c) Nem tautológia.

Például a következő modellben nem igaz: $\mathcal{A} = (A, I)$, ahol $A = \{0,1,2,3\}$ és $\tilde{p} := I(p) : A \times A \to \{0,1\}$ a következő:

$$\tilde{p}(x,y) = \begin{cases} 1, \text{ ha } (y = x+1 \mod 4) \text{ vagy } (x = y \text{ és } x \text{ páratlan}) \\ 0, \text{ különben,} \end{cases},$$

minden $x, y \in \{0, \ldots, 3\}$-ra.

Valóban, ekkor bármely x esetén, legyen $y = x + 1 \mod 4$. Így $\tilde{p}(x,y) \wedge \neg\tilde{p}(y,x) = 1$, ezért az implikáció előtagja igaz. De $\tilde{p}(x,x) \leftrightarrow \tilde{p}(y,y) = 0$, ezért az implikáció utótagja hamis. Mivel minden x értékhez létezik ilyen y választás, összességében az egész formula hamis.

3. fejezet

Logikai következmények, ekvivalenciák

Elméleti összefoglaló

A szemantika 4 alapfogalma közül kettőt, a formulák és formulahalmazok *kielégíthetőségét* és a *tautológiák* definícióját az előző fejezet végén már ismertettük. A másik két fontos fogalom, nevezetesen a *logikai következmény* és a formulák *ekvivalenciájának* bevezetése, most következik.
Legyen Σ egy formulahalmaz, ekkor Σ modelljeinek a halmaza legyen

$$\mathrm{Mod}(\Sigma) := \{\mathcal{A} \mid \mathcal{A} \models \Sigma\}.$$

Egy Σ formulahalmaznak *logikai következménye* egy F formula, ha Σ minden modellje modellje F-nek is. Jelölése: $\Sigma \models F$. Röviden:

$$\Sigma \models F \;\Leftrightarrow\; \mathrm{Mod}(\Sigma) \subseteq \mathrm{Mod}(F).$$

Azt mondjuk, hogy az F és G formula *ekvivalens*, ha F *pontosan ugyanazokban a modellekben igaz*, mint G. Jele: $F \equiv G$. Röviden:

$$F \equiv G \;\Leftrightarrow\; \mathrm{Mod}(F) = \mathrm{Mod}(G).$$

Az ekvivalencia ($\equiv$) tehát a formulák halmazán értelmezett reláció, fontos, hogy ne tévesszük össze az elvivalencia logikai művelettel ($\leftrightarrow$), és a szöveges megfogalmazásokban két állítás közötti ,,akkor és csak akkor" viszonyt kifejező $\Leftrightarrow$ rövidítéssel.
E négy szemantikai alapfogalom között szoros kapcsolat van. Mindegyik visszavezethető a kielégíthetőségre, mert bármely F (elsőrendű vagy zérusrendű) formulára igazak az alábbi összefüggések. Ezek a fogalmak definíciójából közvetlenül levezethetők, néhány közülük a feladatokban is szerepel.

- $\models F$, azaz tautológia $\Leftrightarrow \neg F$ kielégíthetetlen;
- $F \equiv G \Leftrightarrow \models F \leftrightarrow G \Leftrightarrow \neg(F \leftrightarrow G)$ kielégíthetetlen;
- $\Sigma \models F \Leftrightarrow \Sigma \cup \{\neg F\}$ kielégíthetetlen.

Ezért elegendő csupán a kielégíthetőség-kielégíthetetlenség kérdés eldöntésére algoritmust adnunk ahhoz, hogy a másik három fogalomhoz tartozó tulajdonságokat eldöntsük. A kielégíthetetlenség sokszor indirekt módon igazolható, és kielégíthetetlenség igazolására alkalmas módszer lesz a később ismertetendő rezolúció.

Feladatok

3.1. Feladat. Igazoljuk, hogy tetszőleges F, G, H formulákra fennállnak a következők:

a) Ha $\models F \to G$, akkor $F \models G$.

b) Ha $F \models G$, akkor $\models F \to G$.

c) Ha $F \models\downarrow$, akkor F kielégíthetetlen.

d) Ha F kielégíthetetlen, akkor $F \models\downarrow$.

e) Ha $\uparrow\models F$, akkor F tautológia.

f) Ha F tautológia, akkor $\uparrow\models F$.

g) Ha $F \models G$ és G kielégíthetetlen, akkor F is.

h) Ha $F \models G$ és F tautológia, akkor G is.

i) Ha $F \models G$ és $G \models H$, akkor $F \models H$.

j) Ha $F \models G$ és $F \models H$, akkor $F \models G \wedge H$.

k) Ha $F \models G$ és F kielégíthető, akkor G is.

l) Ha $(F \wedge G) \models H$, akkor $F \models (G \to H)$.

m) Ha $F \models (G \to H)$, akkor $(F \wedge G) \models H$.

n) $F \models \exists x F$.

o) $\forall x F \models F$.

p) Ha $\models F$, akkor $\emptyset \models F$.

q) Ha $\emptyset \models F$, akkor $\models F$.

3.1. Feladat megoldása. Az alábbi bizonyítások egy-egy speciális gondolatmenetet követnek, számos más megoldás is létezik, például a definíciók másféle kifejtése szerint vagy indirekt módon okoskodva.

a) Definíció szerint $\models F \to G$ azt jelenti, hogy $\forall A$ modellre $\mathcal{A}(F \to G) = 1$. Ez az implikáció értelmezése szerint annyit tesz, hogy $\forall A$ modellre, ha $\mathcal{A}(F) = 1$, akkor $\mathcal{A}(G) = 1$. Ez utóbbi viszont pontosan $F \models G$ definíciója.

b) Végezzük az előző rész bizonyítási lépéseit fordított sorrendben!

c) $F \models\downarrow$ pontosan akkor teljesül, ha $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(\downarrow)$. De $\downarrow$ azonosan hamis, ezért $\mathrm{Mod}(\downarrow) = \emptyset$. Így $\mathrm{Mod}(F) = \emptyset$, ami pontosan azt jelenti, hogy F kielégíthetetlen.

d) Végezzük az előző rész bizonyítási lépéseit fordított sorrendben!

e) $\uparrow\models F$, azt jelenti, hogy $\forall A$ modellre, ha $\mathcal{A} \models\uparrow$, akkor $\mathcal{A} \models F$. De $\uparrow$ azonosan igaz, ezért $\mathcal{A} \models\uparrow$ minden $\mathcal{A}$ modellre teljesül, így $\mathcal{A} \models F$ is igaz minden $\mathcal{A}$ modellre, vagyis F tautológia.

f) Végezzük az előző rész bizonyítási lépéseit fordított sorrendben!

g) $F \models G$ akkor és csak akkor, ha $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(G)$. Mivel viszont G kielégíthetetlen, azért $\mathrm{Mod}(G) = \emptyset$. Ezért szükségképpen $\mathrm{Mod}(F) = \emptyset$ is teljesül, vagyis F is kielégíthetetlen.

h) Ha $F \models G$, akkor $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(G)$. Mivel viszont F tautológia, $\mathrm{Mod}(F)$ az összes modell halmaza, ezért $\mathrm{Mod}(G)$ is az összes modell halmaza, vagyis G is tautológia.

i) A logikai következmény modellek halmazával felírt definícióját használva, $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(G)$ és $\mathrm{Mod}(G) \subseteq \mathrm{Mod}(H)$ valóban maga után vonja $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(H)$-t.

j) A konjunkció értelmezése miatt $G \wedge H$ modelljeinek halmaza pontosan G és H közös modelljeiből áll, azaz $\mathrm{Mod}(G \wedge H) = \mathrm{Mod}(G) \cap \mathrm{Mod}(H)$. Ezt felhasználva $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(G)$ és $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(H)$-ból már könnyen látható, hogy $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(G) \cap \mathrm{Mod}(H) = \mathrm{Mod}(G \wedge H)$, mely pontosan azt jelenti, hogy $F \models G \wedge H$.

k) $F \models G$ értelmében F minden modellje G-nek is modellje. Viszont, mivel F kielégíthető, F-nek van modellje. Ezért az G-nek is modellje, tehát G is kielégíthető.

l) Ha $(F \wedge G) \models H$, akkor $F \models (G \to H)$. Megmutatjuk, hogy $\mathrm{Mod}(F) \subseteq \mathrm{Mod}(G \to H)$. Ehhez vegyünk egy tetszőleges $\mathcal{A}$ modellt, melyre $\mathcal{A}(F) = 1$. Ha most $\mathcal{A}(G) = 0$, akkor $\mathcal{A}(G \to H) = 1$, mert ha az előtagja hamis, az implikáció biztosan igaz. Ha viszont $\mathcal{A}(G) = 1$, akkor $(F \wedge G) \models H$ miatt azt kapjuk, hogy $\mathcal{A}(H) = 1$, és ekkor is $\mathcal{A}(G \to H) = 1$, mert ha az utótagja igaz, az implikáció szintén biztosan igaz.

m) Bizonyítsuk ezt indirekt módon. Ha $(F \wedge G) \models H$ nem teljesül, akkor létezik olyan $\mathcal{A}$ modell, melyre $\mathcal{A}(F) = 1$, $\mathcal{A}(G) = 1$, de $\mathcal{A}(H) = 0$. Ám ekkor ebben a modellben $\mathcal{A}(F) = 1$, de $\mathcal{A}(G \to H) = 0$ (mert igaz előtagból hamis utótag nem következik), ami ellentmond annak, hogy $F \models (G \to H)$.

n) Legyen $\mathcal{A} = (A, I, \varphi)$ egy tetszőleges modell, melyre $\mathcal{A}(F) = 1$, Ekkor $a = \varphi(x)$ választással $\mathcal{A}_{[x \mapsto a]} = \mathcal{A} \models F$ alapján láthatjuk, hogy a $\exists$ kvantor definíciójában megkövetelt feltétel teljesül, ezért $\mathcal{A} \models \exists x F$ is igaz.

o) Az előző részhez hasonlóan, közvetlenül a definíciók kifejtésével bizonyítható.

p) Ha $\models F$, akkor $\mathrm{Mod}(F)$ az összes modell halmaza, $\mathrm{Mod}(\emptyset)$ pedig mindig az összes modell halmaza, ezért $\mathrm{Mod}(\emptyset) \subseteq \mathrm{Mod}(F)$, mely azt jelenti, hogy $\emptyset \models F$.

q) Az előző részhez hasonlóan bizonyítható.

3.2. Feladat.
Legyen Γ és Δ két formulahalmaz, és F és G legyenek formulák. Bizonyítsuk be a következőket!

a) Ha $\Gamma \models F$ és $\Gamma \subseteq \Delta$, akkor $\Delta \models F$;

b) $\Gamma \cup \{F\} \models G$ akkor és csak akkor, ha $\Gamma \models F \to G$.

c) Ha $\Gamma \cup \{\neg F\} \models G$ és $\Gamma \cup \{\neg F\} \models \neg G$, akkor $\Gamma \models F$;

d) Ha $\Gamma \cup \{F\} \models H$ és $\Gamma \cup \{G\} \models H$, akkor $\Gamma \cup \{F \vee G\} \models H$;

e) Ha $\Gamma \models F$ és $\Delta \models \neg F$, akkor $\Gamma \cup \Delta$ kielégíthetetlen.

3.2. Feladat megoldása.

a) Triviális. Ha Δ minden formulája igaz, akkor Γ-é is. Ezért, bármely $\mathcal{A}$ modellre, ha $\mathcal{A} \models \Delta$, akkor $\Gamma \models F$ miatt $\mathcal{A} \models F$, ami éppen $\Delta \models F$ definíciója.

b) *A szükségesség bizonyítása* (baloldalból következik a jobboldal): Tegyük fel, hogy $\Gamma \cup \{F\} \models G$, meg kell mutatnunk, hogy ekkor $\Gamma \models F \to G$.

Ez utóbbit definíció szerint igazolhatjuk. Tetszőleges $\mathcal{A}$ modellre, ha $\mathcal{A} \models \Gamma$ akkor vagy

I. eset: $\mathcal{A}(F) = 0 \quad \Rightarrow \quad \mathcal{A}(F \to G) = 1$, vagy,

II. eset: $\mathcal{A}(F) = 1 \quad \Rightarrow \quad \mathcal{A} \models \Gamma \cup \{F\}$, és így $\Gamma \cup \{F\} \models G$ miatt $\mathcal{A}(G) = 1$. Ezért szintén $\mathcal{A}(F \to G) = 1$.

Az elégségesség bizonyítása (jobboldalból következik a baloldal): Tegyük fel, hogy $\Gamma \models F \to G$, meg kell mutatnunk, hogy ekkor $\Gamma \cup \{F\} \models G$.

Ennek belátásához vegyünk egy tetszőleges $\mathcal{A}$ modellt, melyre $\mathcal{A} \models \Gamma \cup \{F\}$. Ez azt jelenti, hogy $\mathcal{A} \models \Gamma$ és $\mathcal{A}(F) = 1$. Mivel $\mathcal{A} \models \Gamma$, a $\Gamma \models F \to G$ feltételből azt kapjuk, hogy $\mathcal{A}(F \to G) = 1$, mely az implikáció definíciója és $\mathcal{A}(F) = 1$ miatt arra vezet, hogy $\mathcal{A}(G) = 1$, melyet igazolnunk kellett.

c) Tudjuk, hogy $\Gamma \models F \Leftrightarrow \Gamma \cup \{\neg F\}$ kielégíthetetlen. Ez utóbbi állítás indirekt módon bizonyíthatjuk: Tegyük fel, hogy $\Gamma \cup \{\neg F\}$ kielégíthető, azaz létezik olyan $\mathcal{A}$ modell, melyre $\mathcal{A} \models \Gamma \cup \{\neg F\}$. Ha most $\mathcal{A}(G) = 0$, akkor $\Gamma \cup \{\neg F\} \models G$ nem teljesül. Ha viszont $\mathcal{A}(G) = 1$, akkor $\Gamma \cup \{\neg F\} \models \neg G$ nem teljesül. Mindenképpen ellentmondásra jutottunk, ezért $\Gamma \cup \{\neg F\}$ kielégíthetetlen, és ezért $\Gamma \models F$.

d) Definíció szerint bizonyítva, minden $\mathcal{A}$ modellre, ha $\mathcal{A} \models \Gamma \cup \{F \vee G\}$
$\Rightarrow \mathcal{A} \models \Gamma$ és $\mathcal{A}(F \vee G) = 1$
$\Rightarrow \mathcal{A} \models \Gamma$ és ($\mathcal{A}(F) = 1$ vagy $\mathcal{A}(G) = 1$)
$\Rightarrow$ ($\mathcal{A} \models \Gamma$ és $\mathcal{A}(F) = 1$) vagy ($\mathcal{A} \models \Gamma$ és $\mathcal{A}(G) = 1$).
Mindkét esetben a feltételek miatt $\mathcal{A}(H) = 1$, melyet igazolnunk kellett.

e) Indirekt úton tegyük fel, hogy $\Gamma \cup \Delta$ kielégíthető. Ez azt jelenti, hogy létezik olyan $\mathcal{A}$ modell, melyre $\mathcal{A} \models \Gamma$ és $\mathcal{A} \models \Delta$. Ekkor két eset lehetséges:

I.eset: Ha $\mathcal{A}(F) = 0$, akkor $\Gamma \models F$ nem teljesül.

II.eset: Ha $\mathcal{A}(F) = 1$, akkor $\Delta \models \neg F$ nem teljesül.

Mindkét eset ellentmond a feltételeknek, ezért $\Gamma \cup \Delta$ kielégíthetetlen, melyet bizonyítani kellett.

3.3. Feladat. Igazoljuk, hogy tetszőleges Σ, Δ, Γ formulahalmazokra, valamint F, G, H formulákra fennállnak a következők:

a) $\mathrm{Mod}(\Sigma \cup \Delta) = \mathrm{Mod}(\Sigma) \cap \mathrm{Mod}(\Delta)$.

b) $\Sigma \models \Delta, \Sigma \models \Gamma \;\Rightarrow\; \Sigma \models \Delta \cup \Gamma$.

c) $\Sigma \models \Delta \;\Rightarrow\; \Sigma \models \Delta \cap \Gamma$.

d) $\Sigma \models \Delta, \Delta \models \Gamma \;\Rightarrow\; \Sigma \models \Gamma$.

e) $\Sigma \models \Delta \;\Rightarrow\; \Sigma \cup \Gamma \models \Delta$.

f) $\mathrm{Mod}(\emptyset) = \mathrm{Mod}(F) \cup \mathrm{Mod}(\neg F)$.

g) $\mathrm{Mod}(\uparrow) = \mathrm{Mod}(\emptyset)$.

h) $\mathrm{Mod}(\downarrow) = \mathrm{Mod}(F) \cap \mathrm{Mod}(\neg F)$.

i) $\mathrm{Mod}(\downarrow) = \mathrm{Mod}(\{F, \neg F\})$.

j) $\{F, F \to G\} \models G$.

k) $\{F, \neg G \to \neg F\} \models G$.

l) $\{F \vee G, \neg F \vee H\} \models G \vee H$.

m) Ha $\Sigma \models F$, akkor $\Sigma \cup \{\neg F\}$ kielégíthetetlen.

n) Ha $\Sigma \cup \{\neg F\}$ kielégíthetetlen, akkor $\Sigma \models F$.

o) Ha $F \equiv G$, akkor $\models F \leftrightarrow G$.

p) Ha $\models F \leftrightarrow G$, akkor $F \equiv G$.

q) Ha $\Gamma \cup \{F \to G\} \models\downarrow$, akkor $\Gamma \cup \{\neg F\} \models\downarrow$ és $\Gamma \cup \{G\} \models\downarrow$.

3.3. Feladat megoldása. Az állítások az előző feladathoz hasonlóan közvetlenül a definíciók alapján vagy indirekt módon bizonyíthatók.

3.4. Feladat. Igazoljuk a következőket:

a) $\models$ reflexív reláció.

b) $\models$ tranzitív reláció.

c) $\models$ *nem* szimmetrikus reláció.

d) $\models$ *nem* antiszimmetrikus reláció.

e) $\models$ előrendezés.

f) $\equiv$ reflexív reláció.

g) $\equiv$ tranzitív reláció.

h) $\equiv$ szimmetrikus reláció.

i) $\equiv$ *nem* antiszimmetrikus reláció.

j) $\equiv$ ekvivalenciareláció.

A $\models$ és $\equiv$ relációkat mind a formulák halmazán, mind a formulahalmazok halmazán tekinthetjük.

3.4. Feladat megoldása. A pozitív állítások igazolásához legegyszerűbb a

$$\Sigma \models \Delta \iff \mathrm{Mod}(\Sigma) \subseteq \mathrm{Mod}(\Delta)$$

és a

$$\Sigma \equiv \Delta \iff \mathrm{Mod}(\Sigma) = \mathrm{Mod}(\Delta)$$

jellemzésekre, valamint a $\subseteq$ és $=$ halmazműveletek hasonló tulajdonságaira hivatkozni. A negatív állításokhoz pedig azt vegyük észre, hogy $\mathrm{Mod}(\Sigma) = \mathrm{Mod}(\Delta)$ abban az esetben is teljesülhet, ha $\Sigma \neq \Delta$.
Részletesebben

a) $\Sigma \models \Sigma$, mert $\mathrm{Mod}(\Sigma) \subseteq \mathrm{Mod}(\Sigma)$, hiszen $\subseteq$ reflexív.

b) Könnyen látható, mert $\subseteq$ tranzitív reláció.

c) Egy ellenpélda például, p és $\uparrow$, mert $p \models \uparrow$, de $\uparrow \not\models p$.

d) Egy ellenpélda például, $p \wedge q$ és $q \wedge p$, mert $p \wedge q \models q \wedge p$, de $p \wedge q \neq q \wedge p$, mint azt az antiszimmetria megkövetelné. (Formulák csak akkor egyenlőek, ha mint jelsorozatok, azaz a jelkészlet feletti szavak megegyeznek.)

e) Mert ez definíció szerint azt jelenti, hogy $\models$ reflexív és tranzitív.

f) Mert $=$ reflexív reláció.

g) Mert $=$ tranzitív reláció.

h) Mert $=$ szimmetrikus reláció.

i) Mint korábban, $p \wedge q \equiv q \wedge p$, de $p \wedge q \neq q \wedge p$.

j) Mert ez definíció szerint azt jelenti, hogy $\equiv$ reflexív, szimmetrikus és tranzitív.

3.5. Feladat. Igazoljuk az alábbiakat:

a) $\forall x p(x) \models p(0)$.

b) $p(0) \models \exists x p(x)$.

c) $(\forall x p(x)) \wedge (\forall y(p(y) \to q(y))) \models \forall x q(x)$.

d) $\forall x(p(x) \leftrightarrow q(x)) \models (\forall x p(x)) \leftrightarrow (\forall x q(x))$.

e) $\exists y \forall x p(x, y) \models \forall x \exists y p(x, y)$.

f) $\exists x(p(x) \wedge q(x)) \models (\exists x p(x)) \wedge (\exists x q(x))$.

g) $(\exists x p(x)) \wedge (\forall y(p(y) \to q(y))) \models \exists x q(x)$.

h) $\forall x \forall y((0 + x = y) \to (x = y)) \models 0 + 0 = 0$.

i) $\exists x p(f(x)) \wedge \forall y(p(y) \to q(y)) \models \exists z q(z)$.

j) $\exists z p(z, c) \wedge \forall x \forall y(p(x, y) \to p(y, x)) \models \exists x p(c, x)$.

k) $\exists z p(z, c) \wedge \forall x \forall y(p(x, y) \to p(y, f(x))) \models \exists x p(c, x)$.

l) $\forall x \forall y(p(x, f(x), y) \to p(f(x), x, y)) \models \forall y(p(c, f(c), y) \to p(f(c), c, y))$.

m) $\forall x \forall y(p(x, f(x), y) \to p(f(x), x, y)) \models \big(\exists y p(c, f(c), y)\big) \to (\exists y p(f(c), c, y))$.

n) $\exists x(p(x) \vee q(x)) \models \big(\forall x \neg p(x)\big) \to \exists y q(y)$.

o) $\exists x(p(x) \to q(x)) \models \big(\forall x p(x)\big) \to \exists y q(y)$.

p) $\forall x \exists y\big(p(x, y) \vee p(y, y)\big) \wedge \forall x(\neg p(x, x)) \models \big(\forall x \exists y p(x, y)\big)$.

q) $\forall x \exists y\big(p(x, y) \vee p(y, y)\big) \models \big(\exists x \forall y \neg p(x, y)\big) \to \exists x p(x, x)$.

3.5. Feladat megoldása.

a) $\forall x p(x) \models p(0)$:

$\forall x p(x) \models p(0) \Leftrightarrow \{\forall x p(x), \neg p(0)\}$ kielégíthetetlen.

Indirekt úton tegyük fel, hogy van olyan $\mathcal{A} = (A, I)$ modell, melyre $\mathcal{A} \models \forall x p(x)$ és $\mathcal{A} \models \neg p(0)$.

Ekkor

(1) $\mathcal{A} \models \forall x p(x) \Leftrightarrow \forall a \in A$-ra $\mathcal{A}_{[x \mapsto a]} \models p(x) \Leftrightarrow \forall a \in A$-ra $\tilde{p}(a) = 1$, azaz $\tilde{p}$ azonosan igaz predikátum.

Ugyanakkor

(2) $\mathcal{A} \models \neg p(0) \Leftrightarrow \neg \tilde{p}(\tilde{0}) = 1 \Leftrightarrow \tilde{p}(\tilde{0}) = 0$, azaz az $\tilde{p}$ predikátum hamis $\tilde{0}$-on.

Mivel (1) ellentmond (2)-nek, a $\forall x p(x) \models p(0)$ logikai következtetés igaz.

b) $p(0) \models \exists x p(x)$:

$p(0) \models \exists x p(x) \Leftrightarrow \{p(0), \neg \exists x p(x)\}$ kielégíthetetlen.

Indirekt módon tegyük fel, hogy, $\exists \mathcal{A} = (A, I)$ modell, melyre $\mathcal{A} \models p(0)$ és $\mathcal{A} \models \neg \exists x p(x)$.

Ekkor

(1) $\mathcal{A} \models p(0) \Leftrightarrow \tilde{p}(\tilde{0}) = 1$.

Ugyanakkor

(2) $\mathcal{A} \models \neg \exists x p(x) \Leftrightarrow \mathcal{A} \not\models \exists x p(x) \Leftrightarrow$ Nem igaz, hogy $\exists a \in A : \tilde{p}(a) = 1 \Leftrightarrow \forall a \in A : \tilde{p}(a) = 0$.

Mivel azonban (2)-ben az $a = \tilde{0}$ választás is lehetséges, (1) ellentmond (2)-nek, így a $p(0) \models \exists x p(x)$ logikai következtetés igaz.

c) $(\forall x p(x)) \wedge (\forall y (p(y) \to q(y))) \models \forall x q(x)$:

Tegyük fel, hogy $(\forall x p(x)) \wedge (\forall y (p(y) \to q(y))) \not\models \forall x q(x)$, vagyis $\mathrm{Mod}((\forall x p(x)) \wedge (\forall y (p(y) \to q(y)))) \not\subseteq \mathrm{Mod}(\forall x q(x))$.

Ekkor van olyan $\mathcal{A} = (A, I, \varphi)$, amire $\mathcal{A}(\forall x p(x)) \wedge (\forall y (p(y) \to q(y))) = 1$ és $\mathcal{A}(\forall x q(x)) = 0$.

$\mathcal{A}(\forall x p(x)) \wedge (\forall y (p(y) \to q(y))) = 1$-ből kapjuk az $\wedge$ szemantikája szerint, hogy $\mathcal{A}(\forall x p(x)) = 1$ és $\mathcal{A}(\forall y (p(y) \to q(y))) = 1$.

$\mathcal{A}(\forall x q(x)) = 0$-ból kapjuk $\forall x$ szemantikája szerint, hogy van olyan $a \in A$ elem, amire $\mathcal{A}_{[x \mapsto a]}(q(x)) = 0$, vagyis $\tilde{q}(a) = 0$.

$\mathcal{A}(\forall x p(x)) = 1$-ből kapjuk $\forall x$ szemantikája szerint, hogy tetszőleges $b \in A$ elemre $\mathcal{A}_{[x \mapsto b]}(p(x)) = 1$, vagyis $\tilde{p}(b) = 1$.

Speciálisan $b := a$-ra is, vagyis $\tilde{p}(a) = 1$.

$\mathcal{A}(\forall y(p(y) \to q(y))) = 1$-ből kapjuk $\forall y$ szemantikája szerint, hogy tetszőleges $c \in A$ elemre $\mathcal{A}_{[y \mapsto c]}(p(y) \to q(y)) = 1$.

Ezt tovább fejtve a $\to$ szemantikája szerint kapjuk, hogy tetszőleges $c \in A$ elemre $\mathcal{A}_{[y \mapsto c]}(p(y)) = 0$ vagy $\mathcal{A}_{[y \mapsto c]}(q(y)) = 1$, vagyis $\tilde{p}(c) = 0$ vagy $\tilde{q}(c) = 1$.

Speciálisan $c := a$-ra is, vagyis $\tilde{p}(a) = 0$ vagy $\tilde{q}(a) = 1$.

Azt tudjuk, hogy $\tilde{p}(a) = 1$, ezzel kombinálva a fenti összefüggést azt kapjuk, hogy $\tilde{q}(a) = 1$, ami ellentmondás, tehát az eredeti állítás igaz, a következmény fennáll.

d) $\forall x(p(x) \leftrightarrow q(x)) \models (\forall x p(x)) \leftrightarrow (\forall x q(x))$:

Tegyük fel, hogy $\forall x(p(x) \leftrightarrow q(x)) \not\models (\forall x p(x)) \leftrightarrow (\forall x q(x))$, vagyis $\mathrm{Mod}(\forall x(p(x) \leftrightarrow q(x))) \not\subseteq \mathrm{Mod}((\forall x p(x)) \leftrightarrow (\forall x q(x)))$.

Ekkor van olyan $\mathcal{A} = (A, I, \varphi)$, amire $\mathcal{A}(\forall x(p(x) \leftrightarrow q(x))) = 1$ és $\mathcal{A}((\forall x p(x)) \leftrightarrow (\forall x q(x))) = 0$.

$\mathcal{A}((\forall x p(x)) \leftrightarrow (\forall x q(x))) = 0$-ból a $\leftrightarrow$ szemantikája szerint kapjuk, hogy két eset lehetséges: vagy $\mathcal{A}(\forall x p(x)) = 1$ és $\mathcal{A}(\forall x q(x)) = 0$, vagy fordítva. Mindkét esetet megnézzük külön-külön és mindkettőben ellentmondásra jutunk.

- Ha $\mathcal{A}(\forall x p(x)) = 1$ és $\mathcal{A}(\forall x q(x)) = 0$, akkor utóbbiból a $\forall x$ szemantikája szerint kapjuk, hogy van olyan $b \in A$, amire $\mathcal{A}_{[x \mapsto b]}(q(x)) = 0$, vagyis $\tilde{q}(b) = 0$.

  $\mathcal{A}(\forall x p(x)) = 1$-ből a $\forall x$ szemantikája szerint kapjuk, hogy tetszőleges $c \in A$-ra $\mathcal{A}_{[x \mapsto c]}(p(x)) = 1$, vagyis $\tilde{p}(c) = 1$. Speciálisan $c := b$-re is, vagyis $\tilde{p}(b) = 1$.

  Ugyanakkor $\mathcal{A}(\forall x(p(x) \leftrightarrow q(x))) = 1$-ből a $\forall x$ szemantikája szerint kapjuk, hogy tetszőleges $a \in A$ elemre $\mathcal{A}_{[x \mapsto a]}(p(x) \leftrightarrow q(x)) = 1$.

  Ezt tovább fejtve a $\leftrightarrow$ szemantikája szerint kapjuk, hogy tetszőleges $a \in A$ elemre $\mathcal{A}_{[x \mapsto a]}(p(x)) = \mathcal{A}_{[x \mapsto a]}(q(x))$, vagyis $\tilde{p}(a) = \tilde{q}(a)$. Speciálisan $a := b$-re is, vagyis $\tilde{p}(b) = \tilde{q}(b)$, ez ellentmondás.

- Ha $\mathcal{A}(\forall x p(x)) = 0$ és $\mathcal{A}(\forall x q(x)) = 1$, akkor előbbiből a $\forall x$ szemantikája szerint kapjuk, hogy van olyan $b \in A$, amire $\mathcal{A}_{[x \mapsto b]}(p(x)) = 0$, vagyis $\tilde{p}(b) = 0$.

  $\mathcal{A}(\forall x q(x)) = 1$-ből a $\forall x$ szemantikája szerint kapjuk, hogy tetszőleges $c \in A$-ra $\mathcal{A}_{[x \mapsto c]}(q(x)) = 1$, vagyis $\tilde{q}(c) = 1$. Speciálisan $c := b$-re is, vagyis $\tilde{q}(b) = 1$.

  Ugyanakkor $\mathcal{A}(\forall x(p(x) \leftrightarrow q(x))) = 1$-ből a $\forall x$ szemantikája szerint kapjuk, hogy tetszőleges $a \in A$ elemre $\mathcal{A}_{[x \mapsto a]}(p(x) \leftrightarrow q(x)) = 1$.

Ezt tovább fejtve a $\leftrightarrow$ szemantikája szerint kapjuk, hogy tetszőleges $a \in A$ elemre $\mathcal{A}_{[x\mapsto a]}(p(x)) = \mathcal{A}_{[x\mapsto a]}(q(x))$, vagyis $\tilde{p}(a) = \tilde{q}(a)$. Speciálisan $a := b$-re is, vagyis $\tilde{p}(b) = \tilde{q}(b)$, ez ellentmondás.

Minden ágon ellentmondást kaptunk, tehát az eredeti állítás igaz, a következmény fennáll.

e) $\exists y \forall x p(x,y) \models \forall x \exists y p(x,y)$.

Tegyük fel, hogy $\exists y \forall x p(x,y) \not\models \forall x \exists y p(x,y)$, vagyis $\mathrm{Mod}(\exists y \forall x p(x,y)) \not\subseteq \mathrm{Mod}(\forall x \exists y p(x,y))$.

Ekkor létezik olyan $\mathcal{A} = (A, I, \varphi)$, amire $\mathcal{A}(\exists y \forall x p(x,y)) = 1$ és $\mathcal{A}(\forall x \exists y p(x,y)) = 0$.

$\mathcal{A}(\exists y \forall x p(x,y)) = 1$-ből a $\exists y$ szemantikája szerint kapjuk, hogy van olyan $a \in A$ elem, amire $\mathcal{A}_{[y\mapsto a]}(\forall x p(x,y)) = 1$.

$\mathcal{A}(\forall x \exists y p(x,y)) = 0$-ból a $\forall x$ szemantikája szerint kapjuk, hogy van olyan $b \in A$ elem, amire $\mathcal{A}_{[x\mapsto b]}(\exists y p(x,y)) = 0$.

$\mathcal{A}_{[y\mapsto a]}(\forall x p(x,y)) = 1$-ből kapjuk a $\forall x$ szemantikája szerint, hogy tetszőleges $c \in A$-ra $\mathcal{A}_{[y\mapsto a, x\mapsto c]}(p(x,y)) = 1$, vagyis $\tilde{p}(c,a) = 1$. Speciálisan $c := b$-re is, vagyis $\tilde{p}(b,a) = 1$.

$\mathcal{A}_{[x\mapsto b]}(\exists y p(x,y)) = 0$-ból a $\exists y$ szemantikája szerint kapjuk, hogy tetszőleges $d \in A$-ra $\mathcal{A}_{[x\mapsto b, y\mapsto d]}(p(x,y)) = 0$, vagyis $\tilde{p}(b,d) = 0$. Speciálisan $d := a$-ra is, vagyis $\tilde{p}(b,a) = 0$, ami ellentmondás, tehát az eredeti állítás igaz, a következmény fennáll.

f) $\exists x(p(x) \wedge q(x)) \models (\exists x p(x)) \wedge (\exists x q(x))$:

Indirekt módon bizonyítható. Először a premissza kvantorához bevezetett $a \in A$ elemről láthatjuk, hogy $\tilde{p}(a) = \tilde{q}(a) = 1$. Ezt követően két esetre bomlik a feladat további része: a $\wedge$ szemantikája szerint ha 0 az eredmény, akkor $\mathcal{A}(\exists x p(x)) = 0$ vagy $\mathcal{A}(\exists x q(x)) = 0$.

Az első esetben azzal jutunk ellentmondásra, hogy az általánosan bevezetett b-re $\tilde{p}(b) = 0$ kell legyen, $b := a$-val ellentmondás; a másodikban pedig hasonlóképp, $\tilde{q}(a) = 1$-gyel jutunk ellentmondásra.

g) $(\exists x p(x)) \wedge (\forall y(p(y) \to q(y))) \models \exists x q(x)$:

Szintén indirekt úton okoskodhatunk: az $\exists x p(x)$-nél bevezetett a elemmel specializálva a $\forall y(p(y) \to q(y))$-nál bevezetett b elemet kapjuk, hogy $\tilde{q}(a) = 1$, míg ha a $\mathcal{A}(\exists x q(x)) = 0$-ból kapott tetszőleges c-t specializáljuk a-ra, azt kapjuk, hogy $\tilde{q}(a) = 0$, ez vezet ellentmondásra.

h) $\forall x \forall y((0 + x = y) \to (x = y)) \models 0 + 0 = 0$:

Bármely $\mathcal{A}$ modellre a $\forall x$-ra bevezetett tetszőleges a elemet $\tilde{0} = I(0)$-val, a $\forall y$-ra bevezetett tetszőleges b elemet $\tilde{0}\tilde{+}\tilde{0})$-val ($\mathcal{A}(0 + 0)$-val) specializálva kapjuk, hogy $(\tilde{0}\tilde{+}\tilde{0} = \tilde{0}\tilde{+}\tilde{0})$-ból következik $(\tilde{0} = \tilde{0}\tilde{+}\tilde{0})$. Ennek az implikációnak a $(\tilde{0}\tilde{+}\tilde{0} = \tilde{0}\tilde{+}\tilde{0})$ előtagja nyilvánvalóan igaz, ezért a $(\tilde{0} = \tilde{0}\tilde{+}\tilde{0})$ utótagja is. De ez éppen azt jelenti, hogy a logikai következtetés konkluzióját jelentő $0 + 0 = 0$ is igaz $\mathcal{A}$-ban.

i) $\exists x p(f(x)) \land \forall y(p(y) \to q(y)) \models \exists z q(z)$.

Az $\exists x p(f(x))$ részből kapjuk, hogy valamely a elemre $\tilde{p}(\tilde{f}(a)) = 1$, ezt az $\tilde{f}(a)$-t a $\forall y(p(y) \to q(y))$-ból kifejtett összefüggés tetszőleges b elemének helyére helyettesítve kapjuk, hogy $\tilde{q}(\tilde{f}(a)) = 1$, másrészt a jobb oldalban a $\exists z$ szemantikájára behozott általános c elemnek is $\tilde{f}(a)$-t adva értékül kapjuk, hogy $\tilde{q}(\tilde{f}(a)) = 0$, ez okozza az ellentmondást.

j) $\exists z p(z, c) \land \forall x \forall y(p(x, y) \to p(y, x)) \models \exists x p(c, x)$:

$\exists z p(z, c)$-t kifejtve kapunk egy egzisztenciálisan bevezetett $a \in A$ elemet.

A $\forall x \forall y(p(x, y) \to p(y, x))$-t kifejtve az x-nél bevezetett b helyére a-t, az y helyére bevezetett d helyére $\tilde{c}$-t helyettesítve kapjuk, hogy $\tilde{p}(a, \tilde{c}) = 0$ vagy $\tilde{p}(\tilde{c}, a) = 1$. A $\exists x p(c, x)$-nél az x helyére bevezetett (tetszőleges) e elem helyére szintén a-t helyettesítve kapjuk az ellentmondást.

k) $\exists z p(z, c) \land \forall x \forall y(p(x, y) \to p(y, f(x))) \models \exists x p(c, x)$:

Az előzőhöz hasonlóan, csak $\tilde{p}(\tilde{c}, \tilde{f}(a)) = 1$-et kapunk és az utolsó lépésben az x helyére bevezetett e helyére $\tilde{f}(a)$-t helyettesítünk.

l) $\forall x \forall y(p(x, f(x), y) \to p(f(x), x, y)) \models \forall y(p(c, f(c), y) \to p(f(c), c, y))$:

Kezdünk a $\mathcal{A}(\forall y(p(c, f(c), y) \to p(f(c), c, y))) = 0$ oldallal: $\forall y$-et kifejtve kapjuk, hogy van olyan a, amire az implikáció eredménye 0, ebből $\tilde{p}(\tilde{c}, \tilde{f}(\tilde{c}), a) = 1$ és $\tilde{p}(\tilde{f}(\tilde{c}), \tilde{c}, a) = 0$ jön ki.

Majd a másik formulában az x helyére vezetett b helyére $\tilde{c}$-t, az y helyére bevezetett d helyére pedig a-t helyettesítve megkapjuk az ellentmondást.

A feladat többi része hasonló gondolatmenettel igazolható.

3.6. Feladat. Cáfoljuk meg (alkalmas $\mathcal{A}$ ellenpélda megadásával), hogy az alábbi következmények fennállnak!

a) $\forall x(0 + x = x) \models \neg(1 + 1 = 0)$.

b) $\forall x((x = 0) \leftrightarrow \exists y(x + y = y)) \models \forall x(0 + x = x)$.

c) $\forall x \forall y((x + y = x) \leftrightarrow (y = 0)) \models \forall x(0 + x = x)$.

3.6. Feladat megoldása.

a) Tekintsük $\mathbb{Z}_2$-t, a modulo 2 maradékosztályok additív csoportját, vagyis $\mathbb{Z}_2 = (\{0, 1\}, \oplus)$-t, ahol $\oplus$ az összeadás modulo 2, más néven a kizáró vagy, azaz a XOR művelet. Ekkor $0 \oplus 0 = 0$ és $0 \oplus 1 = 1$, mégis $1 \oplus 1 = 0$ teljesül.

b) Tekintsük például a következő modellt: $\mathcal{A} = (A, I)$, ahol $A = \{a, b\}$, és $+$, interpretációja, $I(+) : A \times A \to A$ (az egyszerűség kedvéért $I(+)$ helyett $+$-szal jelölve) az alábbi:

$$\begin{array}{ll} a{+}a = a & a{+}b = a \\ b{+}a = b & b{+}b = a, \end{array}$$

továbbá $I(0) = a$.

c) Tekintsük például a következő modellt: $\mathcal{A} = (A, I)$, ahol $A = \{0, 1, 2\}$, és $+$, interpretációja, $I(+) : A \times A \to A$ (az egyszerűség kedvéért $I(+)$ helyett $+$-szal jelölve) az alábbi:

$+$	0	1	2
0	0	1	1
1	1	2	2
2	2	0	0

Továbbá $\tilde{0} = a$.

3.7. Feladat. Bizonyítsuk vagy cáfoljuk, hogy az alábbi ekvivalenciák fennállnak!

a) $\forall x[\forall y p(x, y) \leftrightarrow \forall z p(x, z)] \equiv \forall x \forall y p(x, y) \leftrightarrow \forall x \forall z p(x, z)$.

b) $\forall x[\exists y p(x, y) \leftrightarrow \forall z p(x, z)] \equiv \forall x \forall y p(x, y)$.

c) $\forall x(\forall y(x = y) \wedge F) \equiv \exists x(\forall y(x = y) \wedge F)$ tetszőleges F formulára.

3.7. Feladat megoldása.

a) Az ekvivalencia fennáll.

A formulákban a kötött változók átnevezése mindig ekvivalens átalakítás. Nevezzük át az első formulában a harmadik kvantor z változóját y-ra! Így azt kapjuk, hogy

$$\forall x[\forall y p(x,y) \leftrightarrow \forall y p(x,y)].$$

A második formulában a negyedik kvantor z változóját pedig nevezzük át u-ra:

$$\forall x \forall y p(x,y) \leftrightarrow \forall u \forall z p(u,z)$$

Ezután már nyilvánvaló, hogy mindkét formula tautológia, ezért ekvivalensek.

b) Az ekvivalencia nem áll fenn.

A második formula, $\forall x \forall y p(x,y)$ azt fejezi ki, hogy p interpretációja az univerzális reláció.

Ugyanakkor az első formula, $\forall x[\exists y p(x,y) \leftrightarrow \forall z p(x,z)]$, csak azt állítja, a p reláció gráfjában, minden egyes x csúcsra, vagy x-ből egyetlen él sem indul ki, vagy x-ből minden csúcsba vezet él.

Így például, az alábbi modellben első formula igaz, ám a második nem: $\mathcal{A} = (A, I)$, ahol $A = \{a, b\}$ és $\tilde{p} := I(p) : A \times A \to \{0, 1\}$ a következő:

$$\begin{aligned} \tilde{p}(a,a) &= 1 & \qquad \tilde{p}(b,a) &= 0 \\ \tilde{p}(a,b) &= 1 & \qquad \tilde{p}(b,b) &= 0. \end{aligned}$$

c) Az ekvivalencia fennáll. Az univerzális kvantorra és konjunkcióra vonatkozó azonosságot használva az első formulára azt kapjuk, hogy

$$\forall x(\forall y(x = y) \wedge F) \equiv \forall x \forall y(x = y) \wedge \forall x F.$$

Ebből $\forall x \forall y(x = y)$ azt fejezi ki, hogy a modell egyetlen elemből áll. Ezért $\forall x F$ helyére $\exists x F$ is írható, hiszen, ha a modell egyelemű, mindegy, hogy melyik kvantort használjuk. Az is nyilvánvaló, hogy $\forall x \forall y(x = y) \equiv \exists x \forall y(x = y)$. Így azt kapjuk, hogy

$$\forall x \forall y(x = y) \wedge \forall x F \equiv \exists x \forall y(x = y) \wedge \exists x F.$$

És bár általában $\exists x F_1 \wedge \exists x F_2 \equiv \exists x(F_1 \wedge F_2)$ nem igaz, esetünkben, mivel a formuláknak csak egyelemű modelljei lehetnek, mégis

$$\exists x \forall y(x = y) \wedge \exists x F \equiv \exists x(\forall y(x = y) \wedge F).$$

3.8. Feladat. Igazoljuk, hogy ha $F \models G$, akkor $\exists x(F \wedge G) \equiv (\exists xF) \wedge (\exists xG)$!

3.8. Feladat megoldása. Könnyen ellenőrizhető, hogy $\exists x(F \wedge G) \models (\exists xF) \wedge (\exists xG)$ mindig fennáll, hiszen, ha van közös x, melyre mind F mind G igaz, akkor F-hez és G-hez is külön-külön létezik olyan x, mely őket igazzá teszi.
Tegyük fel tehát $F \models G$, és ekkor kell még $(\exists xF) \wedge (\exists xG) \models \exists x(F \wedge G)$-t igazolnunk. Ehhez legyen $\mathcal{A} = (A, I)$ egy tetszőleges olyan modell, melyre $\mathcal{A} \models (\exists xF) \wedge (\exists xG)$. Ekkor $\mathcal{A} \models \exists xF$ miatt, $\exists a \in A : \mathcal{A}_{[x \mapsto a]} \models F$ teljesül. Felhasználva, hogy $F \models G$, azt kapjuk, hogy $\mathcal{A}_{[x \mapsto a]} \models G$. Így $\mathcal{A}_{[x \mapsto a]} \models F \wedge G$ a választott $a \in A$-ra, mely pontosan azt jelenti, hogy $\mathcal{A} \models \exists x(F \wedge G)$, melyet igazolni akartunk.

3.9. Feladat. Fennállnak-e az alábbi ekvivalenciák? Ha igen, igazoljuk; ha nem, adjunk ellenpéldát!

a) $\exists x(F \wedge G) \equiv (\exists xF) \wedge (\exists xG)$.

b) $\exists x(F \vee G) \equiv (\exists xF) \vee (\exists xG)$.

c) $\exists x(F \to G) \equiv (\exists xF) \to (\exists xG)$.

d) $\exists x(F \to G) \equiv (\forall xF) \to (\exists xG)$.

e) $\forall x(F \wedge G) \equiv (\forall xF) \wedge (\forall xG)$.

f) $\forall x(F \vee G) \equiv (\forall xF) \vee (\forall xG)$.

g) $\forall x(F \to G) \equiv (\forall xF) \to (\forall xG)$.

3.9. Feladat megoldása.

a) Nem igaz. Legyen a modell a természetes számok halmaza, $F(x)$ legyen igaz, ha x páros, $G(x)$ pedig, ha x páratlan.

b) Igaz.

$$\begin{aligned}
&\mathcal{A} \models \exists x(F \vee G) \\
&\Leftrightarrow \exists a \in A : \mathcal{A}_{[x \mapsto a]} \models F \vee G \\
&\Leftrightarrow \exists a \in A : (\mathcal{A}_{[x \mapsto a]} \models F \text{ vagy } \mathcal{A}_{[x \mapsto a]} \models G) \\
&\Leftrightarrow \exists a \in A : \mathcal{A}_{[x \mapsto a]} \models F \text{ vagy } \exists a \in A : \mathcal{A}_{[x \mapsto a]} \models G \\
&\Leftrightarrow \mathcal{A} \models \exists xF \text{ vagy } \mathcal{A} \models \exists xG \\
&\Leftrightarrow \mathcal{A} \models (\exists xF) \vee (\exists xG).
\end{aligned}$$

c) Nem igaz. Lásd a d) részt. Ellenpéldának pedig jó az a) rész modellje.

d) Igaz.

$\exists x(F \to G) \equiv \exists x(\neg F \vee G) \equiv (\exists x \neg F) \vee (\exists x G) \equiv (\neg \forall x F) \vee (\exists x G) \equiv \neg(\forall x F) \vee (\exists x G) \equiv (\forall x F) \to (\exists x G)$.

e) Igaz. A b) ponthoz hasonlóan igazolható.

f) Nem igaz. Az a) rész modelljében: ,,minden szám páros vagy páratlan" nem ugyanaz, mint ,,(minden szám páros) vagy (minden szám páratlan)".

g) Nem igaz. A ,,páros-páratlan" modellben szintén nem teljesül: A baloldali formula hamis, a jobboldali viszont igaz, mert az implikáció előtagja (mely szerint minden szám páros) hamis. Egyébként $\forall x(F \to G) \equiv \forall x(\neg F \vee G)$-t tovább nem tudjuk bontani, mert a $\forall$ kvantorra és a $\vee$ műveletre **nem** igaz, hogy $\forall x(F_1 \vee F_2) \equiv \forall x F_1 \vee \forall x F_2$. Az előző, f) rész pont ezt bizonyítja.

3.10. Feladat. Adjunk meg olyan F és G formulákat, amikre sem $F \models G$, sem $G \models F$ nem áll fenn, de mégis $\exists x(F \wedge G) \equiv (\exists x F) \wedge (\exists x G)$!

3.10. Feladat megoldása.
Legyen például $F = p(x) \wedge (x = 1)$ és $G = p(x) \wedge (x = 2) \wedge \neg(1 = 2)$, ahol 1 és 2 két konstans függvényszimbólum. Ekkor $F \not\models G$, mert abban a modellben, melyben $\varphi(x) = \tilde{1}$ és $\tilde{p}(\tilde{1})$ igaz, F igaz, G ellenben hamis. Hasonlóan $G \not\models F$. Ugyanakkor, mivel $x = 1$ és $x = 2$ egyszerre nem állhat fenn, sem $\exists x(F \wedge G)$, sem $(\exists x F) \wedge (\exists x G)$ nem lehet igaz egyetlen modellben sem, vagyis ezek a formulák ekvivalensek: Mindkettő azonosan hamis, mert

$$\mathrm{Mod}(\exists x(F \wedge G)) = \mathrm{Mod}((\exists x F) \wedge (\exists x G)) = \emptyset.$$

3.11. Feladat. Bizonyítsuk be, hogy minden zárt F formulára és olyan G formulára, melyben szabad változó az x, továbbá minden $\mathcal{A}$ struktúrára $\mathcal{A} \models (F \to \exists x G)$ akkor és csak akkor, ha $\mathcal{A} \models \exists x(F \to G)$.

3.11. Feladat megoldása. Definíció szerint bármely $\mathcal{A} = (A, I, \varphi)$ struktúrára

$$\begin{aligned}
&\mathcal{A} \models (F \to \exists x G)\\
&\Leftrightarrow \mathcal{A}(F) = 0 \text{ vagy } \mathcal{A}(\exists x G) = 1\\
&\Leftrightarrow \mathcal{A}(F) = 0 \text{ vagy } \exists a \in A : \mathcal{A}_{[x \mapsto a]}(G) = 1\\
&\overset{\star}{\Leftrightarrow} \exists a \in A : (\mathcal{A}_{[x \mapsto a]}(F) = 0 \text{ vagy } \mathcal{A}_{[x \mapsto a]}(G) = 1)\\
&\Leftrightarrow \mathcal{A} \models \exists x(F \to G)
\end{aligned}$$

A csillaggal ellátott ekvivalenciánál kihasználtuk, hogy F zárt formula, ezért $\mathcal{A}_{[x\mapsto a]}(F) = \mathcal{A}(F)$, tetszőleges $a \in A$ esetén. Ehhez különben elég lenne annyit feltételezni, hogy x nem fordul elő F-ben szabad változóként.

4. fejezet

A kompaktsági tétel, elméletek, axiómarendszerek.

Elméleti összefoglaló

Mind a zérusrendű mind az elsőrendű logikának igen fontos összefüggése a kompaktsági tétel, melynek megfogalmazása a következő:
Kompaktsági tétel. Elsőrendű vagy zérusrendű formulák egy Σ halmaza akkor és csak akkor elégíthető ki, ha Σ *minden* ***véges*** *részhalmaza kielégíthető.*
A kompaktsági tétel következménye. Egy Σ elsőrendű formulahalmaznak akkor és csak akkor logikai következménye egy F formula, ha létezik Σ-nak olyan **véges** Σ_0 részhalmaza, melynek F logikai következménye. Azaz

$$\Sigma \models F \iff \exists \Sigma_0 \subseteq \Sigma \text{ véges halmaz} : \Sigma_0 \models F.$$

A Γ_0 formulahalmaz a Γ formulahalmaz axiómarendszere, ha

$$\mathrm{Mod}(\Gamma_0) = \mathrm{Mod}(\Gamma)$$

azaz $\{\mathcal{A} \mid \mathcal{A} \models \Gamma_0\} = \{\mathcal{A} \mid \mathcal{A} \models \Gamma\}$.
A Γ formulahalmazt *végesen axiomatizálhatónak* nevezzük, ha Γ-nak van véges axiómarendszere.
Könnyen látható, hogy minden végesen axiomatizálható struktúrahalmaz egyetlen axiómával is axiomatizálható, hiszen véges sok formulát mindig összevonhatunk egyetlen egy formulává konjunkciók használatával.

Feladatok

4.1. Feladat. Legyen Γ egy kielégíthetetlen formulahalmaz. Mutassuk meg, hogy Γ-nak van olyan véges részhalmaza, mely kielégíthetetlen!

4.1. Feladat megoldása. A feladat állítása nem más, mint a kompaktsági tétel átfogalmazása tagadva annak mindkét oldalát. Valóban, indirekt módon okoskodva, ha egy kielégíthetetlen Γ-nak nem lenne kielégíthetetlen véges részhalmaza, akkor Γ minden véges részhalmaza kielégíthető lenne. Ez azonban a kompaktsági tétel miatt ahhoz vezetne, hogy Γ kielégíthető, ami ellentmondás. Ezért minden kielégíthetetlen Γ-nak kell, hogy legyen véges kielégíthetetlen részhalmaza.

4.2. Feladat. Bizonyítsuk be, hogy ha Γ olyan formulahalmaz, hogy minden véges részhalmaza kielégíthető, akkor egy tetszőleges F formula esetén $\Gamma \cup \{F\}$ vagy $\Gamma \cup \{\neg F\}$ minden véges részhalmaza kielégíthető!

4.2. Feladat megoldása. Tegyük fel, hogy Γ minden véges részhalmaza kielégíthető, és legyen F egy tetszőleges formula. Ekkor a kompaktsági tétel értelmében az egész Γ formulahalmaz is kielégíthető, azaz létezik olyan $\mathcal{A}$ modell, melyre $\mathcal{A} \models \Gamma$. Ha most $\mathcal{A}(F) = 1$, akkor $\mathcal{A} \models \Gamma \cup \{F\}$, különben, azaz ha $\mathcal{A}(F) = 0$, akkor $\mathcal{A} \models \Gamma \cup \{\neg F\}$. Tehát mindenképpen vagy $\Gamma \cup \{F\}$ vagy $\Gamma \cup \{\neg F\}$ kielégíthető, és így speciálisan kielégíthető ezen halmazok valamelyikének összes véges részhalmaza is.

4.3. Feladat. Mutassuk meg, hogy a kompaktsági tétellel ekvivalens állítás a kompaktsági tétel következménye, vagyis maga a kompaktsági tétel bebizonyítható a kompaktsági tétel következményéből.

4.3. Feladat megoldása. Tegyük fel, indirekt módon, hogy nem igaz a kompaktsági tétel, azaz van olyan Σ halmaz, melynek minden véges részhalmaza kielégíthető, de maga Σ nem az.
Mivel Σ kielégíthetetlen, így Σ-nak bármely formula logikai következménye, például az azonosan hamis formula is, tehát $\Sigma \models \downarrow$.
Most kihasználhatjuk, hogy tudjuk, hogy igaz a kompaktsági tétel következménye, ezért $\Sigma \models \downarrow$ azt eredményezi, hogy

$$\exists \Sigma_0 \subseteq \Sigma \text{ véges halmaz, melyre } \Sigma_0 \models \downarrow .$$

De ez csak akkor teljesülhet, ha Σ_0 is kielégíthetetlen, ami ellentmond annak a feltételezésünknek, hogy Σ minden véges részhalmaza kielégíthető.

4.4. Feladat. Legyen $\Sigma = \{F_1, F_2, \ldots\}$ formulák végtelen halmaza. Bizonyítsuk be, hogy Σ akkor és csak akkor kielégíthető, ha végtelen sok n-re $F_1 \wedge \ldots \wedge F_n$

kielégíthető.

4.4. Feladat megoldása.
A szükségesség triviális: Amennyiben Σ kielégíthető, létezik olyan $\mathcal{A}$ modell, melyben Σ *minden* formulája igaz, azaz $\mathcal{A} \models F_i$, minden $i = 1, 2, \ldots$-ra. Nyilvánvaló, hogy ez az $\mathcal{A}$ modell kielégíti az összes $F_1 \wedge \ldots \wedge F_n$ formulát is.
Az elégségesség igazolásához pedig a kompaktsági tételt hívjuk segítségül, mely szerint Σ kielégíthetőségének igazolásához elég megmutatnunk, hogy Σ minden véges részhalmaza kielégíthető. Legyen ezért Δ a Σ formulahalmaz egy tetszőleges *véges* részhalmaza. Alkalmas indexek választásával Δ-t felírhatjuk, mint

$$\Delta = \{F_{i_1}, F_{i_2}, \ldots, F_{i_k}\},$$

sőt még azt is feltehetjük, hogy $i_1 < i_2 < \ldots < i_k$. Mivel a feltétel szerint $F_1 \wedge \ldots \wedge F_n$ végtelen sok n-re kielégíthető, ezen n indexek között biztosan találhatunk olyat, mondjuk n'-t, mely i_k-nál nagyobb. Ekkor azonban $F_1 \wedge \ldots \wedge F_{n'}$ kielégíthetősége maga után vonja Δ kielégíthetőségét, hiszen Δ minden formulájának indexe kisebb mint n'. Mivel ez az érvelés Σ tetszőleges véges Δ részhalmazára alkalmazható, a kompaktsági tétel alapján kapjuk, hogy Σ kielégíthető.

4.5. Feladat. Tegyük fel, hogy $\{F_1, F_2, \ldots\}$ egy Γ formulahalmaz axiómarendszere és, hogy minden $n \geq 1$-re $F_{n+1} \models F_n$ de $F_n \not\models F_{n+1}$. Bizonyítsuk be, hogy ekkor Γ nem végesen axiomatizálható.

4.5. Feladat megoldása. Jelöljük a feladatban szereplő $\{F_1, F_2, \ldots\}$ axiómarendszert Δ-val. Az, hogy Δ a Γ formulahalmaz axiómarendszere azt jelenti, hogy $\mathrm{Mod}(\Delta) = \mathrm{Mod}(\Gamma)$.
A feladat megoldásához indirekt módon tegyük fel, hogy Γ végesen axiomatizálható, azaz létezik olyan $\Gamma_0 = \{G_1, \ldots, G_k\}$ véges halmaz, melyre $\mathrm{Mod}(\Gamma_0) = \mathrm{Mod}(\Gamma)$. Világos, hogy ekkor $G = G_1 \wedge \ldots \wedge G_k$ egymagában is axiomatizálja Γ-t, így $\mathrm{Mod}(G) = \mathrm{Mod}(\Gamma) = \mathrm{Mod}(\Delta)$. Ez többek közt azt is jelenti, hogy $G \models \Delta$ és $\Delta \models G$.
Most alkalmazhatjuk a kompaktsági tétel következményét: Mivel $\Delta = \{F_1, F_2, \ldots\}$-nak logikai következménye G, ezért létezik Δ-nak olyan Δ_0 véges részhalmaza, melynek már logikai következménye G, azaz $\Delta_0 \models G$. Jelöljük e véges Δ_0 formulahalmaz elemeit a következő módon: $\Delta_0 = \{F_{i_1}, F_{i_2}, \ldots, F_{i_k}\}$, ahol $i_1 < i_2 < \ldots < i_k$.
Mivel a feltételek szerint $F_{n+1} \models F_n$, minden $n \geq 1$-re, ezért $F_{i_k} \models F'$, bármely $F' \in \Delta_0$-ra. Így

$$F_{i_k} \models \Delta_0, \quad \Delta_0 \models G, \quad G \models \Delta.$$

A logikai következmény tranzitivitása miatt $F_{i_k} \models \Delta$. Ám ekkor $F_{i_k+1} \in \Delta$ miatt $F_{i_k} \models F_{i_k+1}$ is teljesül, de ez ellentmond a feladat feltételeinek. Az ellentmondás

oka, hogy feltettük, hogy Γ végesen axiomatizálható, ezzel bebizonyítottuk, hogy nem az.

4.6. Feladat. Legyen $F \equiv G$. Mutassuk meg, hogy $F' \equiv G'$, ahol F'-t és G'-t úgy kapjuk F-ből és G-ből, hogy bennük felcseréljük a $\vee$-t $\wedge$-re és a $\wedge$-t $\vee$-re! (F-ben és G-ben csak $\neg, \vee$ és $\wedge$ műveleti jelek szerepelhetnek.)

4.6. Feladat megoldása. Legyen F egy olyan ítéletkalkulusbeli formula, melyben csak az x_1, x_2, ..., x_n, $(n \geq 0)$, ítéletváltozók, valamint a $\neg$, $\vee$ és $\wedge$ logikai műveletek szerepelnek. Azt az F' formulát, melyet F-ből úgy kapunk, hogy benne a $\vee$ és $\wedge$ műveleteket felcseréljük, F *duálisának* nevezzük, és F felépítése szerinti indukcióval a következőképpen definiáljuk:

$$F' = \begin{cases} x_i, & \text{ha } F = x_i \text{ alakú} \\ \neg F_1', & \text{ha } F = \neg F_1 \text{ alakú} \\ F_1' \wedge F_2', & \text{ha } F = F_1 \vee F_2 \text{ alakú} \\ F_1' \vee F_2', & \text{ha } F = F_1 \wedge F_2 \text{ alakú.} \end{cases}$$

Az állítás bizonyítása azon az észrevételen alapszik, hogy a $\vee$ és $\wedge$ műveletek egymás duálisai abban az értelemben, hogy $\neg x_1 \wedge \neg x_2 = \neg(x_1 \vee x_2)$ és $\neg x_1 \vee \neg x_2 = \neg(x_1 \wedge x_2)$. Ezért, ha az F duális formulájába, $F'(x_1, x_2, \ldots, x_n)$-be az x_1, ..., x_n változók helyére rendre $\neg x_1$, ... $\neg x_n$-et helyettesítünk, akkor $\neg F$-fel ekvivalens formulát kapunk. Ennek formális bizonyítása a következő: **Lemma.** Bármely csak a $\neg$, $\vee$ és $\wedge$ műveleteket tartalmazó $F(x_1, \ldots, x_n)$ formulára és annak $F'(x_1, \ldots, x_n)$ duálisára fennáll a következő összefüggés:

$$F'(\neg x_1, \ldots, \neg x_n) \equiv \neg F(x_1, \ldots, x_n).$$

Bizonyítás. A lemmát F felépítése szerinti indukcióval bizonyítjuk.

- Ha $F = x_i$, $(1 \leq i \leq n)$, azaz F egyetlen ítéletváltozóból áll, akkor $F' = x_i$ így $F'(\neg x_1, \ldots, \neg x_n) = \neg x_i = \neg F(x_1, \ldots, x_n)$.
- Ha $F = \neg F_1$ alakú, azaz F legkülső művelete negáció, akkor F_1-re alkalmazva az indukciós feltételt azt kapjuk, hogy

  $$F_1'(\neg x_1, \ldots, \neg x_n) \equiv \neg F_1(x_1, \ldots, x_n),$$

  és így

  $$F'(\neg x_1, \ldots, \neg x_n) \equiv \neg F_1'(\neg x_1, \ldots, \neg x_n) \equiv$$
  $$\equiv \neg(\neg F_1(x_1, \ldots, x_n)) \equiv \neg F(x_1, \ldots, x_n).$$

- Ha $F = F_1 \vee F_2$ alakú, azaz F legkülső művelete diszjunkció, akkor F_1-re és F_2-re alkalmazva az indukciós feltételt azt kapjuk, hogy

$$\begin{aligned}
&F'(\neg x_1, \ldots, \neg x_n) \equiv \\
&\equiv F_1'(\neg x_1, \ldots, \neg x_n) \wedge F_2'(\neg x_1, \ldots, \neg x_n) \equiv \\
&\equiv \neg F_1(x_1, \ldots, x_n) \wedge \neg F_2(x_1, \ldots, x_n) \equiv \\
&\equiv \neg[F_1(x_1, \ldots, x_n) \vee F_2(x_1, \ldots, x_n)] \equiv \\
&\equiv \neg F(x_1, \ldots, x_n).
\end{aligned}$$

- Ha $F = F_1 \wedge F_2$ alakú, azaz F legkülső művelete konjunkció, akkor az előző esethez teljesen hasonló módon

$$\begin{aligned}
&F'(\neg x_1, \ldots, \neg x_n) \equiv \\
&\equiv F_1'(\neg x_1, \ldots, \neg x_n) \vee F_2'(\neg x_1, \ldots, \neg x_n) \equiv \\
&\equiv \neg F_1(x_1, \ldots, x_n) \vee \neg F_2(x_1, \ldots, x_n) \equiv \\
&\equiv \neg[F_1(x_1, \ldots, x_n) \wedge F_2(x_1, \ldots, x_n)] \equiv \\
&\equiv \neg F(x_1, \ldots, x_n).
\end{aligned}$$

Ezzel a lemma bizonyítását befejeztük.
Ezután a lemma segítségével már könnyen igazolható F' és G' ekvivalenciája. Valóban

$$\begin{aligned}
&F'(x_1, \ldots, x_n) \equiv F'(\neg(\neg x_1), \ldots, \neg(\neg x_n)) \equiv \neg F(\neg x_1, \ldots, \neg x_n) \equiv \\
&\equiv \neg G(\neg x_1, \ldots, \neg x_n) \equiv G'(\neg(\neg x_1), \ldots, \neg(\neg x_n)) \equiv G'(x_1, \ldots, x_n).
\end{aligned}$$

Ezzel a feladat állítását igazoltuk.

4.7. Feladat. Tekintsük az ekvivalencia relációk alábbi axiómarendszerét

$$\begin{aligned}
F_1 &= \forall x p(x, x), \\
F_2 &= \forall x \forall y (p(x, y) \to p(y, x)), \\
F_3 &= \forall x \forall y \forall z \big((p(x, y) \wedge p(y, z)) \to p(x, z) \big).
\end{aligned}$$

Mutassuk meg, hogy semelyik F_i nem következménye a másik kettőnek!

4.7. Feladat megoldása. A feladat állítását úgy igazolhatjuk, hogy megadunk egy-egy egy olyan modellt, melyben a három formula közül kettő igaz, a harmadik azonban nem. Világos, hogy a modellekben az első, második és harmadik formula rendre a p reláció reflexív, szimmetrikus és tranzitív tulajdonságát fejezi ki. Ezért célunknak megfelelő modellek azok, melyekben p interpretációja a fenti három tulajdonság közül csak pontosan kettővel bír.

- Reflexív és szimmetrikus, de nem tranzitív reláció például a nemüres halmazok körében a ,,két halmaz nem diszjunkt (azaz van közös elemük)" reláció.
- Reflexív és tranzitív, de nem szimmetrikus reláció például az egész számokon a ,,kisebb vagy egyenlő" reláció.
- Végül szimmetrikus és tranzitív, de nem reflexív reláció például az egész számokon az üres reláció, mely mindig hamis.

Ezzel igazoltuk, hogy az ekvivalencia relációt definiáló három axióma egymástól független.

4.8. Feladat. Legyen

$$F = \exists x \forall y \exists z([p(y,z) \to p(x,z)] \to [p(x,x) \to p(y,x)]).$$

a) Mutassuk meg, hogy F-et kielégíti az összes olyan struktúra, amelyben az univerzum véges!

b) Bizonyítsuk be, hogy F nem tautológia!

4.8. Feladat megoldása. Útmutatás.

a) Alakítsuk át F-et megfelelően, és mutassuk meg, hogy minden véges $\mathcal{A}$-ra $\mathcal{A} \not\models \neg F$.

b) Keressünk olyan $\mathcal{A}$ struktúrát, amelyben az univerzum végtelen, és $\mathcal{A} \models \neg F$.

Megoldás. Kezdjük a b) ponttal, mert az könnyebb, és megoldásával könnyebben megérthetjük, hogy mit is állít az F formula. F akkor nem tautológia, ha $\neg F$ kielégíthető. Alakítsuk át ezért egy kicsit $\neg F$-et felhasználva, a kvantoros De-Morgan azonosságokat és azt, hogy az implikáció csak akkor hamis, ha előtagja igaz és utótagja hamis.

$$\begin{aligned}
\neg F = \neg\exists x \forall y \exists z([p(y,z) \to p(x,z)] \to [p(x,x) \to p(y,x)]) \equiv \\
\equiv \forall x \exists y \forall z \neg([p(y,z) \to p(x,z)] \to [p(x,x) \to p(y,x)]) \equiv \\
\equiv \forall x \exists y \forall z([p(y,z) \to p(x,z)] \wedge \neg[p(x,x) \to p(y,x)]) \equiv \\
\equiv \forall x \exists y \forall z([p(y,z) \to p(x,z)] \wedge p(x,x) \wedge \neg p(y,x)) \equiv \\
\equiv \forall x p(x,x) \wedge \forall x \exists y(\forall z[p(y,z) \to p(x,z)] \wedge \neg p(y,x)).
\end{aligned}$$

Ezek után már nem nehéz látni, hogy $\neg F$-et kielégíti az egész számok modellje, ha p-t a szokásos ,,kisebb egyenlő" relációval interpretáljuk. Valóban $\leq$ reflexív, ezért $\forall x p(x,x)$ igaz. A formula második tagjában pedig minden x-hez

y-nak választhatjuk x közvetlen rákövetkezőjét, azaz $y = x + 1$-et, mert ekkor $\forall z[p(y,z) \to p(x,z)]$ azt jelenti, hogy ,,minden z-re, ha $x + 1 \leq z$, akkor $x \leq z$, ami nyilvánvalóan igaz, és $\neg p(y,x)$, azaz $x + 1 \not\leq x$ is teljesül.
Ezután a feladat a) részét indirekt módon igazolhatjuk. Amennyiben létezne olyan $\mathcal{A} = (A, I)$ véges modell, mely nem elégítené ki F-et, akkor ez a modell kielégítené $\neg F$-et, vagyis $\mathcal{A} \models \neg F$ teljesülne.
Feltételezésünk szerint A véges, mondjuk legyen $n > 1$ elemű. Jelöljük A elemeit a következő módon: Először a_1 legyen A egy tetszőleges eleme, majd minden $2 \leq i \leq n$-re a_i legyen egy olyan y elem, melyet $x = a_{i-1}$ választás esetén a $\neg F$ formula $\forall x \exists y(\forall z[p(y,z) \to p(x,z)] \wedge \neg p(y,x))$ részének igazsága ad.
Belátható, hogy az így választott $\mathcal{A} = \{a_1, \ldots, a_n\}$ elemekkel a p reláció $\tilde{p}$ interpretációja szükségképpen az alábbi

$$\tilde{p}(a_i, a_j) = 1 \iff i \leq j \quad \text{bármely } 1 \leq i, j \leq n\text{-re.} \qquad (*)$$

Ennek (például $j - i$ szerinti teljes indukcióval történő) bizonyítását az olvasóra bízzuk.
Végül ezzel ellentmondásra jutottunk, mert $x = a_n$ választással (*) miatt nem létezik olyan y, melyre $\neg p(y,x)$ teljesülne, ezért az $\mathcal{A}$ modellben a $\neg F$ formula mégsem lehet igaz. Az ellentmondás oka az $\mathcal{A}$ modell végességének feltételezése, így egyetlen véges modell sem elégíti ki $\neg F$-et, azaz minden véges modell kielégíti F-et, melyet bizonyítanunk kellett.

5. fejezet

Az ítéletkalkulus, normálformák, Boole-függvények, teljes rendszerek.

Elméleti összefoglaló

Boole-függvényen a logikai értékek $\{0,1\}$ halmaza feletti függvényeket értünk, azaz egy $n \geq 0$ változó Boole-függvény nem más, mint egy tetszőleges $f : \{0,1\}^n \to \{0,1\}$ leképezés. Jól ismert konstans Boole-függvények a $\downarrow$ és $\uparrow$, egyváltozós Boole-függvény a $\neg$, kétváltozósak a $\vee, \wedge, \to$ és $\leftrightarrow$, de más és kettőnél több változós Boole-függvényeket is definiálhatunk.
Az ítéletkalkulus formuláit *Boole-formuláknak* is nevezzük. *Literálnak* hívunk egy p ítéletváltozót, vagy annak $\neg p$ tagadását. Az előbbit *pozitív* az utóbbit *negatív* literálként is említjük. Ezek egymás *ellentetjei*. Egy ℓ literál ellentettjének a jele $\bar{\ell}$, azaz

$$\bar{\ell} = \begin{cases} \neg p, & \text{ha } \ell = p, \\ p & \text{ha } \ell = \neg p. \end{cases}$$

Azt mondjuk, hogy egy formula *konjunktív normálformában* (röviden *KNF-ben*) van, ha literálok diszjunkciójának konjunkciójaként áll elő, azaz

$$\bigwedge_{i=1}^{n} \bigvee_{j=1}^{m_i} \ell_{i,j}$$

alakú, ahol $\ell_{i,j}$ literálok, $1 \leq j \leq m_i$, $1 \leq i \leq n$, $n \geq 0$. Az $\bigvee_{j=1}^{m_i} \ell_{i,j}$ részformulák a konjunktív normálforma (konjunkciós) *tagjai* vagy *klózai*.

Hasonlóan egy formula a *diszjunktív normálformája* (*DNF*) a következő:

$$\bigvee_{i=1}^{n} \bigwedge_{j=1}^{m_i} \ell_{i,j},$$

ahol $\ell_{i,j}$ literálok, $1 \leq j \leq m_i$, $1 \leq i \leq n$, $n \geq 0$.
Boole-függvények egy rendszere *teljes* vagy *adekvát*, ha segítségükkel minden (akárhány változós) Boole-függvény felírható. Jól ismert, hogy minden Boole-függvényhez megadható egy konjunktív normálformájú és egy diszjunktív normálformájú formula. Ebből közvetlenül adódik, hogy a $\{\vee, \wedge, \neg\}$ függvényrendszer teljes. Könnyen látható, hogy a De-Morgan azonosságok miatt $x \wedge y \equiv \neg(\neg x \vee \neg y)$ és $x \vee y \equiv \neg(\neg x \wedge \neg y)$, ezért a $\vee$ vagy $\wedge$ műveletek egyikét elhagyva még mindig teljes rendszert kapunk, vagyis $\{\vee, \neg\}$ és $\{\wedge, \neg\}$ szintén teljes rendszerek. Végül beláthatjuk, hogy $\{\rightarrow, \downarrow\}$ is teljes, mert a $\{\vee, \neg\}$ halmaz tagjai kifejezhetők az elemeikkel: $\neg x \equiv x \rightarrow\downarrow$ és $x \vee y \equiv (x \rightarrow\downarrow) \rightarrow y$.

Feladatok

5.1. Feladat. Bizonyítsuk be igazságtáblázat módszerrel, hogy az alábbi formulák tautológiák, azaz azonosan igaz formulák, tetszőleges F és G elsőrendű formulákra!

a) $(F \rightarrow G) \leftrightarrow ((\neg G) \rightarrow (\neg F))$.

b) $(F \rightarrow G) \leftrightarrow (\neg F \vee G)$;

c) $(F \rightarrow G) \leftrightarrow \neg(F \wedge \neg G)$;

d) $(F \rightarrow G) \leftrightarrow (\neg F \vee (F \wedge G))$.

5.1. Feladat megoldása.

a) Elég azt igazolnunk, hogy a formula Boole-váza, azaz $(p \rightarrow q) \leftrightarrow (\neg q \rightarrow \neg p)$ tautológia (az ítéletkalkulusban.)

- A sorokban a p és q változók minden lehetséges igazságértéke szerepel.
- Minden műveleti jel alá az általa képzett részformula igazságértékét írjuk.
- A formula értéke a legkülső műveleti jel, esetünkben a $\leftrightarrow$ alatt található.

p	$\rightarrow$	q	$\leftrightarrow$	$(\neg$	q	$\rightarrow$	$\neg$	$p)$
0	1	0	1	1	0	1	1	0
0	1	1	1	0	1	1	1	0
1	0	0	1	1	0	0	0	1
1	1	1	1	0	1	1	0	1

A feladat többi része hasonlóan igazolható.

5.2. Feladat. Tekintsük a következő ítéleteket:

- p_1: Fizetésemelést kapok;
- p_2: Szabadságot kapok;
- p_3: Elmegyek kirándulni;
- p_4: Megjavítom az autómat;
- p_5: Az autómmal utazok.

Formalizáljuk a következő mondatokat:

- Szabadságot kapok.
- Ha szabadságot kapok, akkor megjavítom az autómat.
- Vagy fizetésemelést kapok, vagy nem javítom meg az autómat.
- Ha megjavítom az autómat, akkor az autómmal megyek kirándulni.
- Nem fordulhat elő az, hogy ha megjavítom az autómat, akkor nem kapok fizetésemelést vagy nem megyek el kirándulni.

A feladat arra igyekszik rávilágítani, hogy a természetes nyelvi megfogalmazásoknak a logika szigorúan egyértelmű nyelvére való átültetése korántsem egyszerű feladat. A következő fejezetek algoritmusaival majd azt is igazolni tudjuk, hogy az első négy állításnak logikai következménye az ötödik.

5.2. Feladat megoldása.

- Szabadságot kapok: $F_1 = p_2$.
- Ha szabadságot kapok, akkor megjavítom az autómat: $F_2 = p_2 \rightarrow p_4$.

- Vagy fizetésemelést kapok, vagy nem javítom meg az autómat: Vegyük észre, hogy kizáró vagyról van szó, hiszen épp azt szeretnénk kifejezni, hogy a két eset egyszerre nem teljesülhet. Ezért $F_3 = p_1 \oplus \neg p_4 \equiv (p_1 \vee \neg p_4) \wedge (\neg p_1 \vee p_4)$. Ez ugyanaz, mint, ha azt mondanám, hogy, ha fizetésemelést kapok, megjavítom az autóm, és fordítva, ha megjavítom az autóm, akkor megkaptam a fizetésemelést, azaz $p_1 \leftrightarrow p_4$.

- Ha megjavítom az autómat, akkor az autómmal megyek kirándulni: $p_4 \rightarrow (p_3 \wedge p_5)$.

- Nem fordulhat elő az, hogy ha megjavítom az autómat, akkor nem kapok fizetésemelést vagy nem megyek el kirándulni: $\neg[p_4 \rightarrow (\neg p_1 \vee \neg p_3)]$.

5.3. Feladat. Formalizáljuk ítéletkalkulusbeli formulával és oldjuk meg az alábbi logikai fejtörőt!

- A lovagok mindig igazat mondanak.
- A lókötők mindig hazudnak.

Döntsük el, hogy milyen típusú emberekkel beszélgettünk!

- **A** mondja: ,,**B** lókötő."
- **B** mondja: ,,Mit képzel! Nem vagyok lókötő. De, ha **A** lókötő, akkor **C** is."

5.3. Feladat megoldása. Tekintsük azt a speciális modellt, amelyben a (kivételesen nagybetűvel jelölt) A ítéletváltózó értéke igaz, ha **A** lovag, hamis, ha **A** lókötő. Hasonlóan a B és C változó fejezze ki **B** és **C** lovag vagy lókötő voltát: értékük pontosan akkor legyen igaz, ha az illető lovag.
Vegyük észre, hogy a feladat feltétei szerint minden szereplő ,,lovagsága", vagyis igazmondása, meg kell, hogy egyezzen az általa állított kijelentés igazságával. Például **A** azt mondta, hogy ,,**B** lókötő", vagyis $\neg B$ pontosan akkor igaz, ha **A** lovag, ezért biztos, hogy $A \leftrightarrow \neg B$ igaz. A többiek állítását is figyelembe véve, azt kapjuk, hogy

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

Belátható, hogy a fenti formula csak akkor igaz, ha B értéke 1, A és C értéke 0, vagyis a formula ekvivalens $\neg A \wedge B \wedge \neg C$-vel. Azaz minden modellben, melyben a feladat feletételeit leíró formula igaz, teljesül $\neg A \wedge B \wedge \neg C$ is. Esetünkben ez azt jelenti, hogy A és C lókötő, B pedig lovag.

5.4. Feladat. Hozzuk konjunktív és diszjunktív normál alakra a következő formulákat:

a) $(p \to q) \leftrightarrow \neg r$;

b) $(p \to q) \leftrightarrow (p \to r)$;

c) $(p \leftrightarrow q) \to (r \vee s)$;

d) $(\neg p \to q) \vee ((p \wedge \neg r) \leftrightarrow q)$;

e) $(p \leftrightarrow (q \leftrightarrow r))$.

f) $\neg(\neg((p \to q) \wedge (q \to r) \wedge \neg(s \wedge r)) \vee (p \to \neg s))$;

g) $((p \vee q \vee r) \wedge \neg p \wedge \neg r) \to p$;

h) $((p \to q) \wedge (p \to r) \wedge \neg(p \to (q \wedge r)))$;

i) $(p \to q) \to (\neg q \to \neg p)$.

5.4. Feladat megoldása. Csak a feladat c) részét ismertetjük részletesen, a többi esetben csak a végeredményeket közöljük.

1.lépés: $F \to G$ helyett $\neg F \vee G$
$F \leftrightarrow G$ helyett $(F \to G) \wedge (G \to F) \equiv (\neg F \vee G) \wedge (\neg G \vee F)$ írása.
$(p \leftrightarrow q) \to (r \vee s) \equiv \neg(p \leftrightarrow q) \vee (r \vee s) \equiv \neg \Big((\neg p \vee q) \wedge (\neg q \vee p) \Big) \vee (r \vee s) \equiv$

2.lépés: $\neg$ bevitele
$\neg(F \vee G) \equiv \neg F \wedge \neg G$, $\neg(F \wedge G) \equiv \neg F \vee \neg G$, és $\neg\neg F \equiv F$ alapján.
$\equiv \Big(\neg(\neg p \vee q) \vee \neg(\neg q \vee p) \Big) \vee (r \vee s) \equiv \Big((\neg\neg p \wedge \neg q) \vee (\neg\neg q \wedge \neg p) \Big) \vee r \vee s \equiv$
$(p \wedge \neg q) \vee (q \wedge \neg p) \vee r \vee s$.
Ez már DNF, a harmadik lépésre most a DNF-hez nincs szükség.

3. lépés: A disztributivitás használata:
KNF-nál a $\wedge$ kivitele: $(F \wedge G) \vee H \equiv (F \vee H) \wedge (G \vee H)$ alapján.
DNF-nál a $\vee$ kivitele: $(F \vee G) \wedge H \equiv (F \wedge H) \vee (G \wedge H)$ alapján.
$\Big((p \wedge \neg q) \vee (q \wedge \neg p) \Big) \vee r \vee s \equiv \Big((p \vee q) \wedge (p \vee \neg p) \wedge (\neg q \vee q) \wedge (\neg q \vee \neg p) \Big) \vee r \vee s \equiv$
$\Big((p \vee q) \wedge (\neg q \vee \neg p) \Big) \vee r \vee s \equiv (p \vee q \vee r \vee s) \wedge (\neg p \vee \neg q \vee r \vee s)$ Ez már KNF.

A feladat többi részének csak a végeredményét közöljük. elképzelhető, hogy a számítás eredménye jó, még sem pontosan az alábbiak jönnek ki, ilyenkor meg kell vizsgálni, egyszerűsíthető-e valamelyik válasz.

a) $(\neg p \vee q \vee r) \wedge (\neg r \vee p) \wedge (\neg r \vee \neg q)$ és
$(\neg r \wedge \neg p) \vee (\neg r \wedge q) \vee (p \wedge \neg q \wedge r)$.

b) $(\neg p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r)$ és
$\neg p \vee (q \wedge \neg p) \vee (q \wedge r) \vee (p \wedge \neg q \wedge \neg r) \equiv \neg p \vee (r \wedge q) \vee (\neg q \wedge \neg r)$.

c) $(\neg p \vee \neg q \vee r \vee s) \wedge (p \vee q \vee r \vee s)$ és
$(\neg p \wedge q) \vee (\neg q \wedge p) \vee r \vee s$.

d) KNF: $\uparrow$ (tautológia, üres KNF.)
DNF: $p \vee \neg p$. Vigyázat, az üres DNF azonosan hamis, ezért nem jó megoldásnak.

e) $(\neg p \vee \neg q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (p \vee \neg q \vee \neg r) \wedge (p \vee q \vee r)$ és
$(\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge \neg r) \vee (p \wedge q \wedge r)$.

f) KNF: $\downarrow$ (kielégíthetetlen formula) vagy $p \wedge \neg p$ vagy $\square$ (csak az üresklózt tartalmazó formula.)
DNF: $\emptyset$ (üres diszjunktív normálforma) vagy $p \wedge \neg p$, mint egyetlen diszjunkciós klóz.

g) $\neg q \vee p \vee r$ és
$\neg q \vee p \vee r$ (Ugyanaz jó mert KNF és DNF egyszerre.)

h) $\downarrow$ (azonosan hamis).

i) $\uparrow$ (azonosan igaz).

5.5. Feladat. Írjuk fel a kétváltozós Boole-függvényeket!

5.5. Feladat megoldása.
A $2^{2^4} = 16$ darab kétváltozós Boole-függvény van, mert az igazságtáblázatban 4 helyre írhatunk egymástól függetlenül 0 vagy 1 értéket. Ezeket a függvényeket, aszerint, hogy az igazságtáblázatukban hány százalékban szerepel 1-es érték a következőképpen csoportosíthatjuk:

$0-100\%$	$\downarrow$	és tagadása:	$\uparrow$
$25-75\%$	$\wedge, \not\rightarrow, \not\leftarrow, \|\|$	és tagadásuk:	$\mid, \rightarrow, \leftarrow, \vee$
$50-50\%$	$x_1, x_2, \leftrightarrow$	és tagadásuk:	$\neg x_1, \neg x_2, \not\leftrightarrow$

ahol

- x_1 az $f(x_1, x_2) = x_1$, továbbá x_2 az $f(x_1, x_2) = x_2$ függvényt jelöli.
- $x|y = \neg(x \wedge y)$ a NAND függvény, (néhol jele $\uparrow$, de az nálunk a konstans igaz művelet.)
- $x||y = \neg(x \vee y)$ a NOR függvény, (néhol jele $\downarrow$, de az nálunk a konstans hamis.)
- $x \leftarrow y := y \rightarrow x$ (fordított irányú implikáció);
- $x \not\rightarrow y := \neg(x \rightarrow y)$ (az implikáció tagadása);
- $x \not\leftarrow y := \neg(x \leftarrow y)$ (a fordított implikáció tagadása);
- $x \not\leftrightarrow y := \neg(x \leftrightarrow y)$ (nem ekvivalencia, más jelöléssel $x \oplus y$, kizáró vagy, XOR);

5.6. Feladat. Mutassuk meg, hogy $\{\rightarrow, \neg\}$ adekvát, azaz teljes rendszert alkot!

5.6. Feladat megoldása. $x \vee y = \neg x \rightarrow y$, és a $\{\neg, \vee\}$ halmazról pedig már tudjuk, hogy teljes.

5.7. Feladat. Fejezzük ki

a) $\wedge$-t és $\rightarrow$-t a $\vee$ és a $\neg$ művelet segítségével;

b) $\vee$-t és $\rightarrow$-t a $\wedge$ és a $\neg$ művelet segítségével;

c) $\wedge$-t és $\vee$-t a $\rightarrow$ és a $\neg$ művelet segítségével;

d) $\wedge$, $\vee$, $\neg$-t és $\rightarrow$-t a $|$ művelet segítségével;

e) $\neg$-t a $\rightarrow$ és a $\downarrow$ művelet segítségével;

f) $\neg$-t a $\oplus$ (kizáró vagy, $\not\leftrightarrow$) a $\uparrow$ művelet segítségével;

g) $\vee$-t az $\rightarrow$ művelet segítségével.

5.7. Feladat megoldása.

a) $x \wedge y = \neg(\neg x \vee \neg y), \quad x \rightarrow y = \neg x \vee y;$

b) $x \vee y = \neg(\neg x \wedge \neg y), \quad x \rightarrow y = \neg(x \wedge \neg y);$

c) $x \wedge y = \neg(x \to \neg y), \quad x \vee y = \neg x \to y;$

d) $\neg x = (x|x), \quad x \wedge y = (x|y)|(x|y), \quad x \vee y = (x|x)|(y|y),$
$x \to y = x|(y|y);$

e) $\neg x = x \to\!\downarrow;$

f) $\neg x = x \oplus \uparrow;$

g) $x \vee y = (x \wedge \neg y) \vee y = \neg(x \to y) \vee y = (x \to y) \to y.$

5.8. Feladat. Bizonyítsuk, be hogy az alábbi rendszerek teljesek!

a) $\{\neg, \leftarrow\};$

b) $\{\neg, \not\leftarrow\};$

c) $\{\to, \not\to\};$

d) $\{\to, \not\leftarrow\};$

e) $\{\leftarrow, \not\leftarrow\};$

f) $\{\wedge, \not\to, \uparrow\}.$

5.8. Feladat megoldása. Az előző feladathoz hasonlóan fejezzük ki valamely teljes rendszer elemeit a megadott műveletekkel. Például a b) részben

$$x \to y \equiv \neg(y \not\leftarrow x),$$

mert $\neg(y \not\leftarrow x) \equiv \neg(\neg(y \leftarrow x)) \equiv \neg\neg(x \to y)$.
Így $\neg$ szerepel a halmazban, $\to$ pedig a fenti módon kifejezhető. $\neg, \to$-tól pedig már tudjuk, hogy teljes rendszer, ezért $\{\neg, \not\leftarrow\}$ is az. Konkrétan a többi esetben

a) $x \to y \equiv y \leftarrow x$ és $\{\neg, \to\}$-ről már tudjuk, hogy teljes.

b) Lásd korábban, vagy $x \leftarrow y = \neg(x \not\leftarrow y)$ és hivatkozzunk az a) részre.

c) $\downarrow = x \not\to x$ és $\{\to, \downarrow\}$-ról már tudjuk, hogy teljes.

d) $\downarrow = x \not\leftarrow x$ és $\{\to, \downarrow\}$-ról már tudjuk, hogy teljes.

e) $x \to y = y \leftarrow x$ és hivatkozzunk a d) részre.

f) $\neg x \equiv x \not\to\uparrow$ és $\{\neg, \wedge\}$-ről már tudjuk, hogy teljes.

5.9. Feladat. Mutassuk meg, hogy a $\{\wedge, \vee, \rightarrow\}$ halmaz *nem* adekvát (azaz nem alkot teljes rendszert.)

5.9. Feladat megoldása. A negáció nem fejezhető ki. Minden, a halmaz műveleteiből felépített egyváltozós $f(x)$ függvényre, $f(1) = 1$, mert $1 \vee 1 = 1 \wedge 1 = 1 \rightarrow 1 = 1$.
Ezt a tulajdonságot úgy hívjuk, hogy a fenti függvények 1-*tartóak*.
De a negáció nem 1-tartó, mert $\neg 1 = 0$, ezért a negáció nem fejezhető ki.

5.10. Feladat. Mutassuk meg, hogy nem fejezhető ki

a) $\neg$ a $\wedge$, $\vee$, $\rightarrow$ és $\leftrightarrow$ segítségével;

b) $\rightarrow$ a $\wedge$ és $\vee$ segítségével;

c) $\wedge$ a $\vee$ és $\rightarrow$ segítségével.

5.10. Feladat megoldása.

a) Mint az előző példában, a felsorolt függvények $\wedge$, $\vee$, $\rightarrow$ és $\leftrightarrow$ valamennyien 1-tartóak, de a negáció nem az.

b) A megadott függvények 0-tartóak, mert $0 \wedge 0 = 0 \vee 0 = 0$.

   Ezért minden, a halmaz műveleteiből felépített kétváltozós $f(x, y)$ függvényre, $f(0, 0) = 0$.

   De az implikáció nem 0-tartó, hiszen $0 \rightarrow 0 = 1$, ezért az implikáció nem fejezhető ki.

c) Megmutatjuk, hogy azok az $f(x_1, x_2, \ldots, x_n)$ függvények, melyek kifejezhetők $\vee$ és $\rightarrow$ segítségével rendelkeznek a következő tulajdonsággal:

   Létezik i, hogy $x_i = 1$ *esetén* $f(x_1, x_2, \ldots, x_n) = 1$ *mindig teljesül, azaz ,,egyetlen változó igazzá tételével az egész függvény igazzá tehető."*

   Nyilvánvaló, hogy $\vee$ és $\rightarrow$ rendelkezik ezzel a tulajdonsággal.

   Az is könnyen látható, hogy ezt a tulajdonságot a függvénykompozíció megőrzi: A külső függvényt igazzá tevő változó helyére helyettesített belső függvényt egy változóval igazzá téve az összetett függvény is igazzá tehető.

   De $\wedge$ nem rendelkezik ezzel a tulajdonsággal, mert sem az első sem a második változójával nem tehető igazzá, ezért nem fejezhető ki.

5.11. Feladat. Mutassuk meg, hogy a $\{\neg, \leftrightarrow\}$ halmaz *nem adekvát*! (Van olyan kétváltozós Boole-függvény, ami nem fejezhető ki.)

5.11. Feladat megoldása.
1. Megoldás. Mutassuk meg, hogy $\neg$-val és $\leftrightarrow$-val csak olyan kétváltozós Boole-függvények fejezhetők ki, melyek igazságtáblájában páros sok 1-es szerepel, így például $\vee$ nem fejezhető ki.
Valóban, világos, hogy $\leftrightarrow$ igazságtáblája ilyen, és hogy a $\neg$ alkalmazása megőrzi ezt a tulajdonságot. Lássuk be, hogy ha $f_1(x, y)$ és $f_2(x, y)$ ilyen tulajdonságú, akkor $f_1 \leftrightarrow f_2$ is. Valóban, az

$$x \leftrightarrow 1 \equiv x \quad \text{és} \quad x \leftrightarrow 0 \equiv \neg x$$

azonosságok miatt az eredmény, azaz $f_1(x, y) \leftrightarrow f_2(x, y)$ igazságtáblája pontosan azokon a pozíciókon tér el az első argumentum, azaz $f_1(x, y)$ igazságtáblájától, ahol a második argumentum, azaz $f_2(x, y)$ igazságtáblázatában 0 érték található. Ilyen pozícióból azonban páros sok van, ezért páros sok helyen változtattuk meg $f_1(x, y)$ igazságtábláját, így az eredményben továbbra is páros sok 1-es van.
2. Megoldás. Egy Boole-függvényt lineárisnak hívunk, ha $x_1 \oplus x_2 \oplus \ldots \oplus x_n$ vagy $x_1 \oplus x_2 \oplus \ldots \oplus x_n \oplus \uparrow$ alakú. Megmutatjuk, hogy $\neg$-val és $\leftrightarrow$-val csak lineáris függvények fejezhetők ki.
Valóban $\neg x = x \oplus \uparrow$ és $x \leftrightarrow y = x \oplus y \oplus \uparrow$ lineáris. Továbbá, ha f és g lineáris függvények, akkor könnyen látható, hogy $\neg f = f \oplus \uparrow$ és $f \leftrightarrow g = f \oplus g \oplus \uparrow$ szintén lineáris függvények. (Ehhez fel kell használni, hogy $\oplus$ asszociatív és kommutatív, $x \oplus x = \downarrow$ és $x \oplus \downarrow = x$.) De például a konjunkció nem lineáris, ezért nem fejezhető ki.

5.12. Feladat. Legyen F egy olyan formula, melyben csak a $\neg$ műveleti jel szerepel. Lehet-e F tautológia? Indokoljuk a választ!

5.12. Feladat megoldása.
Nem lehet. A csak a negáció műveletét tartalmazó formulák szükségképpen $\underbrace{\neg\neg\ldots\neg}_{k\text{-szor}} x$ alakúak, valamely $k \geq 0$-ra, és így ekvivalensek $\neg x$-szel, (ha k páratlan), vagy x-szel, (ha k páros), de ezek egyike se tautológia.

5.13. Feladat. Legyen F egy olyan formula, melyben csak a $\vee$ műveleti jel szerepel. Lehet-e F tautológia? Indokoljuk a választ!

5.13. Feladat megoldása.
Nem lehet. Mivel $\vee$ 0-tartó, vele is csupán 0-tartó formulák fejezhetők ki, ám a tautológiák nem azok.

5.14. Feladat. Legyen F egy olyan formula, melyben csak a $\rightarrow$ műveleti jel szerepel. Lehet-e F tautológia? Indokoljuk a választ!

5.14. Feladat megoldása. Az F formula lehet tautológia, legyen például $F = x \rightarrow x$.

5.15. Feladat. Legyen F egy olyan formula, melyben csak a $\rightarrow$ műveleti jel szerepel. Lehet F kielégíthetetlen? Indokoljuk a választ!

5.15. Feladat megoldása.
Nem lehet. Mivel $\rightarrow$ 1-tartó, vele is csupán 1-tartó formulák fejezhetők ki, ám a kielégíthetetlen formulák nem azok.

5.16. Feladat. (Craig interpoláció tétele.) Tegyük fel, hogy $\models (F \rightarrow G)$ és, hogy F-ben és G-ben van legalább egy közös ítéletváltozó. Mutassuk meg, hogy ekkor van olyan H, az F és a G közös ítéletváltozóiból felépülő formula, melyre $\models (F \rightarrow H)$ és $\models (H \rightarrow G)$!

5.16. Feladat megoldása. Útmutatás. Az állítás bizonyítása nem egyszerű, de megtehető például az F-ben szereplő, de G-ben nem szereplő ítéletváltozók száma szerinti teljes indukcióval.

5.17. Feladat. Bizonyítsuk be, hogy ha egy klózhalmaz nem tartalmaz pozitív (negatív) klózt, akkor az kielégíthető!

5.17. Feladat megoldása.
Amennyiben a formulában nincsen pozitív klóz, akkor egy kielégítő kiértékelést kapunk, ha minden ítéletváltozó értékét hamisnak választjuk.
Hasonlóan, ha a formulában nincsen negatív klóz, akkor egy kielégítő kiértékelést kapunk, ha minden ítéletváltozó értékét igaznak választjuk.

5.18. Feladat. Igazoljuk, hogy a

$$F = (p_1 \wedge q_1) \vee (p_2 \wedge q_2) \vee \ldots \vee (p_n \wedge q_n)$$

formulával ekvivalens *legrövidebb* CNF hossza $\Omega(2^n)$.

5.18. Feladat megoldása. Az igaz, hogy a disztributivitás alkalmazásával olyan formulát kapunk, mely minden olyan klózt tartalmaz, melyben minden i-re vagy a p_i, vagy a q_i szerepel, aminek hossza $n \cdot 2^n$ (2^n darab, egyenként n hosszú klóz), de *nem ez a kérdés*! Hanem, hogy ennél rövidebb nincs is egyáltalán.

Tegyük fel, hogy $G = C_1 \wedge \ldots \wedge C_m$ egy legrövidebb CNF-je F-nek. Azt tudjuk, hogy nincsenek benne triviális klózok (melyekben egyszerre szerepel egy literál és komplementere), hisz ezt elhagyhatnánk.
Akkor az nem lehet, hogy negatív literál is szerepelne bármelyik klózban: mivel F egy monoton formula, így ha $A(F) = 1$ egy A étékadásra, akkor $A'(F) = 1$ mindig, ha $A' \geq A$, vagyis ha A'-t úgy kapjuk A-ból, hogy benne egy hamisat igazra cserélünk. Mármost ha egy C_i klózban szerepel a $\neg r$ literál, és így igaz A-ra, akkor is igaz marad, ha tetszőleges $A' \geq A$ mellett értékeljük ki. Konkrétan akkor is, ha $A'(r) = 1$ és minden más s változóra $A'(s) = A(s)$. Ekkor C_i-ben $\neg r$ hamis, tehát A' kielégít C_i-ben egy $\neg r$-től különböző literált, így A is kielégíti ezt a literált. Tehát ekkor A már $C - \{\neg r\}$-t is kielégítette. Mivel ez minden A-ra, amire $A(F) = 1$, fennáll, így ha C_i-ből elhagyjuk $\neg r$-t (ezzel a transzformációval új kielégítő kiértékeléseket nem kaphatunk, csak régieket veszíthetünk el), nem vesztünk el kielégítő értékadást, azaz az így kapott CNF G-vel ekvivalens és rövidebb, ellentmondás.
Tehát G-ben csupa pozitív literál szerepel. Az is igaz kell legyen, hogy minden G-beli C klózra és $1 \leq i \leq n$-re vagy p_i, vagy q_i szerepel C-ben. Ugyanis a $p_i = q_i = 1$, minden más 0 értékadás kielégíti F-et. Mivel G a fentiek szerint csak pozitív literálokat tartalmazhat, így ha egy klózból is hiányzik p_i is és q_i is, akkor az a klóz emellett az értékadás mellett hamissá válik, így G is hamis lesz. Ez azt is jelenti, hogy G minden klózában legalább n literál szerepel.
Most vizsgáljuk F-et. Tetszőleges $I \subseteq \{1, \ldots, n\}$-re igaz, hogy ha vesszük azt az A_I értékadást, melyben $p_i = 1$ pontosan akkor, ha $i \in I$ és $q_i = \neg p_i$, akkor (mivel egyszerre nem állítjuk igazra p_i-t és q_i-t) $A_I(F) = 0$. Tehát minden ilyen A_I értékadás hamisra kell állítson legalább egy klózt G-ben. Ez azt jelenti, hogy tetszőleges I-re van olyan C klóz G-ben, mely legfeljebb az A_I szerint nullára beálló n darab literált tartalmazza: $C \subseteq \{p_i : i \notin I\} \cup \{q_i : i \in I\}$. Csakhogy ez a C így egy legfeljebb n-elemű klóz, azt meg fentebb láttuk, hogy minden klóz legalább n-elemű, tehát ez a C pontosan n-elemű kell legyen, vagyis $C = \{p_i : i \notin I\} \cup \{q_i : i \in I\}$ szerepel G-ben minden I-re.
Ez pontosan azt jelenti, hogy a disztributivitással az eredetiből megkapható CNF valóban nem más ebben az esetben, mint a minimális hosszú CNF, és tényleg $\Omega(n \cdot 2^n)$ hosszú.

6. fejezet

A SAT probléma és speciális esetei: 2SAT és HORNSAT.

Elméleti összefoglaló

A SAT probléma, vagyis az ítéletkalkulusbeli formulák kielégíthetőségének vizsgálata a számítástudomány egyik igen fontos, kiemelkedő gyakorlati jelentőségű problémája. Valóban, felhasználva, hogy logikai formulákkal könnyen kifejezhetjük más problémák megoldhatóságát, számos eldöntési és optimalizálási probléma megoldható a SAT segítségével, többek között a mesterséges intelligencia, operációkutatás, programhelyesség bizonyítás és az áramkörök tervezése területén.
Ugyanakkor a SAT az egyik első és legalapvetőbb NP-teljes probléma is. Ez röviden azt jelenti, hogy nem ismerünk rá hatékony (értsd polinom futásidejű) algoritmust, és a bonyolultságelmélet eredményei alapján igen erős sejtésünk, hogy ilyen algoritmus nem is létezik.
Bár az általános SAT nehéz, szerencsére vannak olyan speciális esetei, elsősorban a HORNSAT és a 2SAT, melyek mégis polinom időben megoldhatók. Ebben a fejezetben ezeket tekintjük át.
SAT esetében tetszőleges konjunktív normálformát megadhatunk inputként. Bármely $k \geq 2$ esetén kSAT bemenetei csak olyan konjunktív normálformák lehetnek, amelyben minden klóz pontosan k darab literálból áll. Végül HORNSAT esetén az input mindig Horn-formula kell hogy legyen, azaz olyan konjunktív normálforma, melyben minden klóz legfeljebb egy pozitív literált tartalmaz. Mind a három változat esetében azt kérdezzük, hogy a formula kielégíthető-e. Sőt, az algoritmustól általában nemcsak egy igen/nem választ várunk, hanem pozitív válasz esetén szükségünk van a változók egy, a formulát igazzá tevő kiértékelésre is. Látni fogjuk a megoldó algoritmusok működéséből, hogy ennek megadása nem

okoz nehézséget.

Algoritmus a 2SAT probléma megoldására:

Bemenet: Egy konjunktív normálformájú ítéletkalkulusbeli formula, melyben minden klóz pontosan 2 darab literálból áll.

Kimenet: Egy, a formulát kielégítő változóhozzárendelés, amennyiben ilyen létezik, „NEM" válasz különben.

Algoritmus: Készítsük el a formulához tartozó *implikációs (irányított) gráfot*, az alábbi módon. A gráf csúcsai legyenek a formula változói és azok tagadásai, azaz minden, a formulában szereplő x változóra vegyük fel az $\ell = x$ és $\bar{\ell} = \neg x$ literálokat a csúcsok közé.

Ezután vegyük sorra a formula klózait, és minden $(\ell \vee \ell')$ klózra vegyük fel a gráfba az $(\bar{\ell}, \ell')$ és $(\bar{\ell'}, \ell)$ irányított éleket. Például az $(x \vee \neg y)$ klóz esetén a $\neg x$-ból $\neg y$-ba mutató és az y-ból x-be vezető irányított éleket kell a gráfhoz hozzáadnunk.

Miután a gráf elkészült, segítségével el tudjuk dönteni a formula kielégíthetőségét. A formula akkor és csak akkor kielégíthető, ha a gráfban nincs olyan erősen összefüggő komponens, mely tartalmazza egy ellentétes literál pár mindkét tagját. Másképp fogalmazva, ha nincs olyan x változó, hogy az implikációs gráfban egyszerre igaz, hogy $\neg x$ elérhető irányított úttal x, és x is elérhető $\neg x$-ből.

Amennyiben nincs ilyen változó, egy, a formulát kielégítő $\mathcal{A} : \mathit{Var} \to \{0, 1\}$ változókiértékelést az alábbi módon képezhetünk. Amíg a formulában szereplő minden változó értékét nem rögzítettük ismételjük a következőt:

> Válasszunk egy olyan változót, melynek még nincs értéke. Ha az implikációs gráfban x-ből nem érhető el $\neg x$, akkor legyen $\mathcal{A}(x) = 1$, különben biztosan tudjuk, hogy $\neg x$-ből nem érhető el x, ezért ekkor legyen $\mathcal{A}(x) = 0$. Ezzel az x-et tartalmazó egyik literál (x vagy $\neg x$) értékét igaznak választottuk. Folytassuk a változók értékadását oly módon, hogy minden, az implikációs gráfban igaz értékű literálból elérhető többi literál szintén igaz értéket kapjon.

Horn-formulák Minden konjunktív normálformát átírhatunk úgynevezett *implikációs alakba*, minden klózra a

$$p_1 \vee p_2 \vee \ldots \vee p_m \vee \neg q_1 \vee \neg q_2 \vee \ldots \vee \neg q_n \equiv q_1 \wedge q_2 \wedge \ldots \wedge q_n \to p_1 \vee p_2 \vee \ldots \vee p_m$$

azonosságot alkalmazva. Amennyiben $n = 0$, az implikáció előtagja az üres konjunkció: $\uparrow$, amennyiben pedig $m = 0$, az implikáció utótagja az üres diszjunkció: $\downarrow$. Vegyük észre, hogy mivel Horn-formulák esetén minden klózra definíció szerint $m \leq 1$ teljesül, így csak $q_1 \wedge \ldots \wedge q_n \to p_1$ és $q_1 \wedge \ldots \wedge q_n \to \downarrow$ alakú klózaink lesznek.

Algoritmus a HORNSAT probléma megoldására:

- **Bemenet:**: Egy Horn-formula KNF-ben.

- **Kimenet:** IGEN és egy kielégítő kiértékelés, ha a formula kielégíthető, NEM különben.
  Algoritmus:

- 1. Írjuk a formulát implikációs alakba!
- 2. „**Karikázás**": Amíg van olyan implikáció, amelynek a baloldalán minden változó megjelölt és *a jobboldalán változó áll*, karikázzuk be a jobboldali változót és annak minden előfordulását. Ismételjük, amíg lehet!
- 3. **Döntés**: Ha van olyan *hamis jobboldalú tag*, melynek a baloldalán minden változó megjelölt, a válasz **NEM**.
  Különben a válasz **IGEN**, a formula kielégíthető, mégpedig például úgy, hogy a megjelölt változóknak adjunk IGAZ értéket, a nem megjelölteknek HAMIS-at.

Feladatok

6.1. Feladat. Az alábbi ítéletkalkulusbeli formulákat hozzuk konjunktív normálformára, majd írjuk őket implikációs alakba. Végül írjuk fel a formulákat a rezolúciónál használt halmazos formátumban is. A konjunktív normálalakok közül melyek Horn-formulák? A nem Horn-formulák esetleg egyszerűsítéssel azzá tehetők? Döntsük el, hogy közülük a Horn-formulák kielégíthetők-e.

a) $(\neg p \wedge q) \vee (r \wedge \neg q)$;

b) $p \to ((q \to r) \wedge (\neg s \vee r))$;

c) $(p \wedge q \wedge r) \vee (q \wedge \neg r)$;

d) $(p \vee q \vee \neg p \vee \neg q \vee \neg r) \wedge p \wedge \neg q \wedge (\neg p \vee \neg q)$;

e) $\downarrow$;

f) $\uparrow$.

6.1. Feladat megoldása.

a) $(\neg p \wedge q) \vee (r \wedge \neg q) \equiv (\neg p \wedge q) \vee (r \wedge \neg q) \equiv$
$\equiv (\neg p \vee r) \wedge (\neg p \vee \neg q) \wedge (q \vee r) \wedge (q \vee \neg q) \equiv (\neg p \vee r) \wedge (\neg p \vee \neg q) \wedge (q \vee r) \equiv$
$\equiv (p \to r) \wedge (p \wedge q \to \downarrow) \wedge (\uparrow \to q \vee r) \equiv \{\{\neg p, r\}, \{\neg p, \neg q\}, \{q, r\}\}$.

Ez utóbbi KNF nem Horn-formula, mert a harmadik tag nem Horn-tag (mivel egynél több pozitív literált tartalmaz, q-t és r-et).

b) $p \to ((q \to r) \wedge (\neg s \vee r)) \equiv \neg p \vee ((\neg q \vee r) \wedge (\neg s \vee r)) \equiv (\neg p \vee \neg q \vee r) \wedge (\neg p \vee \neg s \vee r) \equiv (p \wedge q \to r) \wedge (p \wedge s \to r) \equiv \{ \{\neg p, \neg q, r\}, \{\neg p, \neg s, r\} \}$.

Ez a KNF Horn-formula. A formula kielégíthető, legyen minden változó hamis.

c) $(p \wedge q \wedge r) \vee (q \wedge \neg r) \equiv (p \vee q) \wedge (q \vee q) \wedge (r \vee q) \wedge (p \vee \neg r) \wedge (q \vee \neg r) \wedge (r \vee \neg r) \equiv \{ \{p, q\}, \{q\}, \{r, q\}, \{p, \neg r\}, \{q, \neg r\}, \{r, \neg r\} \}$.

A formula nem Horn-formula, de lehetőségünk van egyszerűsíteni, mert $\{r, \neg r\}$ biztosan igaz, és mivel q igaz, a q-t tartalmazó tagok is elhagyhatók, így az marad, hogy

$\{\{q\}, \{p, \neg r\}\} \equiv q \wedge (p \vee \neg r) \equiv (\uparrow \to q) \wedge (r \to p)$.

A formula Horn-formula, és kielégíthető, az algoritmus szerint egy kielégítő kiértékelés a következő: $\mathcal{A}(q) = 1$, $\mathcal{A}(p) = \mathcal{A}(r) = 0$.

d) A formula nem Horn-formula, mert az első klóz két pozitív literált tartalmaz.

e) $\downarrow \equiv \uparrow \to \downarrow \equiv \ \square$ (csak az üres klózt tartalmazó KNF). Ez Horn-formula. Nyilvánvalóan kielégíthetetlen.

f) $\uparrow \equiv p \vee \neg p \equiv p \to p \equiv \{\{p, \neg p\}\}$, Horn-formula, kielégíthető.

Egy másik megoldás: $\emptyset$, vagyis az üres formula, azaz egyetlen klózt sem tartalmazó KNF.

6.2. Feladat. Döntse el a Horn-formulák algoritmusával, hogy kielégíthető-e az alábbi Horn formula:

$$(\neg p \vee \neg q \vee \neg r) \wedge (\neg s \vee \neg v \vee p) \wedge \neg t \wedge s \wedge q \wedge (\neg u \vee v) \wedge u$$

6.2. Feladat megoldása. A formula implikációs alakja a következő

$$(p \wedge q \wedge r \to \downarrow) \wedge (s \wedge v \to p) \wedge (t \to \downarrow) \wedge (\uparrow \to s) \\ \wedge (\uparrow \to q) \wedge (u \to v) \wedge (\uparrow \to u)$$

Az algoritmus sorra az alábbi változók összes előfordulását megjelöli:
s, q, u, v, p.
Azaz végül a következőt kapjuk:

$$(\boxed{p} \wedge \boxed{q} \wedge r \to \downarrow) \wedge (\boxed{s} \wedge \boxed{v} \to \boxed{p}) \wedge (t \to \downarrow) \wedge (\uparrow \to \boxed{s}) \\ \wedge (\uparrow \to \boxed{q}) \wedge (\boxed{u} \to \boxed{v}) \wedge (\uparrow \to \boxed{u})$$

Mivel nincs olyan $(q_1 \wedge q_2 \wedge \cdots \wedge q_m \to\downarrow)$ negatív tag, melynek a baloldalán *minden* változó megjelölt, ezért a formula kielégíthető, és

$$\mathcal{A}(s) = \mathcal{A}(q) = \mathcal{A}(u) = \mathcal{A}(v) = \mathcal{A}(p) = 1, \quad \mathcal{A}(r) = \mathcal{A}(t) = 0$$

igazzá teszi. Az algoritmus által megjelölt változók értéke igaz, a meg nem jelölteké pedig hamis.

6.3. Feladat. Kielégíthető-e az alábbi Horn formula:

$$\begin{aligned}&\neg t \wedge v \wedge (q \vee \neg p) \wedge (\neg p \vee r \vee \neg t) \wedge (t \vee \neg t) \wedge s \wedge (u \vee \neg s \vee \neg p)\\ &\wedge (\neg p \vee \neg r \vee \neg s \vee u) \wedge (\neg p \vee \neg r \vee v) \wedge (\neg q \vee \neg u \vee \neg s \vee \neg w)\\ &\wedge (\neg v \vee \neg t) \wedge (\neg u \vee \neg q \vee w) \wedge (\neg r \vee \neg v \vee \neg q \vee \neg p) \wedge p\end{aligned}$$

6.3. Feladat megoldása. A formulát implikációs alakba írva, majd rajta a Horn-algoritmust végrehajtva azt kapjuk, hogy

$$\begin{aligned}&(t \to\downarrow) \wedge (\uparrow\to \boxed{v}) \wedge (\boxed{p} \to \boxed{q}) \wedge (\boxed{p} \wedge t \to r) \wedge (t \to t) \wedge (\uparrow\to \boxed{s})\\ &\wedge (\boxed{s} \wedge \boxed{p} \to \boxed{u}) \wedge (\boxed{p} \wedge r \wedge \boxed{s} \to \boxed{u}) \wedge (\boxed{p} \wedge r \to \boxed{v})\\ &\wedge (\boxed{q} \wedge \boxed{u} \wedge \boxed{s} \wedge \boxed{w} \to\downarrow) \wedge (\boxed{v} \wedge t \to\downarrow) \wedge (\boxed{u} \wedge \boxed{q} \to \boxed{w})\\ &\wedge (r \wedge \boxed{v} \wedge \boxed{q} \wedge \boxed{p} \to\downarrow) \wedge (\uparrow\to \boxed{p})\end{aligned}$$

Tehát az algoritmus sorra az alábbi változókat jelöli meg:

$$v, s, p, q, u, w.$$

Mivel a $(\boxed{q} \wedge \boxed{u} \wedge \boxed{s} \wedge \boxed{w} \to\downarrow)$ a negatív tagban a baloldalon *minden* változó megjelölt, ezért a formula kielégíthetetlen.

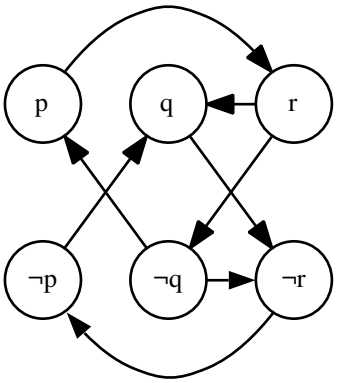
6.4. Feladat. Bizonyítsa be, hogy az alábbi (konjunktív normálformában adott, de nem Horn-formulák) kielégíthetetlenek:

a) $(p \vee q) \wedge (\neg p \vee q) \wedge (p \vee \neg q) \wedge (\neg p \vee \neg q)$;

b) $(p \vee q \vee r \vee s) \wedge (\neg q \vee p) \wedge (\neg r \vee q) \wedge (\neg s \vee r) \wedge (\neg p \vee s) \wedge (\neg p \vee \neg s)$.

6.4. Feladat megoldása.

a) Bármi is egy modellben a p és q változók igazságértéke, biztosan lesz pontosan egy olyan klóz, melyben mindkét literál értéke hamis. Ezért a formula kielégíthetetlen.



6.1. ábra. A 6.68. feladat *a* részének implikációs gráfja

b) Indirekt módon, tegyük fel, hogy van olyan $\mathcal{A} : Var \to \{0, 1\}$ értékadás, mely kielégíti a formulát. Ekkor a $(\neg p \vee s)$ és $(\neg p \vee \neg s)$ miatt $\mathcal{A}(p) = 0$. Ebből a klózok további vizsgálatával $\mathcal{A}(q) = 0$, $\mathcal{A}(r) = 0$, majd $\mathcal{A}(s) = 0$ következik. Ez azonban ellentmond az első klóznak, így ilyen $\mathcal{A}$ kiértékelés nem létezik, vagyis a formula kielégíthetetlen.

6.5. Feladat. Döntse el az implikációs gráf felírásával és elemzésével, hogy kielégíthetők-e az alábbi 2SAT formulák:

a) $(p \vee q) \wedge (\neg r \vee q) \wedge (\neg p \vee r) \wedge (\neg q \vee \neg r)$;

b) $(p \vee q) \wedge (\neg r \vee q) \wedge (\neg p \vee r) \wedge (\neg q \vee \neg r) \wedge (p \vee \neg q)$;

c) $(\neg p \vee \neg p) \wedge (p \vee \neg q) \wedge (\neg q \vee \neg r) \wedge (q \vee q)$;

d) $(\neg p \vee \neg r) \wedge (p \vee q) \wedge (\neg r \vee \neg q) \wedge (\neg t \vee r) \wedge (r \vee t)$;

e) $(\neg p \vee \neg r) \wedge (p \vee q) \wedge (\neg r \vee \neg t) \wedge (\neg t \vee r) \wedge (r \vee t)$;

f) $(p \vee q) \wedge (\neg p \vee q) \wedge (p \vee \neg q) \wedge (\neg p \vee \neg q)$.

6.5. Feladat megoldása.

a) Az implikációs gráf a 6.1 ábrán látható. Vegyük észre, hogy benne p-ből elérhető $\neg p$, $\neg q$-ból elérhető q és r-ből elérhető $\neg r$. Ugyanakkor $\neg p$-ből nem érhető el p, és q-ból nem érhető el $\neg q$, valamint $\neg r$-ből nem érhető le r. Így nincs olyan x változó, hogy x-ből $\neg x$ és $\neg x$-ből x is elérhető lenne, ezért a formula kielégíthető. Mivel p-ből elérhető $\neg p$ ezért p értéke csak hamis lehet, vagyis $\mathcal{A}(p) = 0$. Ebből a gráfban $\neg p$-ből elérhető literálok igaz értéke

következik, vagyis $\mathcal{A}(q) = 1$ és $\mathcal{A}(r) = 0$. Könnyen ellenőrizhető, hogy az így definiált $\mathcal{A}$ kiértékelés valóban kielégíti a formulát.

b) Az utolsó klóz a (q, p) és $(\neg p, \neg q)$ éleket adja hozzá az előző feladatban szereplő implikációs gráfhoz. Ezáltal egyaránt p-ből elérhető $\neg p$ és $\neg p$-ből elérhető p. Így a formula kielégíthetetlen.

c) Az implikációs gráfban például p-ből elérhető $\neg p$ és $\neg p$-ből elérhető p, ezért a formula kielégíthetetlen.

d) Az implikációs gráfban például p-ből elérhető $\neg p$ és fordítva $\neg p$-ből is elérhető p. (Ez most éppen bármely más változóra is igaz.) Így a formula kielégíthetetlen.

e) Az implikációs gráfból ezúttal nem következik ellentmondás, ezért a formula kielégíthető. Mivel p-ből elérhet $\neg p$, ezért csak $\mathcal{A}(p) = 0$ lehetséges, amiből $\mathcal{A}(q) = 1$, $\mathcal{A}(t) = 0$ és $\mathcal{A}(r) = 1$ következik.

f) Az implikációs gráfban p-ből elérhető $\neg p$ és fordítva, $\neg p$-ből is elérhető p, ezért a formula kielégíthetetlen. Ezentúl érdemes észrevenni, hogy a gráfban minden él ellentétes párja is szerepel.

7. fejezet

A rezolúciós módszer

Elméleti összefoglaló

A rezolúció az egyik legfontosabb logikai bizonyítási módszer egyrészt alapvető az automatikus tételbizonyítási módszerek között, másrészt ez a logikai programozás (a Prolog programok) működésének elméleti háttere.
Ez a fejezet az ítéletkalkulusbeli rezolúciót mutatja be, az elsőrendű rezolúciót később tárgyaljuk.
Hogy a halmazelméleti műveleteket alkalmazni tudjuk, a rezolúció során minden konjunktív normálformájú formulát azonosítunk a formula klózainak halmazával, magukat a klózokat pedig az őket alkotó literálok halmazával. Így például a $(x \vee \neg y) \wedge (z \vee \neg x \vee z)$ formula $\{\{x, \neg y\}\,\{\neg x, z\}\}$ lesz. A rezolúcióban kiemelt szerepe van az üres klóznak, mely egyetlen literált sem tartalmaz, ennek jele: $\square$. Mivel az üres klóz üres diszjunkció, ezért kielégíthetetlen, sőt nyilvánvalóan kielégíthetetlen minden üres klózt tartalmazó konjunktív normálforma is.
A rezolúció szabálya:
Legyenek C_1 és C_2 zérusrendű klózok. Amennyiben $\ell \in C_1$, és $\bar{\ell} \in C_2$ teljesül valamely ℓ literálra, akkor képezhetjük a C_1 és C_2 klózok egy rezolvensét, az alábbi szabállyal:

$$\frac{C_1, \quad C_2}{(C_1 \setminus \{\ell\}) \cup (C_2 \setminus \{\bar{\ell}\})}.$$

A rezolvensképzést halmazokra is kiterjesztjük, tetszőleges Σ (véges vagy végtelen) klózhalmazt kibővítve, annak rezolvenseivel:

$$\mathrm{Res}(\Sigma) = \Sigma \cup \{R \mid R \text{ valamely } C_1 \text{ és } C_2 \in \Sigma \text{ klózok egy rezolvense.}\}$$

Ezt a klózok halmazán működő operátort természetes módon iterálhatjuk:

$$\mathrm{Res}^n(\Sigma) = \begin{cases} \Sigma, & \text{ha } n = 0, \\ \mathrm{Res}(\mathrm{Res}^{n-1}(\Sigma)), & \text{ha } n > 0. \end{cases}$$

Jelölje $\mathrm{Res}^*(\Sigma)$ a Σ elemeiből akárhány lépésben megkapható klózok halmazát, azaz

$$\mathrm{Res}^*(\Sigma) = \bigcup_{n=0}^{\infty} \mathrm{Res}^n(\Sigma).$$

Belátható, hogy $\mathrm{Res}^*(\Sigma)$ az a legszűkebb halmaz, mely Σ-t tartalmazza és zárt a rezolvensképzésre.
A rezolúciós bizonyítás helyességét és teljességét az alábbi tétel fogalmazza meg.
A rezolúció alaptétele. Bármely Σ klózhalmaz kielégíthetetlen $\Leftrightarrow \square \in \mathrm{Res}^*(\Sigma)$.
Ezt felhasználva a következő algoritmushoz jutunk. Először a kielégíthetőség szempontjából vizsgálni kívánt formulát vagy formulahalmazt konjunktív normálformára hozzuk, majd a klózokat a Σ formulahalmazba gyűjtjük. Ezután Σ, és ezzel az eredeti formula vagy formulahalmaz kielégíthetőségét az alábbi módon dönthetjük el:
A rezolúció algoritmusa:
Bemenet: Σ ítéletkalkulusbeli klózok halmaza
Kimenet: IGEN, ha Σ kielégíthető, NEM különben.
Algoritmus:

1. $R := \Sigma$
2. $R' := \mathrm{Res}(R)$
3. **Ha** $\square \in R'$ **akkor** VÉGE, a válasz NEM.
4. **Ha** $R = R'$ **akkor** VÉGE, a válasz IGEN, **különben** $R := R'$, és **ismételd** 2-től.

Vegyük észre, hogy amennyiben a Σ formulahalmaz kielégíthetetlen, nem szükséges a $\mathrm{Res}^i(\Sigma)$ halmaz minden elemét sorra előállítanunk, elég csupán az üres klóz egy levezetését megadnunk. Ez az alábbi definíció szerint történhet:
A rezolúciós levezetés: Legyen Σ klózok halmaza. Klózok egy véges $C_0, \ldots, C_n$ sorozatát Σ feletti *rezolúciós levezetésnek* (vagy *bizonyításnak*) nevezzük, ha $C_n = \square$ és minden C_k, ahol $(1 \leq k \leq n)$, klózhoz tartozik egy indoklás, mely az alábbiak valamelyike

- „$\in \Sigma$”: azaz C_k szerepel a Σ halmazban
- „Res i, j”: azaz C_k rezolúcióval kapható a levezetésben már korábban szereplő C_i és C_j klózokból, azaz $i, j < k$.

A rezolúciós levezetés tétele: Bármely Σ klózhalmaz kielégíthetetlen $\Leftrightarrow$ létezik Σ feletti rezolúciós bizonyítás (melynek utolsó eleme természetesen $\square$).

Feladatok

7.1. Feladat. Bizonyítsuk be rezolúcióval, hogy a következő formulák *nem kielégíthetők*.

a) $(p \equiv (q \to r)) \wedge ((p \equiv q) \wedge (p \equiv \neg r))$;

b) $\neg(((p \to q) \to \neg q) \to \neg q)$;

c) $\neg(r \wedge \neg q) \wedge (\neg p \to \neg q) \wedge (\neg p \vee \neg s) \wedge \neg(\neg r \vee \neg s)$.

7.1. Feladat megoldása.
Csak a feladat b) részének megoldását ismertetjük részletesen, a többi feladatrész ugyanilyen lépéseken keresztül oldható meg.
A formulát először KNF-re kell hozni: $F = \neg(((p \to q) \to \neg q) \to \neg q) \equiv ((p \to q) \to \neg q) \wedge q \equiv (\neg(p \to q) \vee \neg q) \wedge q \equiv ((p \wedge \neg q) \vee \neg q) \wedge q \equiv (p \vee \neg q) \wedge \neg q \wedge q$
Így F mint klózok halmaza: $F = \{\{p, \neg q\}, \{\neg q\}, \{q\}\}$.
Az üres klóz egy rezolúciós levezetés az alábbi:

1. $\{p, \neg q\}$	$\in F$
2. $\{\neg q\}$	$\in F$
3. $\{q\}$	$\in F$
4. $\square$	Res $2, 3$

Így rezolúcióval levezethető az üres klóz: $\square$, ezért F kielégíthetetlen. Ha az üres klóz nem lenne levezethető (ezt észrevesszük, mert egy idő után nem tudunk rezolúcióval újabb klózokat képezni, amikor már előállítottuk $\mathrm{Res}^*(\Sigma)$ miden elemét), akkor F kielégíthető lenne.

7.2. Feladat. Adjuk meg a $\mathrm{Res}^0(F)$, $\mathrm{Res}^1(F)$, $\mathrm{Res}^2(F)$, $\mathrm{Res}^3(F)$ és $\mathrm{Res}^*(F)$ klózhalmazt! Kielégíthető-e F?

a) $F = \{\{\neg p, q, \neg r\}, \{p\}, \{q, r, s\}, \{\neg r, \neg s\}\}$;

b) $F = \{\{\neg p, \neg q, r\}, \{p, r\}, \{q, r\}, \{\neg r\}\}$.

7.2. Feladat megoldása.
b) megoldása.

1. $\{\neg p, \neg q, r\}$	$\in F$	
2. $\{p, r\}$	$\in F$	
3. $\{q, r\}$	$\in F$	
4. $\{\neg r\}$	$\in F$	
		$1-4. = \mathrm{Res}^0(F)$
5. $\{\neg q, r\}$	Res $1, 2$	
6. $\{\neg p, r\}$	Res $1, 3$	
7. $\{\neg p, \neg q\}$	Res $1, 4$	
8. $\{p\}$	Res $2, 4$	
9. $\{q\}$	Res $3, 4$	
		$1-9. = \mathrm{Res}^1(F)$
10. $\{r\}$	Res $2, 6$	
11. $\{\neg q\}$	Res $4, 5$	
12. $\{\neg p\}$	Res $4, 6$	
		$1-12. = \mathrm{Res}^2(F)$
13. $\square$	Res $4, 10$	
		$1-13. = \mathrm{Res}^3(F) = \mathrm{Res}^*(F)$

Könnyen látható, hogy $\mathrm{Res}^4(F) = \mathrm{Res}^3(F)$, így $\mathrm{Res}^*(F) = \mathrm{Res}^3(F)$, és $\square \in \mathrm{Res}^*(F)$, ezért F kielégíthetetlen.

7.3. Feladat. Döntsük el rezolúcióval, hogy a következő formulák tautológiák-e?

a) $\neg[(p \to q) \wedge p \wedge \neg q]$;

b) $((p \to q) \wedge \neg q) \vee \neg p$;

c) $(p \to q) \to ((q \to r) \to (p \to r))$;

d) $(p \to r) \to ((q \to r) \to (p \vee q \to r))$.

7.3. Feladat megoldása.
Minden esetben a bizonyítás alapja, az hogy F tautológia $\Leftrightarrow \neg F$ kielégíthetetlen.

a) $\neg F \equiv (\neg p \vee q) \wedge p \wedge \neg q = \{\{\neg p, q\}, \{p\}, \{\neg q\}\}$.

1. $\{\neg p, q\} \quad \in F$
2. $\{p\} \quad \in F$
3. $\{\neg q\} \quad \in F$
4. $\{q\} \quad$ Res 1,2
5. $\square \quad$ Res 3,4

Így $\neg F$ kielégíthetetlen, ezért F **tautológia**.

b) **Nem tautológia.**

c) **Tautológia.**

d) **Tautológia.**

7.4. Feladat. Döntsük el rezolúcióval hogy az alábbi formulák ekvivalensek-e?

a) $p \to \neg q$ és $q \to \neg p$;

b) $(p \vee q \to r)$ és $(\neg r \to \neg p \vee \neg q)$.

7.4. Feladat megoldása. Minden esetben a bizonyítás alapja, az hogy $F \equiv G \Leftrightarrow \models F \leftrightarrow G \Leftrightarrow \neg(F \leftrightarrow G)$ kielégíthetetlen. Ezért a $\neg(F \leftrightarrow G)$ formulát kell konjunktív normálformára hoznunk.

a) **Ekvivalensek.**

b) Belátható, hogy $\neg(F \leftrightarrow G) \equiv (\neg p \vee \neg q) \wedge (p \vee q) \wedge \neg r$. Így

$$\Sigma := \{\{\neg p, \neg q\}, \{p, q\}, \{\neg r\}\}$$

kielégíthetetlenségét kell bizonyítanunk.

1. $\{\neg p, \neg q\} \quad \in \Sigma$
2. $\{p, q\} \quad \in \Sigma$
3. $\{\neg r\} \quad \in \Sigma$
4. $\{\neg q, q\} \quad$ Res $1, 2$
5. $\{\neg p, p\} \quad$ Res $1, 2$

1-5. $= \mathrm{Res}^1(\Sigma) = \mathrm{Res}^2(\Sigma) = \mathrm{Res}^*(\Sigma)$.

Mivel most $\square \notin \mathrm{Res}^*(\Sigma)$, ezért Σ kielégíthető, így F és G nem ekvivalensek.

7.5. Feladat. Döntsük el rezolúcióval hogy az alábbi logikai következmények fennállnak-e?

a) $\{q, r \to (q \to p), \} \models r \to p$;

b) $\{q \to r, r \to p, q \vee t, t \to (s \to p), \neg p\} \models \neg q \wedge \neg r$;

c) $\{q \to r, r \to p, q \vee t, t \to (s \to p), \neg p\} \models \neg t$;

d) $\{u \to w \vee q, \neg q, u\} \models w$.

7.5. Feladat megoldása. A megoldás alapja minden esetben a következő tétel: $\{F_1, F_2, \ldots, F_n\} \models G \Leftrightarrow \Sigma = \{F_1, F_2, \ldots, F_n, \neg G\}$ kielégíthetetlen.

a) Igaz.

$F_1 = \underbrace{q}$, $F_2 = r \to (q \to p) \equiv \underbrace{\neg r \vee \neg q \vee p}$, $G = r \to p$ és $\neg G = \neg(r \to p) \equiv \underbrace{r} \wedge \underbrace{\neg p}$

Így most

$$\Sigma = \{\{q\}, \{\neg r, \neg q, p\}, \{r\}, \{\neg p\}\}.$$

kielégíthetetlenségét kell igazolnunk.

Az üres klóz egy rezolúciós levezetése:

1. $\{q\}$	$\in \Sigma$
2. $\{\neg r, \neg q, p\}$	$\in \Sigma$
3. $\{r\}$	$\in \Sigma$
4. $\{\neg p\}$	$\in \Sigma$
5. $\{\neg r, p\}$	Res $1, 2$
6. $\{p\}$	Res $3, 5$
7. $\square$	Res $4, 6$

Ezért Σ kielégíthetetlen, ami azt mutatja, hogy a logikai következtetés helyes.

b) Igaz.

c) Nem igaz.

d) Igaz.

7.6. Feladat. Formalizálja az alábbi mondatokat és döntse el rezolúcióval, hogy az első két mondatnak logikai következménye-e a harmadik.

- F_1 : Ha Peti busszal utazik és a busz késik, akkor Peti nem ér oda a találkozóra.
- F_2 : Petinek nem kell hazamennie, ha nem ér oda a találkozóra és ha rosszkedvű.
- F_3 : Ha Petinek haza kell mennie, és Peti busszal utazik, akkor Peti nem lesz rosszkedvű, ha késik a busz.

7.6. Feladat megoldása. Vezessük be az alábbi ítéletváltozókat

- B = „ Peti busszal utazik."
- K = „ A busz késik."
- O = „ Peti odaér a találkozóra."
- H = „ Petinek haza kell mennie."
- R = „ Peti rosszkedvű (lesz)."

Ekkor az állítások egy lehetséges formalizálása

- $F_1 = B \wedge K \to \neg O \equiv \neg B \vee \neg K \vee \neg O$;
- $F_2 = \neg O \wedge R \to \neg H \equiv O \vee \neg R \vee \neg H$;
- $F_3 = (H \wedge B) \to (K \to \neg R)$;
- $\neg F_3 = H \wedge B \wedge K \wedge R$;

Végül, a klózhalmaz, melynek kielégíthetetlenségét igazolnunk kell

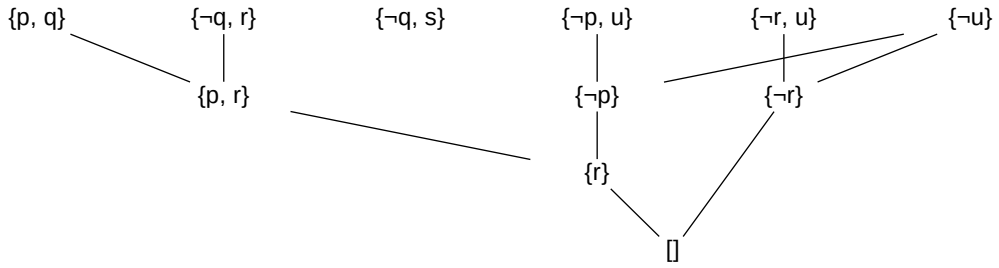
$$\Sigma = \{\{\neg B, \neg K, \neg O\}, \{O, \neg R, \neg H\}, \{H\}, \{B\}, \{K\}, \{R\}\}$$

Innen az üres klóz levezethetősége könnyen látható. Így a logikai következtetés helyes.

7.7. Feladat. Bizonyítsa be rezolúcióval a következőket:

a) $\{\, p \vee q,\, q \to r \wedge s,\, p \vee r \to u \,\} \models u$;

b) $\{\, p \vee (q \wedge r),\, q \to s,\, p \to q \,\} \models r \to s$;



7.1. ábra. A 7.75. feladat *a* klózaiból az üres klóz rezolúciós levezetése

c) $\{\,(q \wedge p) \to \neg r,\ p \vee q,\ r,\ p \to q\,\} \models \neg(q \to p)$;

d) $\{\,p,\ p \to (q \vee r) \wedge \neg(q \wedge r),\ p \to (s \vee t) \wedge \neg(s \wedge t),\ s \to q,\ \neg r \to t\,\} \models t \wedge \neg s$;

e) $\models (p \to (q \to r)) \to ((p \to q) \to (p \to r))$;

f) $\{\,r \to s \vee t,\ \neg s \wedge \neg u,\ \neg u \to \neg t,\ \neg p\,\} \models (\neg p \vee q) \wedge \neg r$;

g) $(q \to p \vee s) \wedge (h \to (r \vee t)) \wedge (q \vee h) \wedge \neg(p \vee s) \wedge \neg t \models r$.

7.7. Feladat megoldása.

a) Csak ennek a résznek a megoldását ismertetjük részletesen, a többi rész ugyanilyen lépéseken keresztül oldható meg.

$\{\,p \vee q,\ q \to r \wedge s,\ p \vee r \to u\,\} \models u$ akkor és csak akkor teljesül, ha $\{\,p \vee q,\ q \to r \wedge s,\ p \vee r, \neg u\,\}$ kielégíthetetlen formulahalmaz. Ezért ennek elemeit kell konjunktív normálformára hozznunk.

$p \vee q$ már KNF;
$q \to r \wedge s \equiv \neg q \vee (r \wedge s) \equiv (\neg q \vee r) \wedge (\neg q \vee s)$;
$p \vee r \to u \equiv \neg(p \vee r) \vee u \equiv (\neg p \wedge \neg r) \vee u \equiv (\neg p \vee u) \wedge (\neg r \vee u)$;
$\neg u$ ez is KNF.

Így az ezekből képzett klózhalmaz, melynek kielégíthetetlenségét igazolnunk kell

$$\{\,\{p, q\}, \{\neg q, r\}, \{\neg q, s\}, \{\neg p, u\}, \{\neg r, u\}, \{\neg u\}\}.$$

Az üres klóz egy levezetését mutatja a 7.1. ábra, a levezetés menetét pedig a **2. animáció** szemlélteti.

b) A továbbiakban, egy kivételtől eltekintve, csak a KNF-kból származó klózhalmazokat adjuk meg, azokból az üres klóz levezetése már könnyen elvégezhető. Ezen feladat esetében a klózhalmaz

$$\{\{p, q\}, \{p, r\}, \{\neg q, s\}, \{\neg p, s\}, \{r\}, \{\neg s\}\}.$$

c) A rezolválandó klózok halmaza

$$\{\{\neg p, \neg q, \neg r\}, \{p, q\}, \{r\}, \{\neg p, q\}, \{\neg q, p\}\}.$$

d) A rezolválandó klózok halmaza

$$\Sigma := \{\{p\}, \{\neg p, q, r\}, \{\neg p, \neg q, \neg r\}, \{\neg p, s, t\}, \{\neg p, \neg s, \neg t\}, \{\neg s, q\}, \{r, t\}, \{\neg t, s\}\}.$$

Mivel itt az üres klóz levezetése sem magától érthetődő, megadunk egy levezetést is:

1.	$\{p\}$	$\in \Sigma$
2.	$\{\neg p, q, r\}$	$\in \Sigma$
3.	$\{\neg p, \neg q, \neg r\}$	$\in \Sigma$
4.	$\{\neg p, s, t\}$	$\in \Sigma$
5.	$\{\neg p, \neg s, \neg t\}$	$\in \Sigma$
6.	$\{\neg s, q\}$	$\in \Sigma$
7.	$\{r, t\}$	$\in \Sigma$
8.	$\{\neg t, s\}$	$\in \Sigma$
9.	$\{\neg p, \neg q, t\}$	Res 3, 7
10.	$\{\neg p, q, t\}$	Res 4, 6
11.	$\{\neg p, t\}$	Res 9, 10
12.	$\{\neg p, \neg t\}$	Res 5, 8
13.	$\{\neg p\}$	Res 11, 12
14.	$\square$	Res 1, 13

e) A rezolválandó klózok halmaza

$$\Sigma := \{\{\neg p, \neg q, r\}, \{\neg p, q\}, \{p\}, \{\neg r\}\}.$$

f) A rezolválandó klózok halmaza

$$\Sigma := \{\{\neg r, s, t\}, \{\neg s\}, \{\neg u\}, \{u, \neg t\}, \{\neg p\}, \{p, r\}, \{\neg q, r\}\}.$$

g) A rezolválandó klózok halmaza

$$\Sigma := \{\{\neg q, p, s\}, \{\neg h, r, t\}, \{q, h\}, \{\neg p\}, \{\neg s\}, \{\neg t\}, \{\neg r\}\}.$$

7.8. Feladat. A Horn-formulákra megismert ,,karikázós" algoritmus implementálható lineáris időigényben. A rezolúcióban a szelekciós operátort (mely eldönti, hogy mely klózok rezolvensét képezzük a következő lépésben) meg tudja-e úgy adni, hogy Horn-formulákra a rezolúciós algoritmus is lineáris időben működjön?

7.8. Feladat megoldása. Ha jobban megnézzük a ,,karikázós" algoritmust, a következőt tesszük:

- Keresünk egy pozitív egységklózt, vagyis egy $\{p\}$ alakú klózt.
- A klózt eldobjuk és $\neg p$-t töröljük minden klózból.
- Ha üres klózhoz érünk, jelentjük, hogy az input KIELÉGÍTHETETLEN.
- Egyébként jelentjük, hogy KIELÉGÍTHETŐ (a karikázós algoritmus egy kielégítő értékadást is szállít ekkor).

A fenti formalizmus szerint ez nem más, mint ha a rezolúcióban pozitív egységklózokkal rezolválnánk, ameddig csak tudunk; ha tehát ezt a szelekciót követjük, akkor a rezolúciós motorunk a Horn-formulákon is felveszi a versenyt a karikázós algoritmussal.
Megjegyzés: a pozitivitási feltétel szükségtelen, egy rezolúciós motorban *mindig* érdemes egységklózokkal rezolválni, amikor csak lehet. Helyes lépés a *subsumption* is, mely szerint ha $C_1 \subsetneq C_2$ és mindkét klózt levezettük, akkor C_2-t eldobhatjuk, mert az üres klóz egy legrövidebb levezetésében már nem lesz rá szükség. Ily módon ha az egységklózzal ,,kiütöttünk" egy literált egy klózból, úgy ott tényleg ,,kiütés" történik és az eredeti klózt eldobhatjuk. Ezek után az egységklózra már nem lesz szükség, hiszen mindent rezolváltunk vele, amit csak lehetett.

8. fejezet

A Hilbert-kalkulus

Elméleti összefoglaló

Minden bizonyítási (levezetési) rendszer *axiómákból* és *következtetési szabályokból* áll. Itt most csak az egyik legegyszerűbb esetet mutatjuk be. Csak olyan ítéletkalkulusbeli formulákat vizsgálunk, melyek csak az implikáció ($\rightarrow$) és az azonosan hamis ($\downarrow$) műveleteket tartalmazzák. Ez nem jelent lényeges megszorítást, mert, mint korábban már láttuk, $\{\rightarrow, \downarrow\}$ teljes rendszert alkot, azaz segítségükkel az ítéletkalkulus bármely művelete kifejezhető. A rendszer építőelemei esetünkben az alábbiak:
Axióma sémák:

a) $(F \rightarrow (G \rightarrow H)) \rightarrow ((F \rightarrow G) \rightarrow (F \rightarrow H))$;

b) $F \rightarrow (G \rightarrow F)$;

c) $((F \rightarrow \downarrow) \rightarrow \downarrow) \rightarrow F$.

Ezek axióma *sémák*, azaz a fenti képletekben F, G és H helyére tetszőleges formulákat helyettesíthetünk.

Következtetési szabály:
Modus Ponens (leválasztás szabálya, MP)

$$\frac{F, F \rightarrow G}{G}$$

Ezt szintén tetszőleges F és G formulákra szabad alkalmazni.
Hogyan lehet ezeket megtanulni?
AX1: $(F \rightarrow (G \rightarrow H)) \rightarrow ((F \rightarrow G) \rightarrow (F \rightarrow H))$,
Ez az ,,implikáció öndisztributivitása": ,,$F \cdot (G + H) = F \cdot G + F \cdot H$".

A „láncolt implikációk” viselkedése miatt: $F \to (G \to H) \equiv F \wedge G \to H$, így az axiómával ekvivalens $(F \wedge G \to H) \wedge (F \to G) \to (F \to H)$, ennek igazsága pedig már könnyen látható.
AX2: $F \to (G \to F)$
Ez arra jó, hogy F-ből, $G \to F$-et készíthessünk. Valóban:

a) F

b) $F \to (G \to F)$ AX2

c) $G \to F$ MP 1,2

Ez alapján, ha már levezettük F-et, levezethetjük $G \to F$-et bármely G formulára.
AX3: $((F \to\downarrow) \to\downarrow) \to F$
Ez a dupla tagadás törvénye, vagyis $\neg\neg F \to F$.
A tautológiák bizonyítása Hilbert rendszerében
A csak $\to$ és $\downarrow$ műveleteket tartalmazó F formula érvényességének *Hilbert-féle bizonyítása* vagy *levezetése* alatt egy olyan $F_1, F_2, \ldots, F_n$ véges sorozatot értünk, amelyben $F_n = F$, és a sorozatban minden F_k $(1 \leq k \leq n)$ lépéshez tartozik egy indoklás, mely az alábbiak valamelyike

- **AXi**, azaz F_k az $i.$ axiómasémából kapható helyettesítéssel $(1 \leq i \leq 3)$, vagy
- **MP i,j**, azaz F_k, a korábbi F_i és F_j formulákból kapható az MP-szabállyal, ahol $i, j < k$.

Ha F levezethető, akkor ennek jele $\vdash_{\mathcal{H}} F$, vagy egyszerűen csak $\vdash F$.
A logikai következtetés bizonyítása
Ekkor az F bizonyításához az axiómákon túl a Σ-beli formulák is felhasználhatók (azok persze helyettesítés nélkül). Ennek jele: $\Sigma \vdash_{\mathcal{H}} F$ vagy $\Sigma \vdash F$.
Tétel. (A Hilbert-kalkulus helyessége és teljessége) Bármely Σ és F zérusrendű formulahalmazra illetve formulára

$$\Sigma \vdash F \Leftrightarrow \Sigma \models F$$

Megjegyzés. Némileg bonyolultabb axiómákkal és szabályokkal ugyan, de a tétel igaz elsőrendben is. Így a logikai következmény fogalma formalizálható.
A Hilbert-féle bizonyítások jóval egyszerűsíthetők az alábbi tétel alkalmazásával.
Dedukció tétel:

$$\Sigma \vdash F \to G \Leftrightarrow \Sigma \cup \{F\} \vdash G$$

Feladatok

8.1. Feladat. Bizonyítsuk Hilbert rendszerében dedukció használata nélkül, hogy $\vdash p \to p$.

8.1. Feladat megoldása.

1. $p \to ((p \to p) \to p)$
   AX2$[F/p, G/(p \to p)]$
2. $(p \to ((p \to p) \to p)) \to ((p \to (p \to p)) \to (p \to p))$
   AX1$[F/p, G/(p \to p), H/p]$
3. $(p \to (p \to p)) \to (p \to p)$
   MP 1,2
4. $p \to (p \to p)$
   AX2$[F/p, G/p]$
5. $p \to p$
   MP 3,4

8.2. Feladat. Bizonyítsuk Hilbert rendszerében dedukció használata nélkül, hogy $\Sigma = \{(p \to\downarrow) \to\downarrow, p \to q\}$ esetén $\Sigma \vdash q$.

8.2. Feladat megoldása.

1.	$(p \to\downarrow) \to\downarrow$	$\in \Sigma$
2.	$((p \to\downarrow) \to\downarrow) \to p$	AX3$[F/p]$
3.	p	MP 1,2
4.	$p \to q$	$\in \Sigma$
5.	q	MP 3,4

8.3. Feladat. Bizonyítsuk Hilbert rendszerében dedukció használata nélkül, hogy $\vdash\downarrow\to p$, (azaz ,,hamisból minden következik")!

8.3. Feladat megoldása.

1. $(\downarrow\to (((p \to\downarrow) \to\downarrow) \to p)) \to ((\downarrow\to ((p \to\downarrow) \to\downarrow)) \to (\downarrow\to p))$
   AX1$[F/\downarrow, G/(p \to\downarrow) \to\downarrow, H/p]$
2. $((p \to\downarrow) \to\downarrow) \to p$ AX3$[F/p]$
3. $(((p \to\downarrow) \to\downarrow) \to p) \to (\downarrow\to (((p \to\downarrow) \to\downarrow) \to p))$
   AX2$[F/((p \to\downarrow) \to\downarrow) \to p, G/\downarrow]$
4. $\downarrow\to (((p \to\downarrow) \to\downarrow) \to p)$ MP 2,3

5. $(\downarrow\rightarrow((p\rightarrow\downarrow)\rightarrow\downarrow))\rightarrow(\downarrow\rightarrow p)$ MP 1,4

6. $\downarrow\rightarrow((p\rightarrow\downarrow)\rightarrow\downarrow)$ AX2$[F/\downarrow, G/p\rightarrow\downarrow]$

7. $\downarrow\rightarrow p$ MP 5,6

8.4. Feladat. Bizonyítsuk Hilbert rendszerében dedukció használata nélkül, hogy $\vdash((p\rightarrow\downarrow)\rightarrow p)\rightarrow((p\rightarrow\downarrow)\rightarrow\downarrow)$!

8.4. Feladat megoldása.

a) $((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow))\rightarrow(((p\rightarrow\downarrow)\rightarrow p)\rightarrow((p\rightarrow\downarrow)\rightarrow\downarrow))$
AX1$[F/(p\rightarrow\downarrow), G/p, H/\downarrow]$

b) $((p\rightarrow\downarrow)\rightarrow(((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow))\rightarrow(p\rightarrow\downarrow)))\rightarrow$
$(((p\rightarrow\downarrow)\rightarrow((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow)))\rightarrow((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow)))$
AX1$[F/p\rightarrow\downarrow, G/((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow)), H/(p\rightarrow\downarrow)]$

c) $(p\rightarrow\downarrow)\rightarrow(((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow))\rightarrow(p\rightarrow\downarrow))$
AX2$[F/p\rightarrow\downarrow, G/(p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow)]$

d) $((p\rightarrow\downarrow)\rightarrow((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow)))\rightarrow((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow))$ MP 2,3

e) $(p\rightarrow\downarrow)\rightarrow((p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow))$
AX2$[F/p\rightarrow\downarrow, G/p\rightarrow\downarrow]$

f) $(p\rightarrow\downarrow)\rightarrow(p\rightarrow\downarrow)$ MP 4,5

g) $((p\rightarrow\downarrow)\rightarrow p)\rightarrow((p\rightarrow\downarrow)\rightarrow\downarrow)$ MP 1,6

8.5. Feladat. Bizonyítsuk a dedukció tétel ismételt alkalmazásával, hogy $\vdash F\rightarrow((F\rightarrow\downarrow)\rightarrow\downarrow)$.

8.5. Feladat megoldása. A dedukció tétel szerint mindaddig, amíg implikációt, azaz $F\rightarrow G$ alakú formulát kell bebizonyítani, annak előtagja, F felvehető a feltételek közé és elég csak G-t bizonyítani.
Esetünkben

$$\vdash F\rightarrow((F\rightarrow\downarrow)\rightarrow\downarrow)\Leftrightarrow\{F\}\vdash(F\rightarrow\downarrow)\rightarrow\downarrow\Leftrightarrow\{F, F\rightarrow\downarrow\}\vdash\downarrow$$

Ezt pedig már könnyű bizonyítani:

1.	F	$\in \Sigma$
2.	$F \to\downarrow$	$\in \Sigma$
3.	$\downarrow$	MP 1,2

Megjegyzés. A dedukció tétel bizonyítása konstruktív, így a fenti bizonyításból algoritmussal elő tudjuk állítani az eredetileg bizonyítandó formula egy levezetését.

8.6. Feladat. Mutassuk meg az alábbiakat, a dedukció tétel használható.

a) $\vdash F \to F$;

b) $\vdash ((p \to\downarrow) \to p) \to ((p \to\downarrow) \to\downarrow)$;

c) $\vdash (p \to (p \to\downarrow)) \to (p \to\downarrow)$;

d) $\vdash\downarrow\to p$;

e) $\vdash (p \to\downarrow) \to (p \to q)$;

f) $\vdash ((F \to\downarrow) \to (G \to\downarrow)) \to (((F \to\downarrow) \to G) \to F)$;

g) $\{(p \to\downarrow) \to (q \to\downarrow)\} \vdash q \to p$;

h) $\{(p \to\downarrow) \to p\} \vdash p$

8.6. Feladat megoldása.

a) $\vdash F \to F$ dedukcióval triviális:

1. F, mert F feltétel.

b) Ez egy korábban már szereplő példa, de dedukcióval jóval könnyebben igazolható:
$\vdash ((p \to\downarrow) \to p) \to ((p \to\downarrow) \to\downarrow) \Leftrightarrow$
$\{(p \to\downarrow) \to p\} \vdash ((p \to\downarrow) \to\downarrow) \Leftrightarrow$
$\{(p \to\downarrow) \to p, p \to\downarrow\} \vdash\downarrow$

Ennek bizonyítása pedig, ha $\Sigma = \{(p \to\downarrow) \to p, p \to\downarrow\}$ a feltételek halmaza:

1. $(p \to\downarrow) \to p$	$\in \Sigma$
2. $p \to\downarrow$	$\in \Sigma$
3. p	MP $1, 2$
4. $\downarrow$	MP $2, 3$

c) Az előző részfeladathoz teljesen hasonlóan bizonyítható.

d) Ez is jóval könnyebb dedukcióval, bár nem triviális:
$\vdash \downarrow \rightarrow p \Leftrightarrow \{\downarrow\} \vdash p$, melynek egy bizonyítása

1.	$\downarrow$	$\in \Sigma$
2.	$((p \rightarrow \downarrow) \rightarrow \downarrow) \rightarrow p$	AX3$[F/p]$
3.	$\downarrow \rightarrow ((p \rightarrow \downarrow) \rightarrow \downarrow)$	AX2$[F/\downarrow, G/p \rightarrow \downarrow]$
4.	$(p \rightarrow \downarrow) \rightarrow \downarrow$	MP 1,3
5.	p	MP 2,4

e) $\vdash (p \rightarrow \downarrow) \rightarrow (p \rightarrow q) \Leftrightarrow \{p \rightarrow \downarrow\} \vdash p \rightarrow q \Leftrightarrow$
$\Leftrightarrow \Sigma := \{p, p \rightarrow \downarrow\} \vdash q$. Ez utóbbi bizonyítása:

1.	p	$\in \Sigma$
2.	$p \rightarrow \downarrow$	$\in \Sigma$
3.	$\downarrow$	MP 1,3
4.	$((q \rightarrow \downarrow) \rightarrow \downarrow) \rightarrow q$	AX3$[F/q]$
5.	$\downarrow \rightarrow ((q \rightarrow \downarrow) \rightarrow \downarrow)$	AX2$[F/\downarrow, G/q \rightarrow \downarrow]$
6.	$(q \rightarrow \downarrow) \rightarrow \downarrow$	MP 3,5
7.	q	MP 4,6

f) $\vdash ((F \rightarrow \downarrow) \rightarrow (G \rightarrow \downarrow)) \rightarrow (((F \rightarrow \downarrow) \rightarrow G) \rightarrow F)$
$\Leftrightarrow \{(F \rightarrow \downarrow) \rightarrow (G \rightarrow \downarrow)\} \vdash ((F \rightarrow \downarrow) \rightarrow G) \rightarrow F$
$\Leftrightarrow \Sigma := \{(F \rightarrow \downarrow) \rightarrow (G \rightarrow \downarrow), (F \rightarrow \downarrow) \rightarrow G\} \vdash F$ Ennek bizonyítása:

1.	$(F \rightarrow \downarrow) \rightarrow (G \rightarrow \downarrow)$	$\in \Sigma$
2.	$(F \rightarrow \downarrow) \rightarrow G$	$\in \Sigma$
3.	$[(F \rightarrow \downarrow) \rightarrow (G \rightarrow \downarrow)] \rightarrow [((F \rightarrow \downarrow) \rightarrow G) \rightarrow ((F \rightarrow \downarrow) \rightarrow \downarrow)]$ AX3$[F/F \rightarrow \downarrow, G/G, H/\downarrow]$	
4.	$((F \rightarrow \downarrow) \rightarrow G) \rightarrow ((F \rightarrow \downarrow) \rightarrow \downarrow)$	MP 1,3
5.	$(F \rightarrow \downarrow) \rightarrow \downarrow$	MP 2,4
6.	$((F \rightarrow \downarrow) \rightarrow \downarrow) \rightarrow F$	AX3$[F/F]$
7.	F	MP 5,6

Megjegyzés. Ha a bizonyított formulában az $F \rightarrow \downarrow$ helyére $\neg F$ és $G \rightarrow \downarrow$ helyére $\neg G$ helyettesítéseket elvégeznénk, akkor egy másik (azonosan hamis helyett negációt tartalmazó) Hilbert-féle kalkulus 3. axiómáját kapnánk (lásd Fülöp Zoltán: Gyakorló feladatok a "Logika a számítástudományban" tárgyhoz I.). A helyettesítéssel kapott formula:

$$(\neg F \rightarrow \neg G) \rightarrow ((\neg F \rightarrow G) \rightarrow F).$$

g) $\{(p \to\!\downarrow) \to (q \to\!\downarrow)\} \vdash q \to p \Leftrightarrow \Sigma := \{(p \to\!\downarrow) \to (q \to\!\downarrow), q\} \vdash p$ Ennek bizonyítása:

1.	$(p \to\!\downarrow) \to (q \to\!\downarrow)$	$\in \Sigma$
2.	q	$\in \Sigma$
3.	$[(p \to\!\downarrow) \to (q \to\!\downarrow)] \to [((p \to\!\downarrow) \to q) \to ((p \to\!\downarrow) \to\!\downarrow)]$ AX3$[F/p \to\!\downarrow, G/q, H/\downarrow]$	
4.	$((p \to\!\downarrow) \to q) \to ((p \to\!\downarrow) \to\!\downarrow)$	MP 1,3
5.	$q \to ((p \to\!\downarrow) \to q)$	AX2$[F/q, G/p \to\!\downarrow]$
6.	$(p \to\!\downarrow) \to q$	MP 2,5
7.	$(p \to\!\downarrow) \to\!\downarrow$	MP 4,6
8.	$((p \to\!\downarrow) \to\!\downarrow) \to p$	AX3$[F/p]$
9.	p	MP 7,8

h) $\Sigma := \{(p \to\!\downarrow) \to p\} \vdash p$ bizonyításához felhasználjuk azt, hogy tetszőleges F formulára $F \to F$-et már le tudjuk vezetni 5 lépésben, lásd a fejezet első feladatát.

1.	$(p \to\!\downarrow) \to p$	$\in \Sigma$
2.	$[(p \to\!\downarrow) \to (p \to\!\downarrow)] \to [((p \to\!\downarrow) \to p) \to ((p \to\!\downarrow) \to\!\downarrow)]$ AX3$[F/p \to\!\downarrow, G/p, H/\downarrow]$	
...	mint $p \to p$ levezetése (5 lépésben):	
7.	$(p \to\!\downarrow) \to (p \to\!\downarrow)$	
8.	$((p \to\!\downarrow) \to p) \to ((p \to\!\downarrow) \to\!\downarrow)$	MP 2,7
9.	$(p \to\!\downarrow) \to\!\downarrow$	MP 1,8
10.	$((p \to\!\downarrow) \to\!\downarrow) \to p$	AX3$[F/p]$
11.	p	MP 9,10

9. fejezet

Normálformák az elsőrendű kalkulusban

Elméleti összefoglaló

Ebben a fejezetben visszatérünk az ítéletkalkulus békés mezsgyéiről az elsőrendű logika csataterére. Mivel számos esetben van arra példa, hogy szakértői rendszerek axiómái nem ítéletkalkulusban, hanem elsőrendű logikában vagy annak egy szeletében vannak reprezentálva, itt már olyan feladatokkal szembesülünk, melyekre esetenként tényleges implementációt is kell fejlesztenünk. Ezt a kritériumot szem előtt tartva az elméleti összefoglalók ettől a fejezettől fogva kiegészülnek implementációs részletekkel, trükkökkel és buktatókkal is.
Ezidáig arra láttunk több algoritmikus módszert, hogy az *ítéletkalkulusban* igazoljuk egy Σ formulahalmaz kielégíthetetlenségét, vagy egy $\Sigma \models F$ következmény fennállását. Módszereink közül a következő fejezetekben a *rezolúciós algoritmust* ki fogjuk terjeszteni elsőrendű logikára is, célunk tehát egy olyan algoritmus-család fejlesztése, melynek specifikációja a következő:

- Input: elsőrendű formulák véges (de legalábbis rekurzíve felsorolható) Σ halmaza.
- Output: IGEN pontosan akkor, ha Σ kielégíthetetlen.

Amennyiben Σ kielégíthető, az algoritmusoknak megengedett (a NEM válasz adása mellett) a végtelen ciklusba esés (melyet ebben a gyűjteményben a ,,nem megállás" szinonimájaként kezelünk, bár a ,,végtelen ciklus" egyes értelmezések szerint a ,,nem megállás"-nak valódi alesete, mégpedig a program állapotainak ciklikus ismétlődése. Ily módon pl. az `i := 1; while( i > 0 ) i++;` program a szigorú értelemben nem számít végtelen ciklusnak. Ebből a szempontból kissé lazán használjuk majd ezt az elnevezést.)

A végtelen ciklus megengedése, ahogy a későbbi fejezetekben látni fogjuk, nem a követelmények mesterséges enyhítése: az elsőrendű logikában (amennyiben a formulák felépítéséhez használt nyelv ,,elég bonyolult", így pl. ha már szerepel benne két unáris függvényjel és egy bináris predikátumjel) *nincs* olyan algoritmus, mely egy input F formula kielégíthetetlenségét *eldöntené*, vagyis mely mindig megállna és helyes válasszal térne vissza.
Algoritmusaink tervezésekor igyekszünk szem előtt tartani, hogy az input Σ halmaz akár egy végtelen nagy halmaz is lehet, így egyszerre csak mindig a soron következő elemét fogjuk lekérni, és nem használjuk a Σ halmaz egészét egyszerre, mindig csak egy véges részhalmazát.
Legelőször is új normálformákat kell bevezetnünk, hiszen a ,,konjunktív normálforma" a kvantorok jelenléte miatt nem világos. Elsőrendű logikában a következő normálformákat ismerjük meg:

a) *Kiigazított alak*: az F formula kiigazított alakú, ha benne ,,nincs névütközés", vagyis ha benne

   - különböző pozíción levő kvantorok különböző változókat is kötnek és
   - nincs olyan változó, mely szabadon és kötötten is előfordul.

b) *Prenex alak*: az F formula prenex alakú, ha benne ,,elöl vannak a kvantorok", vagyis ha $Q_1x_1Q_2x_2\ldots Q_nx_nF^*$ alakú, ahol a Q_i-k kvantorok, F^* pedig kvantormentes. Prenex alakú formula legnagyobb kvantormentes részformuláját (vagyis itt F^*-ot) a formula *magjának* is nevezzük.

c) *Skolem* (ejtsd: szkólem) *alak*: az F formula Skolem alakú, ha prenex és benne minden kvantor univerzális, vagyis ha $\forall x_1\forall x_2\ldots\forall x_nF^*$ alakú, ahol F^* a kvantormentes mag.

d) *Zárt Skolem, CNF maggal*: Skolem alakú formula, melyben nincsenek szabad változó-előfordulások és melynek magja konjunktív normálformájú.

Célunk az lesz, hogy tetszőleges input formulát a legutolsó (zárt Skolem, CNF maggal) alakra hozzunk – lehetőleg gyorsan. Sajnos olyan algoritmus, mely konstruktívan egy *ekvivalens* zárt Skolem-alakra hozna egy formulát, nincs. Mivel a célunk csak annyi, hogy az input Σ halmaz kielégíthetetlenségét felismerjük, ezért elvárásunk a ,,normálformára hozással" kapcsolatban csak annyi lesz, hogy az output az inputtal *s-ekvivalens* legyen: kielégíthető inputból kielégíthető outputot, kielégíthetetlenből pedig kielégíthetetlent készítsen.
A normálformára hozó algoritmusok:

Lezárás, $\leftrightarrow$ és $\rightarrow$ eliminálása

Praktikus okokból érdemes még mindenféle normálformára hozást megelőzően lezárni a formulát és eliminálni belőle a nyilakat. Alkalmazzuk az $F \rightarrow G \equiv (\neg F) \vee G$ és $F \equiv G \equiv (\neg F \vee G) \wedge (\neg G \vee F)$ átírásokat, ezzel a formulában már csak a $\neg$, $\vee$ és $\wedge$ konnektívák maradnak.
Ezt követően a szabad változó-előfordulások mindegyikét *új* konstansjellel helyettesítsük. Ezzel kapcsolatban arra kell figyelni, hogy ha az input Σ formulahalmaz *több* formulájában is szerepel egy x változó szabadon, akkor azt az összes formulában ugyanarra az új c_x konstansjelre kell cseréljük.
Implementációs kérdések: változó-átnevezés

- Előre ismernünk kell, hogy a rekurzívan felsorolható input Σ formulahalmazban milyen konstansjelek szerepelnek.
- Vagy legalábbis szükségünk van egy olyan eljárásra, mely garantáltan olyan konstansjeleket generál tetszőleges számban, melyek az inputban nem fordulnak elő.
- Nyilván kell tartanunk egy `Map<Var,Constant>` struktúrában (egy változó→konstansjel leképezést tároló mapben), hogy eddig melyik szabad változót melyik konstansjelre cseréltük. Praktice helyettesítésnek is felfoghatjuk.
- Ezt a cserét (vagy helyettesítést) a Σ formulahalmaz inputon sorra érkező formuláján végrehajtjuk, majd ha a formulában maradt változó, minden ilyen x változóhoz generálunk egy új c_x konstans, eltesszük a mapbe, és végrehajtjuk a cserét.
- Előfordulhat, hogy nem ismerjük előre a használt konstansjelek körét (pl. mert tetszőleges, kisbetűs string lehet függvény- vagy predikátumjel, mint például a `Prolog` nyelv esetében) és nincs is olyan módszerünk, mellyel ezeket elkerülő konstansjeleket tudnánk generáltatni. Ekkor ütközés-detektáláskor (egy korábban ,,generált" konstansjel ütközik egy, Σ soron következő formulájában ,,alanyi jogon" szereplő konstansjellel) vagy visszamenőleg cseréljük a generált konstanjelet másra minden korábbi formulában, vagy a későbbiekben az ,,alanyi jogon" használt konstansjelet cseréljük mindig egy másik generáltra. Az utóbbi módszer ugyan gyorsabb, de nem tartja az ekvivalenciát, a kielégíthetőséget viszont igen.

Implementációs kérdések: $\rightarrow$, $\leftrightarrow$ eliminálás

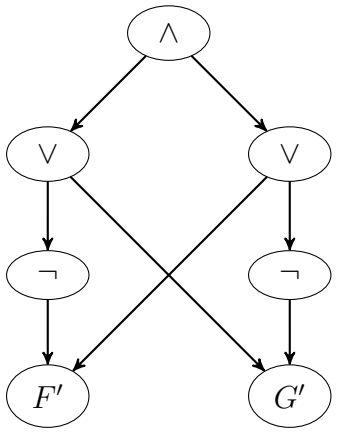
- Praktikus dolog a formulákat nem feltétlenül formulaFA adatszerkezetben tárolni. Ekkor ugyanis egymásba ágyazott $\leftrightarrow$ konnektívák esetén exponenciális méretűvé hízhat az eredeti formulánk, például:

$$\begin{aligned} p_1 \leftrightarrow p_2 &\equiv (\neg p_1 \vee p_2) \wedge (p_1 \vee \neg p_2) \\ (p_1 \leftrightarrow p_2) \leftrightarrow p_3 &\equiv \big(\neg\big((\neg p_1 \vee p_2) \wedge (p_1 \vee \neg p_2)\big) \vee p_3\big) \wedge \\ &\quad \big(\big((\neg p_1 \vee p_2) \wedge (p_1 \vee \neg p_2)\big) \vee \neg p_3\big) \\ &\dots \end{aligned}$$

és így tovább, minden szinten kétszer olyan hosszú a formula, mint az előzőn: $\Omega(2^n)$ hosszú, ha n változót használunk. Ezt a helyzetet *részlegesen* kezelhetjük formulaDAG (directed acyclic graph, irányított körmentes gráf) reprezentációval, avagy *hálózat* reprezentációval, ahol minden `Node` csúcsnak van egy típusa és esetlegesen gyerekei, melyek `Node` referenciák:

- `Atom` típusú csúcsnak nincs gyereke, egy atomi formulát tárol;
- `Negation` típusú csúcsnak egy gyereke lehet;
- `Disjunction` és `Conjunction` csúcsoknak az implementációs hatékonyság, valamint a műveletek kommutativitása, asszociativitása és idempotenciája figyelembevételével gyerekei egy tetszőleges elemszámú `Set<Node>` (halmaz);
- `Exists`, `Forall` csúcsoknak egy változóra mutató mezőjük (ezt a változót köti a kvantor) és egy gyerekük lehet.

Természetesen OOP implementációs nyelv használatakor a ,,típus"t absztrakt osztályból leszármaztatással vagy interface implementálással illik a paradigmát követve készíteni, más esetben structtal vagy extrém esetben unionnal. A kvantorok ill. többoperandusú műveletek számára is külön absztrakt ősosztály / interface készítése javasolt.
Ekkor egy $F \leftrightarrow G$ alakú formula reprezentációja a nyíl eliminálása után:



ahol F' és G' az F ill. G részformulákból készített hálózatok gyökércsúcsai. A transzformáció lineáris idejű és mint ilyen, ekkora outputot is készít.
A módszer nem mindenható, mint az összefoglaló végén szereplő CNF-re hozással kapcsolatos fejtegetésből is kiderül, de a hálózatok, mint láthatjuk, a formulák lényegesen tömörebb reprezentációját teszik lehetővé a fa (vagy string) tárolásnál.

Kiigazítás

A kiigazítás egyszerű: a változónév-ütközéseket kell csak feloldjuk. Ezen a ponton már nincs a formulában szabad változó, így csak arra kell figyelnünk, hogy különböző pozíción levő kvantorok különböző változót kössenek le. Ezt papíron megtehetjük például indexeléssel, implementáláskor pedig egy változó-factoryval, mely mindig garantáltan teljesen új változóneveket hoz nekünk létre.

Prenex alakra hozás

Ha a formulánk kiigazított, és nem tartalmaz nyilakat, akkor (!) a következő ekvivalenciák fennállnak, ezeket balról jobbra elvégezve, míg csak lehet, egy prenex alakú formulát kapunk:

$$\begin{aligned}
\neg\forall x F &\equiv \exists x \neg F \\
\neg\exists x F &\equiv \forall x \neg F \\
\exists x F \vee G &\equiv \exists x (F \vee G) \\
\exists x F \wedge G &\equiv \exists x (F \wedge G) \\
\forall x F \vee G &\equiv \forall x (F \vee G) \\
\forall x F \wedge G &\equiv \forall x (F \wedge G) \\
F \vee \exists x G &\equiv \exists x (F \vee G) \\
F \wedge \exists x G &\equiv \exists x (F \wedge G) \\
F \vee \forall x G &\equiv \forall x (F \vee G) \\
F \wedge \forall x G &\equiv \forall x (F \wedge G)
\end{aligned}$$

Papíron, kis formulákra talán ez a leggyorsabb módszer, melyet érdemes alulról felfelé haladva végrehajtani, a kis részformuláknál elkezdve.
Implementációs kérdések: Prenex alak
Vegyük észre a következőt: a változót kötő kvantor végeredményben pontosan akkor ,,fordul", ha *páratlan sok negáció* hatáskörében szerepel. Így egy gyorsabb algoritmus:

a) A változókat pontosan ugyanabban a sorrendben deklaráljuk a formula elején, ahogy azok a formulában eredetileg (balról jobbra olvasva) szerepeltek.

b) Az x változót kötő kvantor pontosan akkor fordul, ha eredetileg páratlan sok negáció hatáskörében szerepelt.

c) A formula magját úgy kapjuk, hogy az eredeti formulából töröljük a kvantorokat.

Amennyiben a formulát fa-struktúrában reprezentáljuk, a következő módon gyűjthetjük ki a kvantorokat egy verembe lineáris időben:

```
function BEJAR(Node root, boolean flip, Stack stack )
    if root.type == NEGATION then
        BEJAR(root.child, !flip, generatedQuantifiers)
    else if root.type ∈ {∀, ∃} then
        STACK.PUSH(flip⊕(root.type== ∃)?∃root.var:∀root.var);
        BEJAR(root.child,flip,generatedQuantifiers)
    else
        FOREACH node in root.children
        BEJAR(node,flip,stack)
```

Szülőre mutató referenciával egyrészt ezt lehet rekurzió-mentesíteni, másrészt akkor a kvantorok törlését (ha jogunkban áll elrontani az input formula adatszerkezetet) is meg tudjuk oldani a bejárás közbeni átláncolással.

Skolem-alak

Amennyiben a formula már prenex alakban van, elvégezzük rá a következő átalakítást: minden egyes egzisztenciálisan deklarált (tehát $\exists x$ kvantorral kötött) változó esetében

- töröljük a formula elején a kvantorok sorából a $\exists x$-et;
- a formula magjában x minden előfordulása helyett egy $f(z_1, \ldots, z_k)$ alakú termet írunk, ahol:
  - $z_1, \ldots, z_k$ az x előtt, univerzálisan deklarált változók;
  - f pedig egy *új* függvényszimbólum.

Amennyiben több formula is szerepel az inputban, minden egyes ilyen esetben teljesen új függvényjelet kell bevezetnünk!
Megjegyzés: ha tehát a $\exists x$ előtt egyáltalán nem szerepel univerzális kvantor, úgy új *konstans*jelet vezetünk be.
Implementációs kérdések: Skolem-alak
A lezárásnál a változók helyére új konstansok bevezetésekor elmondottak itt is érvényesek, a Skolem-függvényekkel kapcsolatban.

A mag CNF-re hozása

A formula magját végül CNF-re hozzuk a szokásos módon: először a negációkat a deMorgan azonosságok alkalmazásával ill. a dupla negáció eliminálásával az atomi formulák mellé visszük, majd a disztributivitás alkalmazásával a diszjunkciókat a konjunkciók alá kényszerítjük. Itt tehát ami szerepet korábban az ,,ítéletváltozók" játszottak, most az atomi formulák fognak.
Implementációs kérdések: CNF Ha a formulánkban már csak $\vee$ és $\wedge$ szerepel, akkor sem tudjuk az ekvivalenciát és a polinom méretet *egyidőben* biztosítani (ahogy az ítéletkalkulus normálformás fejezetének ide kapcsolódó feladata is mutatta). Eljárhatunk pl. úgy, hogy a mag minden F részformulához bevezetünk egy p_F új predikátumváltozót és a következő átírásokat alkalmazzuk:

$$\begin{aligned} F = p(t_1, \ldots, t_n) &\mapsto (\neg p_F \vee p(t_1, \ldots, t_n)) \wedge (p_F \vee \neg p(t_1, \ldots, t_n)) \\ F = G \vee H &\mapsto (\neg p_F \vee p_G \vee p_H) \wedge (\neg p_G \vee p_F) \wedge (\neg p_H \vee p_F) \\ F = G \wedge H &\mapsto (\neg p_F \vee p_G) \wedge (\neg p_F \vee p_H) \wedge (p_F \vee \neg p_G \vee \neg p_H) \end{aligned}$$

Ezen átírások konjunkciója, és még p_F, ha eredetileg F a formulánk magja, egy lineáris időben elkészíthető, lineáris méretű formula, mely az inputtal s-ekvivalens. A konstrukció akkor is működik, ha az input nem formula, hanem hálózat. A kimenet ekkor is egy kisméretű formula lesz.

Megjegyzések

Amint a lezárást elvégezzük, a kapott formula nem lesz ekvivalens az eredetivel, így első lépésben lezárva máris elveszítjük az ekvivalenciát. Amennyiben ez zavaró, a lezárást természetesen az egész algoritmus-lánc végén is elvégezhetjük, ekkor azonban a kiigazításnál a szabad előfordulásokra is oda kell figyelni.
Mivel azonban a Skolem-függvények bevezetése mindenképp szükséges lépés és elrontja az ekvivalenciát, talán nem érdemes ,,minél tovább ekvivalens alakban" tartani a formuláinkat.

Feladatok

9.1. Feladat. Igazítsuk ki az alábbi formulákat:

a) $\forall x \exists y \big(p(x,y) \to \exists z p(z,y)\big) \ \vee \ \exists x p(x,y)$

b) $\big(\forall x \exists y p(x,y)\big) \leftrightarrow \exists x p(x,y)$

9.1. Feladat megoldása.

a) Átnevezzük az összes kötött változót mondjuk indexeléssel:

$$\forall x_1 \exists y_2 \big(p(x_1, y_2) \to \exists z_3 p(z_3, y_2)\big) \ \vee \ \exists x_4 p(x_4, y).$$

Amennyiben ezt megelőzően eliminálják a $\to$ nyilat és a szabad y helyébe egy új c_y konstanst is bevezetünk, úgy:

$$\forall x_1 \exists y_2 \big(\neg p(x_1, y_2) \vee \exists z_3 p(z_3, y_2)\big) \ \vee \ \exists x_4 p(x_4, c_y).$$

b) Mindenképp eliminálnunk kell a $\leftrightarrow$ jelet. Bevezetünk a szabad y helyébe is egy új c_y konstansjelet:

$$\big((\neg\forall x \exists y p(x, y)) \vee \exists x p(x, y_c)\big) \wedge \big(\forall x \exists y p(x, y) \vee \neg\exists x p(x, y_c)\big).$$

Ezután átindexeljük a kötött változókat:

$$\big((\neg\forall x_1 \exists y_2 p(x_1, y_2)) \vee \exists x_3 p(x_3, y_c)\big) \wedge \big(\forall x_4 \exists y_5 p(x_4, y_5) \vee \neg\exists x_6 p(x_6, y_c)\big).$$

9.2. Feladat. Hozzuk prenex alakra az alábbi kiigazított és $\to$, $\leftrightarrow$-mentesített formulákat:

a) $\neg\Big((\forall x\big(\neg\forall y(p(x, y) \vee \exists z\, q(x, z))\big) \ \wedge \ \exists w p(x, w)\big) \vee \neg\exists v q(v, v)\Big)$

9.2. Feladat megoldása.

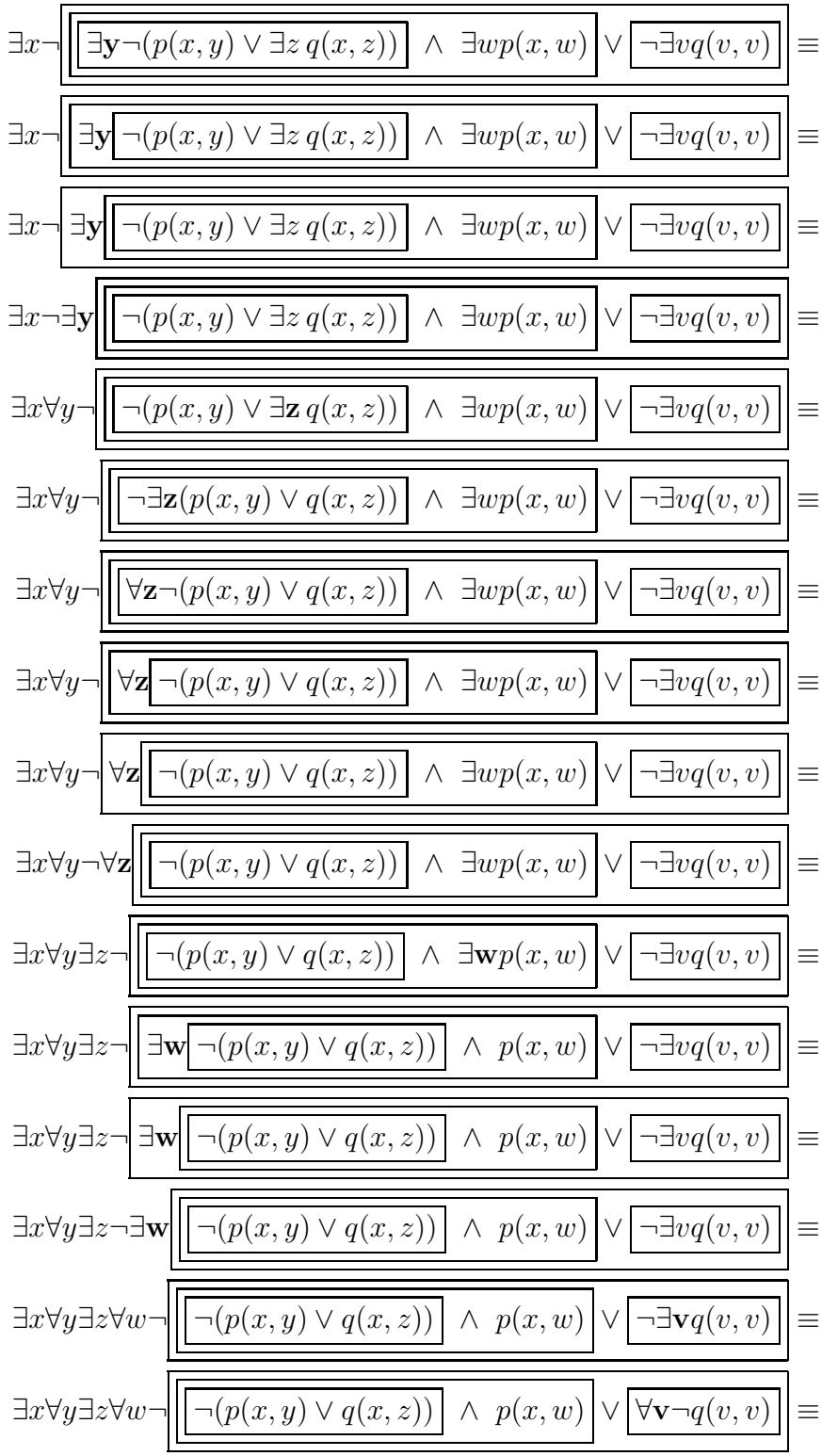
a) Lépésenként átírogatva, mindig egy szinttel kijjebb hozva az első, nem kívül lévő kvantort és helyenként a jobb olvashatóság kedvéért $\boxed{\textit{dobozokkal}}$ jelölve a szintaktikai alegységeket:

$$\neg\Big[\boxed{\forall \mathbf{x}\boxed{\neg\forall y(p(x, y) \vee \exists z\, q(x, z))}\ \wedge\ \exists w p(x, w)} \vee \boxed{\neg\exists v q(v, v)}\Big] \equiv$$

$$\neg\Big[\forall \mathbf{x}\Big[\boxed{\neg\forall y(p(x, y) \vee \exists z\, q(x, z))}\ \wedge\ \exists w p(x, w)\Big] \vee \boxed{\neg\exists v q(v, v)}\Big] \equiv$$

$$\neg\forall \mathbf{x}\Big[\boxed{\neg\forall y(p(x, y) \vee \exists z\, q(x, z))}\ \wedge\ \exists w p(x, w)\Big] \vee \boxed{\neg\exists v q(v, v)} \equiv$$

$$\exists x\neg\Big[\boxed{\neg\forall \mathbf{y}(p(x, y) \vee \exists z\, q(x, z))}\ \wedge\ \exists w p(x, w)\Big] \vee \boxed{\neg\exists v q(v, v)} \equiv$$


$\exists x \neg [\exists \mathbf{y} \neg (p(x,y) \vee \exists z\, q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \neg [\exists \mathbf{y} [\neg (p(x,y) \vee \exists z\, q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \neg [\exists \mathbf{y} [\neg (p(x,y) \vee \exists z\, q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \neg \exists \mathbf{y} [\neg (p(x,y) \vee \exists z\, q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \neg [\neg (p(x,y) \vee \exists \mathbf{z}\, q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \neg [\neg \exists \mathbf{z} (p(x,y) \vee q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \neg [\forall \mathbf{z} \neg (p(x,y) \vee q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \neg [\forall \mathbf{z} [\neg (p(x,y) \vee q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \neg \forall \mathbf{z} [\neg (p(x,y) \vee q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \neg \forall \mathbf{z} [\neg (p(x,y) \vee q(x,z)) \wedge \exists w p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \exists z \neg [\neg (p(x,y) \vee q(x,z)) \wedge \exists \mathbf{w} p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \exists z \neg [\exists \mathbf{w} [\neg (p(x,y) \vee q(x,z)) \wedge p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \exists z \neg [\exists \mathbf{w} [\neg (p(x,y) \vee q(x,z)) \wedge p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \exists z \neg \exists \mathbf{w} [\neg (p(x,y) \vee q(x,z)) \wedge p(x,w)] \vee [\neg \exists v q(v,v)] \equiv$
$\exists x \forall y \exists z \forall w \neg [\neg (p(x,y) \vee q(x,z)) \wedge p(x,w)] \vee [\neg \exists \mathbf{v} q(v,v)] \equiv$
$\exists x \forall y \exists z \forall w \neg [\neg (p(x,y) \vee q(x,z)) \wedge p(x,w)] \vee [\forall \mathbf{v} \neg q(v,v)] \equiv$

$$\exists x\forall y\exists z\forall w\neg\forall \mathbf{v}\boxed{\boxed{\neg(p(x,y)\vee q(x,z))}\ \wedge\ p(x,w)}\vee\boxed{\neg q(v,v)}\equiv$$

$$\exists x\forall y\exists z\forall w\exists v\neg\boxed{\boxed{\neg(p(x,y)\vee q(x,z))}\ \wedge\ p(x,w)}\vee\boxed{\neg q(v,v)}\equiv$$

$$\exists x\forall y\exists z\forall w\exists v\neg(((\neg(p(x,y)\vee q(x,z)))\ \wedge\ p(x,w))\vee(\neg q(v,v))).$$

Érthető, hogy már ekkora input esetén is van igény gyorsabb algoritmusra. Ha ismét az eredeti formulát vesszük, de ezúttal a negációk hatáskörét jelöljük dobozokkal:

$$\neg\boxed{\forall x(\neg\boxed{\forall y(p(x,y)\vee\exists z\,q(x,z))}\ \wedge\ \exists w p(x,w))\vee\neg\boxed{\exists v q(v,v)}}.$$

Ekkor x egy, y és z kettő-kettő, w egy, v kettő negáció-hatáskörbe („dobozba") esik, ezért a páros-páratlan szabály alapján x és w kvantora fordul, a többié nem. Az eredeti sorrendben deklarálva a változókat és leírva a kvantorok törlésével előálló formulát magként:

$$\exists x\forall y\exists z\forall w\exists v\neg\boxed{(\neg\boxed{(p(x,y)\vee q(x,z))}\ \wedge\ p(x,w))\vee\neg\boxed{q(v,v)}}.$$

Valóban, ez a módszer mintha gyorsabb lenne.

9.3. Feladat. Hozzuk Skolem alakra az előző feladat formuláit, majd magjukat CNF-re.

9.3. Feladat megoldása.

a) Tehát a prenex alak:

$$\exists x\forall y\exists z\forall w\exists v\neg(((\neg(p(x,y)\vee q(x,z)))\ \wedge\ p(x,w))\vee(\neg q(v,v))).$$

Helyettesítenünk kell a magban az egzisztenciálisan deklarált változókat, vagyis...

- az x változót egy új *konstansjellel*, mert nem deklaráltunk előtte univerzálisan senkit – legyen ez mondjuk c;
- a z változót mondjuk $f(y)$-nal, mert f-et még nem használtunk sehol és z előtt y van univerzálisan deklarálva;
- a v változót mondjuk $g(y,w)$-vel, mert g-t még nem használtunk sehol és v előtt y és w vannak univerzálisan deklarálva.

Töröljük az egzisztenciális deklarációkat. Amit kapunk:

$$\forall y \forall w \neg(((\neg(p(c,y) \vee q(c,f(y)))) \ \wedge \ p(c,w)) \vee (\neg q(g(y,w),g(y,w)))).$$

A magot hozzuk konjunktív normálformára. Átírogatós módszerrel bevisszük a negációkat, mindig az első olyan negációt tolva lejjebb, mely nem atomi formula mellett áll:

$$\begin{aligned}
\neg(((\neg(p(c,y) \vee q(c,f(y)))) \wedge p(c,w)) \vee (\neg q(g(y,w),g(y,w)))) \equiv \\
(\neg((\neg(p(c,y) \vee q(c,f(y)))) \wedge p(c,w)) \wedge (\neg\neg q(g(y,w),g(y,w)))) \equiv \\
(((\neg\neg(p(c,y) \vee q(c,f(y)))) \vee \neg p(c,w)) \wedge (\neg\neg q(g(y,w),g(y,w)))) \equiv \\
((((p(c,y) \vee q(c,f(y)))) \vee \neg p(c,w)) \wedge (\neg\neg q(g(y,w),g(y,w)))) \equiv \\
(((p(c,y) \vee q(c,f(y))) \vee \neg p(c,w)) \wedge q(g(y,w),g(y,w))).
\end{aligned}$$

Itt most álljunk meg egy kicsit: ha a formulában csak $\wedge$, $\vee$ és $\neg$ szerepel és a negációkat akarjuk bevinni az atomi formulák mellé (ún. NNF-et (Negation Normal Form-ot) akarunk létrehozni), akkor szintén működik a ,,dobozoljuk a negációkat" módszer:

$$\neg\boxed{(\neg\boxed{(p(c,y) \vee q(c,f(y)))} \ \wedge \ p(c,w)) \vee \neg\boxed{q(g(y,w),g(y,w))}}.$$

A páratlan sok negáció hatáskörben lévő $\vee$ $\wedge$-re és viszont fordul, továbbá negatív literálból pozitív lesz és fordítva. Más nem változik, az ,,eredeti" negációk eltűnnek:

$$\boxed{(\boxed{(p(c,y) \vee q(c,f(y)))} \ \wedge \ p(c,w)) \vee \boxed{q(g(y,w),g(y,w))}},$$

amiből csak a középen lévő $p(c,w)$ és a mellette levő két konnektíva van páratlan sok dobozban, ezek fordulnak, tehát:

$$\boxed{(\boxed{(p(c,y) \vee q(c,f(y)))} \ \vee \neg p(c,w)) \wedge \boxed{q(g(y,w),g(y,w))}},$$

dobozokra már nincs szükség, helyettük zárójelek:

$$((((p(c,y) \vee q(c,f(y)))) \ \vee \neg p(c,w)) \wedge (q(g(y,w),g(y,w)))),$$

kész is. Az NNF készítése könnyű. Továbbá, ez a mag már CNF-ben is van, a klózok:

$$\big\{\{p(c,y), q(c,f(y)), \neg p(c,w)\}, \{q(g(y,w),g(y,w)\}\big\}.$$

A formula egyébként kielégíthető, mert pl. a q predikátumot konstans igazra véve kielégítünk minden klózt.

9.4. Feladat. Igazoljuk, hogy tetszőleges F formulához létezik vele s-ekvivalens *változómentes* formula.

9.4. Feladat megoldása. A kielégíthető formulák $\uparrow$-val, a kielégíthetetlenek $\downarrow$-val s-ekvivalensek.

10. fejezet

Az alap rezolúciós algoritmus

Elméleti összefoglaló

A rezolúciós algoritmus *első* kiterjesztése az *alap rezolúciós algoritmus*. Itt már feltesszük, hogy az input olyan zárt Skolem alakú formulák Σ (rekurzívan felsorolható) halmaza, melyek magja CNF-ben van. Ekkor a formulákban szereplő összes változóról tudjuk, hogy a vonatkozó formulákban univerzálisan vannak kötve, így az inputban a kvantorokat már nem jelöljük; a diszjunkció ill. konjunkció kommutativitása, asszociativitása és idempotenciája miatt pedig egy-egy klózt megint literálok *halmazaként*, egy CNF-et pedig klózok halmazaként reprezentálunk. Mivel $\forall x(F \wedge G) \equiv \forall xF \wedge \forall xG$, így azt, hogy melyik klóz melyik formulából „származik", nem kell nyilvántartanunk: az input Σ-ban szereplő összes formula klózait egyetlen Σ' klózhalmazba (tehát a klózok halmaza lehet végtelen is; azonban egy klóz mindig csak véges sok literált tartalmaz!) gyűjthetjük.
Vagyis az alap rezolúciós algoritmus inputja elsőrendű logikai klózok egy Σ' halmaza, egy-egy klóz pedig véges sok literált tartalmaz, egy literál pedig egy atomi formula vagy annak negáltja.
Az output pedig, hogy az ezek univerzális lezártjaként előálló formulahalmaz kielégíthetetlen-e? (A kérdésfelvetés módjához: azt fogjuk tudni megmondani teljes biztonsággal, ha kielégíthetetlen. Kielégíthető input esetében eshetünk végtelen ciklusba is.)
Emlékezzünk vissza: *alaptermn*ek a változómentes termeket neveztük, mint pl. c, $f(c)$, $g(c,c)$, $g(f(c),g(c,f(c)))$ stb. Az alaptermek halmazát T_0 jelöli. Általános esetben T_0 lehet üres is (ha az alkalmazott elsőrendű nyelvben nincs konstans), mivel viszont az alap rezolúciós algoritmus úgy épül fel, hogy a változók helyébe alaptermeket helyettesítünk, ezért ha nincs konstansjel, akkor készítünk egyet (pl c-t) és így már T_0 nem lesz üres. Ha pedig T_0 nemüres, úgy pontosan akkor végtelen, ha a nyelvben van nemkonstans függvényjel is.

A változómentes literálokat *alapliterálnak*, a változómentes atomi formulákat pedig *alap atomi formuláknak* nevezzük. Egy alap atomi formula (mint pl. a $p(f(c), c)$) tetszőleges struktúrában egy igazságértékre (bitre) értékelődik ki (ne feledjük, változóink most le nem írt univerzális kvantorok hatáskörében lesznek!), így az alap atomi formulákat tekinthetjük mint *ítéletváltozókat*. Egy klóz pedig *alapklóz*, ha benne csak alapliterálok szerepelnek.
Az alap rezolúciós algoritmus a következőképp épül fel:

- Listát vezetünk alapklózokról.
- Egy klózt kétféle okból vehetünk fel a listára:
  - ha Σ'-beli klóznak egy alap példánya, VAGY
  - két, a listán már szereplő klóznak rezolvense (ítéletkalkulusi értelemben).

Egy ,,klóznak alap példánya" annyit tesz, hogy a klózban minden változót egy (tetszőleges) alaptermmel helyettesítünk.
Látható, hogy az algoritmus elágazási faktora, így a keresési tér is igen nagy (ha T_0 végtelen és legalább egy klózban szerepel változó, akkor végtelen sokféle felvehető alappéldány létezik), így a kérdés, hogy ,,levezethető-e az üres klóz", egyáltalán nem egyszerű. A kérdés pedig érdekes, hiszen igaz az alaprezolúció helyességi és teljességi tétele, miszerint

,,Σ' pontosan akkor kielégíthetetlen, ha belőle levezethető alaprezolúcióval az üres klóz."

A ,,nem egyszerű" ítéletkalkulusban ,,csak" annyit tett, hogy NP-teljes volt a kérdés, így exponenciális időigényű algoritmusunk volt rá; itt annyit tesz, hogy csak *félig eldönthető* – azaz ha kijöhet az üres klóz, akkor arra ,,előbb-utóbb" rájöhetünk, de ha nem, akkor a végtelen ciklusba esések kívül nincs más opciónk, ha csak helyesen akarunk válaszolni.
A fentiek közül a *helyettesítés* igényelhet matematikai alapozást: ha u és t termek, x pedig változó, akkor az $u[x/t]$ termet rekurzívan u felépítése szerint a következő módon definiáljuk:

$$u[x/t] = \begin{cases} t & \text{, ha } u = x; \\ y & \text{, ha } u = y \in Var - \{x\}; \\ f(u_1[x/t], \ldots, u_n[x/t]) & \text{, ha } u = f(u_1, \ldots, u_n). \end{cases}$$

Ha pedig $p(u_1, \ldots, u_n)$ egy atomi formula (azaz pozitív literál), akkor $p(u_1, \ldots, u_n)[x/t] = p(u_1[x/t], \ldots, u_n[x/t])$ és ha $\ell = \neg F$ negatív literál, akkor $\ell[x/t] = \neg(F[x/t])$.

Ez tényleg azt jelenti, hogy $F[x/t]$-t úgy kapjuk, hogy benne x helyébe mindenhol t-t írunk, míg F egy term vagy egy literál. (Ha F kvantorokat is tartalmazó formula lenne, bonyolódna a helyzet, de a példatárban ilyen eset sehol nem lesz.) Amennyiben C egy klóz, akkor $C[x/t] = \{\ell[x/t] : \ell \in C\}$ az a klóz, melyet úgy kapunk, hogy C-ben minden literálon elvégezzük az $[x/t]$ helyettesítést. Helyettesítés után persze lehet újabb helyettesítést végezni, a köztes szögletes zárójeleket elhagyva $C[x_1/t_1][x_2/t_2]\dots[x_n/t_n]$ helyett csak $C[x_1/t_1, x_2/t_2, \dots, x_n/t_n]$-t írunk. Egy C klóz alappéldányai tehát $C[x_1/t_1, \dots, x_n/t_n]$ alakúak, ahol $\{x_1, \dots, x_n\}$ a C-ben szereplő összes változó halmaza, a t_i-k meg mind alaptermek. A Σ' klózhalmaz alappéldányainak halmazát egyébként $E(\Sigma')$ jelöli, az EXTENSION szóból: ezt a halmazt a Σ' *Herbrand-kiterjesztéseként* ismerjük.

Feladatok

10.1. Feladat. Igazoljuk alap rezolúcióval, hogy az alábbi formulák ill. formulahalmazok kielégíthetetlenek! Ahol kell, hozzunk előbb zárt Skolem alakra, CNF maggal.

a) $\exists x \forall y (p(x,y) \leftrightarrow \neg p(y,y))$

b) $\forall x (p(x) \wedge \neg p(f(x))$

c) $\exists x p(x) \ \wedge \ \forall x (p(x) \to q(x)) \ \wedge \ \forall x (\neg q(x))$

d) $\forall x p(x) \ \wedge \ \forall x (p(f(x)) \to \neg q(x)) \ \wedge \ \exists x (q(x))$

e) $\big(\exists x p(x) \ \vee \exists x q(x)\big) \wedge \forall x (p(x) \to q(x)) \ \wedge \ \forall x (\neg q(x))$

f)

$$\begin{aligned}\forall x \forall y \forall z \big(& p(x, f(y) \\ & \wedge (p(f(x), z) \to q(x, g(z)) \\ & \wedge (\neg q(f(y), g(y)) \vee \neg p(y,y))\big)\end{aligned}$$

g)

$$\begin{aligned}\forall x \forall y \forall z \big(& p(f(x), y) \\ & \wedge (p(x, g(z)) \to q(g(x), f(z))) \\ & \wedge (\neg q(y, x) \vee \neg p(x, g(y)))\big)\end{aligned}$$

h)

$$\forall z \exists y \forall x \big((\neg p(z,a) \vee \neg p(z,x) \vee \neg p(x,z)) \\ \wedge ((p(z,y) \wedge p(y,z)) \vee p(z,a)) \big)$$

10.1. Feladat megoldása.

a) $\exists x \forall y (p(x,y) \leftrightarrow \neg p(y,y))$. A formula prenex alakú. Az egzisztenciálisan deklarált x helyébe c-t bevezetve:

$$\forall y (p(c,y) \leftrightarrow \neg p(y,y).$$

Az $\leftrightarrow$ jel kiküszöbölése:

$$\forall y \Big(\big(\neg p(c,y) \vee \neg p(y,y) \big) \ \wedge \ \big(p(c,y) \vee p(y,y) \big) \Big).$$

A klózok halmaza:

$$\Sigma' = \Big\{ \{\neg p(c,y), \neg p(y,y)\}, \ \{p(c,y), p(y,y)\} \Big\}.$$

Az alaptermek (nem túl nagy) halmaza: $T_0 = \{c\}$.

Az alap rezolúció során megengedett lépések tehát: kiválasztani a két klóz közül egyet és abban a változók (tehát y) helyébe tetszőleges alaptermet (tehát c-t) helyettesíteni:

a) $\{\neg p(c,c)\}$ – az első klóz y/c melletti alappéldánya. Vegyük észre, mindkét literálból ugyanaz az alappéldány készült, de mivel halmazról van szó, csak egyszer tesszük bele az alapliterált.

b) $\{p(c,c)\}$ – a másodiké

c) $\square$ – Res$(1,2)$, a $p(c,c)$ „ítéletváltozó" mentén.

Kész is.

b) $\forall x (p(x) \wedge \neg p(f(x))$. A formula zárt Skolem alakú, magja CNF-ben van. Klózok:

$$\Big\{ \{p(x)\}, \ \{\neg p(f(x))\} \Big\}.$$

Mivel nincs konstansjel, veszünk egyet, legyen ez c.

Akkor az alaptermek: $T_0 = \{c, f(c), f(f(c)), f(f(f(c))), \ldots\}$. Ezeket helyettesíthetjük a klózok változóinak (értsd: x) helyébe. Így tehát:

a) $\{p(c)\}$ – első klóz, x/c

b) $\{\neg p(f(c))\}$ – második klóz, x/c. Ezzel még nem lehet rezolválni, próbáljunk mást.

c) $\{p(f(c))\}$ – első klóz, x/c. Alakul:

d) $\square$ – Res$(2,3)$, $p(f(c))$ mentén.

Kész is, kielégíthetetlen. Az első lépést megspórolhattuk volna.

c) $\exists x p(x) \wedge \forall x(p(x) \to q(x)) \wedge \forall x(\neg q(x))$ Skolem alakra hozás után a klózok:

$$\{p(c)\},\ \{\neg p(x), q(x)\},\ \{\neg q(x)\}.$$

Az alaptermek halmaza $T_0 = \{c\}$, nincs nemkonstans függvényjel, így biztosan megáll az algoritmus:

a) $\{p(c)\}$ – első klóz

b) $\{\neg p(c), q(c)\}$ – második klóz $[x/c]$ alappéldánya

c) $\{q(c)\}$ – Res$(1,2)$

d) $\{\neg q(c)\}$ – harmadik klóz $[x/c]$ alappéldánya

e) $\square$ – Res$(3,4)$. Tehát a formula valóban kielégíthetetlen.

d) $\forall x p(x) \ \wedge \ \forall x(p(f(x)) \to \neg q(x)) \ \wedge \ \exists x(q(x))$

A klózok:

$$\{p(x)\}, \{\neg p(f(x)), \neg q(x)\}, \{q(c)\}.$$

(Itt most először a $\exists x$-et hoztuk ki előre, így lett c. Ugyanakkor, c helyett itt állhatna a skolemizálás után pl. $g(x,y)$ is.)

Alaptermek: $T_0 = \{c, f(c), f(f(c)), \ldots\}$

a) $\{q(c)\}$ – harmadik klóz

b) $\{\neg p(f(c)), \neg q(c)\}$ – második klóz $[x/c]$ példánya

c) $\{\neg p(f(c))\}$ – Res$(1,2)$

d) $\{p(f(c))\}$ – első klóz $[x/f(c)]$ példánya

e) $\square$ – Res$(3,4)$

Tehát a formula valóban kielégíthetetlen.

e) $\big(\exists x p(x) \ \vee \exists x q(x)\big) \wedge \forall x(p(x) \to q(x)) \ \wedge \ \forall x(\neg q(x))$

A klózok:

$$\{p(c), q(d)\}, \{\neg p(x), q(x)\}, \{\neg q(x)\}$$

Alaptermek: $T_0 = \{c, d\}$, véges, az algoritmus meg fog állni

a) $\{p(c), q(d)\}$ – első klóz

b) $\{\neg q(d)\}$ – harmadik klóz $[x/d]$

c) $\{p(c)\}$ – Res$(1, 2)$

d) $\{\neg p(c), q(c)\}$ – második klóz, $[x/c]$

e) $\{q(c)\}$ – Res$(3, 4)$

f) $\{\neg q(c)\}$ – harmadik klóz $[x/c]$

g) $\square$ – Res$(5, 6)$

Tehát a formula valóban kielégíthetetlen.

f)

$$\begin{aligned}\forall x \forall y \forall z \big(& p(x, f(y) \\ & \wedge (p(f(x), z) \rightarrow q(x, g(z)) \\ & \wedge (\neg q(f(y), g(y)) \vee \neg p(y, y))\big)\end{aligned}$$

Klózok:

$$\{p(x, f(y))\},\ \{\neg p(f(x), z), q(x, g(z))\},\ \{\neg q(f(y), g(y)), \neg p(y, y)\}$$

Alaptermek: $T_0 = \{c, f(c), f(f(c)), g(c), f(g(c)), \ldots\}$

a) $\{p(c, f(c))\}$ – első klóz $[x/c][y/c]$.
Itt rájövünk, hogy se $\neg p(f(x), z)$-ből, se $\neg p(y, y)$-ból nem lesz $\neg p(c, f(c))$, másképp kezdjük mégis.

b) $\{p(f(c), f(c))\}$ – első klóz $[x/f(c)][y/c]$

c) $\{\neg p(f(c), f(c), q(c, g(f(c)))\}$ – második klóz, $[x/c][z/f(c)]$, hogy rezolválhassunk p-nél

d) $\{q(c, g(f(c)))\}$ – Res$(2, 3)$
. . . de itt rájövünk, hogy $\neg q(f(y), g(y))$ nem fog $q(c, \ldots)$ alakút kiütni. Újra.

e) $\{\neg q(f(f(c)), g(f(c))), \neg p(f(c), f(c))\}$ – harmadik klóz, $[x/f(c)]$

f) $\{\neg q(f(f(c)), g(f(c)))\}$ – Res$(2, 5)$

g) $\{\neg p(f(f(f(c))), f(c)), q(f(f(c)), g(f(c)))\}$ – második klóz, $[x/f(f(c))][z/f(c)]$

h) $\{\neg p(f(f(f(c))), f(c))\}$ – Res$(6, 7)$

i) $\{p(f(f(f(c))), f(c))\}$ – első klóz, $[x/f(f(f(c)))][y/c]$

j) $\square$ – Res$(8, 9)$

Tehát a formula valóban kielégíthetetlen.

g)

$$\forall x \forall y \forall z \big(p(f(x), y) \\ \wedge (p(x, g(z)) \rightarrow q(g(x), f(z))) \\ \wedge (\neg q(y, x) \vee \neg p(x, g(y))) \big)$$

Klózok:

$$\{p(f(x), y)\},\ \{\neg p(x, g(z)), q(g(x), f(z))\},\ \{\neg q(y, x), \neg p(x, g(y))\}$$

Alaptermek: $T_0 = \{c, f(c), g(c), f(f(c)), f(g(c)), \ldots\}$

a) $\{p(f(c), g(c))\}$ – első klóz, $[x/c][y/g(c)]$

b) $\{\neg p(f(c), g(c)), q(g(f(c)), f(c))\}$ – második klóz, $[x/f(c)][z/c]$

c) $\{q(g(f(c)), f(c))\}$ – Res$(1, 2)$

d) $\{\neg q(g(f(c)), f(c)), \neg p(f(c), g(g(f(c))))\}$ – harmadik klóz, $[x/f(c)][y/g(f(c))]$

e) $\{\neg p(f(c), g(g(f(c))))\}$ – Res$(3, 4)$

f) $\{p(f(c), g(g(f(c))))\}$ – első klóz, $[x/c][y/g(g(f(c)))]$

g) $\square$ – Res$(5, 6)$

Tehát a formula valóban kielégíthetetlen.

h)

$$\forall z \exists y \forall x \big((\neg p(z, a) \vee \neg p(z, x) \vee \neg p(x, z)) \\ \wedge ((p(z, y) \wedge p(y, z)) \vee p(z, a)) \big)$$

Skolemizáláskor y helyére $f(z)$ kerül. Disztributivitást is alkalmazunk. Klózok:

$$\{\neg p(z, a), \neg p(z, x), \neg p(x, z)\},\ \{p(z, f(z)), p(z, a)\},\ \{p(f(z), z), p(z, a)\}$$

Alaptermek: $T_0 = \{a, f(a), f(f(a)), f(f(f(a))), \ldots\}$

a) $\{\neg p(a, a)\}$ – első klóz, $[x/a][y/a][z/a]$

b) $\{p(a, f(a)), p(a, a)\}$ – második klóz, $[z/a]$

c) $\{p(a, f(a))\}$ – Res$(1, 2)$

d) $\{\neg p(f(a), a), \neg p(a, f(a))\}$ – első klóz, $[z/f(a)][x/a]$

e) $\{\neg p(f(a), a)\}$ – Res$(3, 4)$

f) $\{p(f(a), a), p(a, a)\}$ – harmadik klóz, $[z/a]$

g) $\{p(a, a)\}$ – Res$(5, 6)$

h) $\square$ – Res$(1, 7)$

Tehát a formula valóban kielégíthetetlen.

11. fejezet

Az egyesítési algoritmus és az elsőrendű rezolúció

Elméleti összefoglaló

Az alap rezolúciós algoritmus helyes és teljes, de a keresési tér túlzottan nagy. Azzal, hogy már példányosításkor ,,jó előre" el kell döntenünk minden változó esetében, hogy majd sok lépéssel később melyik alapterm helyettesítése bizonyul jó ötletnek, intuitíve is érezhető módon lecsökkennek az esélyeink arra, hogy (,,jó" heurisztika mellett) elkerüljünk ,,zsákutcákat" a levezetés során.
Ezt (részlegesen) kiküszöbölni hivatott az *elsőrendű rezolúció*, melynek a rezolvensképzés művelete összetettebb (közben végre kell hajtsuk az *egyesítési algoritmust*), de nem kell előre helyettesítéseket ,,tippelnünk".
Az egyesítési algoritmus a következő: input egy C klóz, output egy $s = [x_1/t_1][x_2/t_2]\dots[x_n/t_n]$ helyettesítés-sorozat (melyet a továbbiakban szintén helyettesítésnek nevezünk, s mint SUBSTITUTION), melyre $|Cs| \leq 1$ – tehát mely ,,összeejti" a C-beli literálokat, ha ilyen helyettesítés egyáltalán van; ha nincs ilyen, akkor jelezze, hogy NEM EGYESÍTHETŐ.
Az olyan s-eket, melyekre $|Cs| \leq 1$ (egyébként nemüres klózokra ez a méret nyilván konkrétan 1 lesz, az üres klózra pedig 0), a C klóz *egyesítőjének* nevezzük. Az egyesítési algoritmus még azt is tudja, hogy ezek közül az ún. *legáltalánosabb egyesítőt* adja vissza, mellyel mindegyik másik egyesítést elkezdhetünk: formálisan, az s egyesítő a C klóz legáltalánosabb egyesítője, ha bármelyik (másik) s' egyesítőre találunk olyan s'' ,,folytatását" az s-nek, amire $ss'' = s'$.
Az algoritmus pedig:

$s := []$. ▷ Itt [] az üres helyettesítés.
while $|C| > 1$ **do**
    Vegyünk két különböző literált, ℓ_1-et és ℓ_2-t C-ből.

Vegyük az első olyan pozíciót, ahol ℓ_1 és ℓ_2 különbözik.

if itt az egyik literálban egy x változó áll, a másikban egy olyan t term kezdődik, melyben nincs x **then**

$C := C[x/t]$, $s := s[x/t]$

else

return NEM EGYESÍTHETŐ

return s

Implementációs kérdések: egyesítés

- Itt is, ha mezei stringátíró módszereket használunk, könnyen előfordulhat, hogy exponenciális méretű klózt kell nyilvántartsunk. Ehelyett (a formuláknál megismert módon) a termeket és literálokat is DAG-reprezentációban, az azonos résztermeknek egyetlen közös csúcspontot létrehozva elérhetjük, hogy az algoritmus futása során a klóz mérete *egyáltalán ne növekedjen* – hiszen minden helyettesítésnél csak egy változócímkéjű gyerek csúcsot kell átlinkeljünk egy már létező másik pozícióra. A „t-ben nincs x" kitétel pedig biztosítja, hogy kört továbbra sem fogunk létrehozni így a gráfban.

- Az occurs checket elég sokszor futtathatjuk, így megéri gyorsítani pl. a következő módon: minden csúcspontban tároljuk egy `set<Var>`-ban, hogy mely változók szerepelnek az adott résztermben. Létrehozáskor ezt viszonylag gyorsan elő tudjuk állítani. Mikor egy változót lehelyettesítünk, az a klózból teljesen eltűnik (mivel t-ben nincs x), így nem baj, ha az x benne marad ezekben a `set`ekben, hiszen nem fogjuk többé keresni.

- Az „összes változó helyettesítése" kis időigényben megoldható úgy, hogy a változó címkéjű csúcsok változónként csak egy példányban jöjjenek létre egy ismert referencián. Ekkor nem kell átlinkelnünk őket, hanem csak azt az egyetlen referenciát átírni (amennyiben nem immutable a term osztályunk implementációja). Ha immutable, akkor pedig érdemes a változók felől egy backlink-listát nyilvántartani, így hatékonyan meg fogjuk találni az adott x változó összes előfordulását.

Megjegyzés. A PROLOG programban az occurs-check alapértelmezetten ki van kapcsolva, vélhetően pont a művelet költséges mivolta miatt. Ily módon a beleintegrált egyesítési algoritmus nem korrekt.

Tehát az egyesítési algoritmus visszaad egy legáltalánosabb egyesítőt, ha van.

Ezek után az elsőrendű rezolvensképzés (kicsit optimalizált) alakja:

Input: C_1, C_2 klózok.

Nevezzük át (mondjuk) C_1 változóit úgy, hogy ne legyen a két klózban közös változó.

Válasszunk ki néhány (legalább egy) pozitív literált C_1-ből, $\ell_1, \ldots, \ell_k$-t és néhány negatívat, $\neg\ell'_1, \ldots, \neg\ell'_t$-t, legalább egyet innen is úgy, hogy a *predikátumjel* megegyezzen mind a $k+t$ literálban.
Próbáljuk meg egyesíteni a $C = \{\ell_1, \ldots, \ell_k, \ell'_1, \ldots, \ell'_t\}$ klózt.
Ha sikerült és a legáltalánosabb egyesítő s, akkor a rezolvens:

$$R = \big((C_1 - \{\ell_1, \ldots, \ell_k\}) \cup (C_2 - \{\neg\ell'_1, \ldots, \neg\ell'_t\})\big)s.$$

Azaz: kiválasztunk egy vagy több $p(\ldots)$ alakú literált C_1-ből, egy vagy több $\neg p(\ldots)$ alakút C_2-ből, megpróbáljuk egyesíteni az előjel nélküli változataikat. Ha sikerült és az eredmény s, akkor ezt az s-t alkalmazzuk a két klózból együtt megmaradt literálokon, ez lesz a rezolvens.
Ezek után az elsőrendű logika rezolúciós algoritmusa: az input ismét egy Σ' elsőrendű klózhalmaz (melyet vélhetően skolemizálással kaptunk).

- Listát vezetünk klózokról.
- Egy klózt kétféle okból vehetünk fel a listára:
  - ha Σ'-beli, VAGY
  - két, a listán már szereplő klóznak rezolvense (az elsőrendű értelemben).

A vonatkozó helyességi és teljességi tétel pedig:

Σ' pontosan akkor kielégíthetetlen,
ha levezethető belőle elsőrendű rezolúcióval az üres klóz.

Feladatok

11.1. Feladat. Igazoljuk elsőrendű rezolúcióval, hogy a következő formulák kielégíthetetlenek! Ahol kell, hozzunk zárt Skolem alakra, CNF maggal.

a) $\forall x(p(x) \wedge \neg p(f(x))$

b)

$$\forall z \exists y \forall x \big((\neg p(z,a) \vee \neg p(z,x) \vee \neg p(x,z)) \\ \wedge ((p(z,y) \wedge p(y,z)) \vee p(z,a)\big)$$

11.1. Feladat megoldása.

a) $\forall x(p(x) \wedge \neg p(f(x))$. A klózok:

$$\Big\{\{p(x)\},\ \{\neg p(f(x))\}\Big\}.$$

Ezeket persze felvesszük első két lépésben:

1 $\{p(x)\}$

2 $\{\neg p(f(x))\}$

Most rezolválni próbáljuk a két klózt. Először is átnevezzük az 1. klózban az x változót mondjuk y-ra, hogy ne ütközzön:

1' $\{p(y)\}$

Eztán $1'$-ből pozitív, 2-ből negatív literálokat válogatunk ki, legalább egyet-egyet egy C klózba előjel nélkül. Sok választás nincs:

$$C = \{p(y), p(f(x))\}$$

Ezt egyesíti az $[y/f(x)]$ egyesítő, tehát tudunk így rezolválni. Az $1' \cup 2$ klóz maradék literáljain kellene végrehajtani ezt az egyesítőt, de nincs maradék literál, így:

3 $\square$, mert $\mathrm{Res}(1, 2)$.

Tehát a formula tényleg kielégíthetetlen.

b)

$$\forall z \exists y \forall x \big((\neg p(z, a) \vee \neg p(z, x) \vee \neg p(x, z)) \\ \wedge ((p(z, y) \wedge p(y, z)) \vee p(z, a)\big)$$

Skolemizáláskor y helyére $f(z)$ kerül. Disztributivitást is alkalmazunk. Klózok:

$$\{\neg p(z, a), \neg p(z, x), \neg p(x, z)\},\ \{p(z, f(z)), p(z, a)\},\ \{p(f(z), z), p(z, a)\}$$

a) $\{\neg p(z, a), \neg p(z, x), \neg p(x, z)\}$

b) $\{p(z, f(z)), p(z, a)\}$

c) $\{p(f(z), z), p(z, a)\}$

d) Megpróbáljuk az első klózból mindegyik literált, a másodikból pedig a $p(z, a)$ literált kiválasztva rezolválni. Előbb a második klózban átnevezzük z-t w-re:

$$C = \{p(z, a), p(z, x), p(x, z), p(w, a)\}$$

Egyesíthető, $[z/a][x/a][w/a]$ az egyesítő, ezt végrehajtjuk a megmaradt $p(z, f(z))$ literálon:
$\{p(a, f(a))\}$, $\mathrm{Res}(1, 2)$.

e) Mondjuk ezt ismét az egyes klózzal próbáljuk rezolválni, például úgy, hogy abból a harmadik literált választjuk ki:

$$C = \{p(x, z), p(a, f(a))\}$$

Egyesíthető, $[x/a][z/f(a)]$ egyesítővel, ezt végrehajtjuk a maradék literálok $\{\neg p(z, a), \neg p(z, x)\}$ halmazán:
$\{\neg p(f(a), a)\}$ Res$(1, 4)$. (A két literál összeesett.)

f) Mondjuk ezt a hármas klóz első literálja mentén próbáljuk rezolválni, persze $[z/a]$ helyettesítéssel sikerül egyesíteni és marad:
$\{p(a, a)\}$ Res$(3, 5)$.

g) Ezt az egyes klóz összes literáljával megpróbáljuk egyszerre kiválasztani:

$$C = \{\neg p(z, a), \neg p(z, x), \neg p(x, z), p(a, a)\},$$

egyesíthető lesz $[z/a][x/a]$-val, nem maradt literál:
$\square$ Res$(1, 6)$.

Tehát a formula tényleg kielégíthetetlen.

11.2. Feladat. Mutassunk olyan $[x_1/t_1][x_2/t_2]$ helyettesítési sorozatot, melyre $[x_1/t_1][x_2/t_2] \neq [x_2/t_2][x_1/t_1]$.

11.2. Feladat megoldása. Pl. $[x/y][y/c]$ más, mint $[y/c][x/y]$, hiszen pl. $p(x, y)$-en alkalmazva őket:

$$p(x, y)[x/y][y/c] = p(c, c)$$
$$p(x, y)[y/c][x/y] = p(y, c),$$

nem egyenlő az eredmény.

11.3. Feladat. Mutassunk olyan $[x_1/t_1][x_2/t_2]$ helyettesítési sorozatot, melyre $[x_1/t_1][x_2/t_2] \neq [x_2/t_2][x_1/t_1]$ és x_1 nem szerepel t_2-ben, sem x_2 t_1-ben!

11.3. Feladat megoldása. Pl. $[x/c][x/d]$ más, mint $[y/d][x/c]$, hiszen pl. $p(x, y)$-en alkalmazva őket:

$$p(x, y)[x/c][x/d] = p(c, y)$$
$$p(x, y)[x/d][x/c] = p(d, y),$$

nem egyenlő az eredmény.

12. fejezet

Az elsőrendű rezolúciós algoritmus variánsai

Elméleti összefoglaló

Az elsőrendű rezolúciós algoritmus során a lényeges döntési kérdés, hogy *melyik* két klóz rezolvensét próbáljuk képezni; ezek után még az is kérdés lehet (ha netán több lehetőségünk is van), hogy mely literálokat válasszuk ki egyesítésre, azaz mely literálok mentén próbáljunk rezolvenst képezni.
A keresési tér további szűkítését érjük el, ha a *lineáris rezolúció* módszerét alkalmazzuk.
Lineáris rezolúció alkalmazása során a megengedett lépések:

- Első lépésben felvesszük Σ egy elemét.
- Minden további lépésben az utolsó lépésben kapott elemet rezolváljuk
  - a listán már szereplő klózok, VAGY
  - Σ elemeinek

  egyikével.

Tehát: elindulunk Σ egyik klózából, ez lesz a levezetés *bázisa*, eztán mindig az utolsó lépésben kapott klózt rezolváljuk egy már rendelkezésre álló klózzal.
Itt is van helyességi és teljességi tétel (mivel rezolúciót csinálunk továbbra is, a helyesség nyilvánvaló): Tetszőleges input klózhalmaz pontosan akkor kielégíthetetlen,
ha levezethető belőle lineáris rezolúcióval az üres klóz.
Láttuk korábban, hogy a Horn-formulák kielégíthetőségére szolgált egy ,,karikázós" vagy ,,jelölős" algoritmus, mely lényegében nem volt más, mint az egységrezolúció

alkalmazása, pozitív egységklózokkal. Elsőrendű logika esetében nincs ,,karikázós" algoritmus (eltekintve attól az esettől, mikor kizárólag alapklózaink vannak, ekkor ezek literáljai ítéletkalkulusbeli literáloknak is felfoghatók). Ami viszont működik, az az *SLD* (Selective Linear Definite) *rezolúció*: az LD rezolúció olyan lineáris rezolúció, melynek a bázisa egy negatív klóz. (Emlékeztetőül, egy klózt akkor hívtunk negatívnak, ha benne minden literál negatív.)
Ha az input Horn alakú (elsőrendű logikában a Horn-klóz ugyanúgy, mint ítéletkalkulusban, olyan klóz, mely legfeljebb egy pozitív literált tartalmazhat), akkor könnyen látszik, hogy a negatív klózt egy program klózzal (ez pedig olyan klóz, mely pontosan egy literált tartalmaz) lehet rezolválni, mely esetben az eredmény negatív klóz lesz. Így tehát a listán mindvégig negatív klózok lesznek. Ez azt eredményezi, hogy SLD rezolúciót Horn-klózok Σ halmazán futtatva az aktuális elemet mindig egy Σ-beli klózzal kell rezolváljuk! Ez jelentősen lecsökkenti a keresési teret.
Nem rontja el viszont a teljességet, amennyiben az input tényleg Horn:

Horn-klózok egy Σ halmaza pontosan akkor kielégíthetetlen,
ha levezethető belőle SLD rezolúcióval az üres klóz.

Felmerülhet a kérdés, hogy mennyire motivált a Horn-klózokkal kapcsolatos következtetés-optimalizálás a gyakorlatban? A válasz: **nagyon!** A *logikai programozásban* az alapfeladat: input egy *logikai program*, ami definit klózoknak egy Σ halmaza és egy *kérdés*, ami pedig $P = P_1 \wedge P_2 \wedge \ldots \wedge P_k$ alakú formula, ahol a P_i-k atomi formulák.
A kérdés pedig, hogy $\Sigma \models \exists P$ igaz-e, és ha igen, akkor mondjunk is egy olyan s helyettesítést, melyek mellett Ps igaz.
Az SLD rezolúció pontosan az ilyen alakú kérdések megválaszolására alkalmas, hiszen ha negáljuk a kérdést, akkor épp egy univerzálisan kvantifikált *kérdésklózzá* (=negatív klózzá) válik, így a logikai programunkkal együtt egy Horn-klózhalmazunk van, melyben az egyetlen negatív klóz éppen a kérdésklóz! Így ebből indítunk egy SLD rezolúciót és ha kijön az üres klóz, valóban következménye a kérdés a programnak. Ha közben nyilvántartjuk az aktuális helyettesítést (technikailag nem csak az aktuális munkaklózt, de az eddigi összesített helyettesítést is nyilván kell tartanunk, egy ilyen (C, s) párt nevezünk a program egy *konfigurációjának*), akkor a sikeres futás végén (a futás attól sikeres, hogy a program utolsó konfigurációja $(\Box, s)$ alakú, vagyis megkaptuk az üres klózt) csak visszaadjuk az s helyettesítést. Vagy már végre is hajthatjuk az eredeti kérdésen és visszaadhatjuk a Ps formulát, ez a program *eredménye*.
A Prolog pontosan ezt csinálja. Valójában a Prolog nem más, mint egy SLD rezolúciós motor, melyben

- $\{\neg p_1, \ldots, \neg p_k, q\}$ alakú klózokat `q :-` p_1`,` p_2`,...,` p_k formában adunk meg, ha $k > 0$;

- $\{q\}$ alakú klózokat `q.` formában adunk meg;
- a $\{\neg p_1, \ldots, \neg p_k\}$ kérdésklózt pedig `?-` p_1`,` `...,` p_k formában tesszük fel.

A PROLOG verem-alapú literálszelekciót is végez (az S betű az SLD névben arra utal, hogy valamiféle szelekciós függvény is van, mely tovább szűkítheti a keresési teret, így pl. eldöntheti, hogy a munkaklóz melyik literálja mentén rezolváljon a motor). Tehát mindig a legfrissebben bekerült literál mentén próbál rezolválni, mikor pedig egy programklózzal rezolvál, akkor a programklóz *törzsének* (a `:-` utáni rész a klóz törzse, az előtti rész a *feje*) a literáljait hátulról előre haladva teszi be a verembe, így az oldalklóz törzsének első literálja lesz a legfrissebb.
Nézzünk egy példát: adjunk ösze unáris számrendszerben egyet és kettőt PROLOGban.
Először is, unárisan a számokat egy 0 konstansjellel és egy f egyváltozós függvényjellel ábrázolhatjuk, pl. $f(f(f(0)))$ jelöli a hármat; az f szemantikájának az ,,adj hozzá egyet" interpretációt szánjuk, de ezt nem tudjuk betanítani a logikai motornak. (Más kérdés, hogy a PROLOG motorja számolni azért tud, tehát ezt épp oda is adhatnánk neki.)
Az összeadást egy A (ADDITION) ternáris predikátummal reprezentáljuk a szokott módon: $A(x, y, z)$ akkor kellene igaz legyen, ha $x + y = z$. Axiomatizáljuk, amit tudunk:

$$\{\forall x A(x, 0, x), \forall x \forall y \forall z A(x, y, z) \to A(x, f(y), f(z))\}.$$

Világos: $x + 0 = x$ és $x + (y + 1) = (x + y) + 1$, ezek vannak felírva. Ez már elég ahhoz, hogy számolni tudjunk unáris számrendszerben. Ha konkrétan az egyet és a kettőt akarjuk összeadni, akkor egy olyan x-et keresünk, melyre $A(f(0), f(f(0)), x)$ igaz.
A programunk a (negált!) kérdésklózzal egyetemben a ,,szokásos", logikai módon felírva, klózhalmazként:

$$\{\{A(x, 0, x)\}, \{\neg A(x, y, z), A(x, f(y), f(z)\}, \{\neg A(f(0), f(f(0)), x)\}\}.$$

Mindez PROLOG szintaxisban:

$$a(x, 0, x).$$
$$a(x, f(y), f(z)) :- a(x, y, z).$$

$$?- \; a(f(0), f(f(0)), x).$$

A PROLOG szelekciójának az is része, hogy szép sorban, fentről lefelé próbál rezolválni (ezzel a PROLOG program írójára hárul az, hogy belőjön egy jónak mondható sorrendet, így viszont lehet szimulálni procedurális nyelvek vezérlési szerkezetét is, ha a PROLOG programozó épp azt szeretné). Tehát a motor...

a) Első lépésben a kérdésklóz $\neg a(f(0), f(f(0)), x)$ literálját (ne felejtsük, attól, hogy nincs kiírva, még negált minden aktuális literálunk – ami a szeparátor karakter jobb oldalán van, azaz a törzs, negatív, a bal oldalon levő fej pozitív) próbálja egyesíteni az első olyan a-val, amit talál. Ez épp az $\{a(x, 0, x)\}$ klóz feje.

b) Nem egyesíthető, ugyan előbb az x/y változó-átnevezést (hogy diszjunktak legyenek a változók a két klózban), majd az $y/f(0)$ helyettesítést elvégzi, de 0 és $f(f(0))$ egyike se változó, backtrack.

c) Most az $\{a(x, f(y), f(z)), \neg a(x, y, z)\}$ programklóz fejével próbálja rezolválni a $\neg a(f(0), f(f(0)), x)$ literált.

   a) Először elvégzi a munkaklózban az $[x/w]$ helyettesítést, hogy különbözzenek a változók.

   b) Egyesíti az $\{a(f(0), f(f(0)), w), a(x, f(y), f(z))\}$ halmazt: $[x/f(0)][y/f(0)][w/f(z)]$ a legáltalánosabb egyesítő.

   c) Ezt végrehajtja a maradék literálokon: $\{\neg a(x, y, z)\}$-n, eredmény: $\{\neg a(f(0), f(0), z)\}$ az eredmény. Ez a munkaklózunk, az aktuális helyettesítésünk pedig $[x/w][x/f(0)][y/f(0)][w/f(z)]$.

d) Most az egyetlen literált, $\neg a(f(0), f(0), z)$-t próbálja ütni, ismét az első a-s fejű, vagyis a $\{a(x, 0, x)\}$ klózt használva.

   Ez nem sikerül, $[x/f(0)]$ még igen, de 0 nem ugyanaz, mint $f(0)$.

e) Második próbálkozásként az $\{a(x, f(y), f(z)), \neg a(x, y, z)\}$ klóz fejével próbál ütni:

   a) Előbb átnevezi a munkaklózban z-t w-re

   b) Eztán egyesíti az $\{a(x, f(y), f(z)), a(f(0), f(0), w)\}$ halmazt. Sikerül: $[x/f(0)][y/0][w/f(z)]$.

   c) Ezt az egyesítőt végrehajtja a maradék literálokon, ismét $\{\neg a(x, y, z)\}$-n.

   d) Az aktuális konfiguráció: $\big(\{\neg a(f(0), 0, z)\}, [x/w][x/f(0)][y/f(0)][w/f(z)][z/w][x/f(0)][y/0][w/f(z)]\big)$.

f) Megint az egyetlen literálját, $\neg a(f(0), 0, z)$-t próbálja ütni, ismét az $\{a(x, 0, x)\}$ klóz fejével.

g) Ezúttal sikerül! Egyesítve az $\{a(f(0), 0, z), a(x, 0, x)\}$ halmazt $[x/f(0)][z/f(0)]$ lesz az egyesítőnk.

h) Az aktuális konfiguráció: $\big(\square, [x/w][x/f(0)][y/f(0)][w/f(z)][z/w][x/f(0)][y/0][w/f(z)][x/f(0)][z/f(0)]\big)$.

i) Mivel az üres klózhoz értünk, a konfigurációban szereplő helyettesítést végrehajtjuk az eredeti kérdésklózon és megkapjuk, hogy $a(f(0), f(f(0)), x)$-ben az x-et w-re, azt $f(z)$-re, abban z-t w-re, azt $f(z)$-e, abban z-t $f(0)$-ra cserélve...

j) $a(f(0), f(f(0)), f(f(f(0))))$. Ezt adja vissza a PROLOG.

k) Láthatjuk: $1 + 2 = 3$.

Feladatok

12.1. Feladat. Igazoljuk lineáris rezolúcióval, hogy az alábbi klózhalmazok kielégíthetetlenek!

a) $\{\{p, q\},\ \{\neg p, q\},\ \{\neg p, \neg q\},\ \{p, \neg q\}\}$.

12.1. Feladat megoldása.

a) $\Sigma = \{\{p, q\},\ \{\neg p, q\},\ \{\neg p, \neg q\},\ \{p, \neg q\}\}$.

   a) $\{p, q\}$ eleme az inputnak
   b) $\{q\}$ – rezolválunk $\{\neg p, q\} \in \Sigma$-val
   c) $\{p\}$ – rezolválunk $\{p, \neg q\} \in \Sigma$-val
   d) $\{\neg q\}$ – rezolválunk $\{\neg p, \neg q\} \in \Sigma$-val
   e) $\square$ – rezolválunk 2-vel.

   Tehát Σ tényleg kielégíthetetlen.

12.2. Feladat. Mutassuk meg lineáris rezolúcióval, hogy az alábbi következtetés helyes!

a)

$$\{\boxed{\text{Segítünk}}, \boxed{\text{Ne hagyja el gépjárművét}}, \boxed{\text{Ha elfogy az üzemanyag, üljön át másik gépjárműbe}}, \boxed{\text{Ha átül másik gépjárműbe, elhagyja gépjárművét}}\} \models \boxed{\text{Nem fogy el az üzemanyag}}!$$

12.2. Feladat megoldása.

a)

$$\{\boxed{\text{Segítünk}}, \boxed{\text{Ne hagyja el gépjárművét}}, \boxed{\text{Ha elfogy az üzemanyag, üljön át másik gépjárműbe}}, \boxed{\text{Ha átül másik gépjárműbe, elhagyja gépjárművét}}\} \models \boxed{\text{Nem fogy el az üzemanyag}}!$$

A ,,Segítünk"-nek s, ,,elhagyja gépjárművét"-nek g, ,,elfogy az üzemanyag"-nak u, ,,átül másik gépjárműbe"-nek pedig m ítéletváltozót megfeleltetve azt kapjuk, hogy a feladat

$$\{s, \neg g, u \to m, m \to g\} \models \neg u,$$

a jobb oldal negáltját és a bal oldali formulákat CNF-re hozva a feladat a

$$\Sigma = \{\{s\}, \{\neg g\}, \{\neg u, m\}, \{\neg m, g\}, \{u\}\}$$

klózhalmaz kielégíhetetlenségének bizonyítása. Induljunk mondjuk a $\{\neg g\}$ bázisklózból:

1 $\{\neg g\} \in \Sigma$

2 $\{\neg m\}$ – rezolváltunk a Σ-beli $\{\neg m, g\}$-vel

3 $\{\neg u\}$ – rezolváltunk a Σ-beli $\{\neg u, m\}$-mel

4 $\square$ – rezolváltunk a Σ-beli $\{u\}$-val.

A fenti lineáris rezolúciós levezetés egyébként SLD rezolúciós levezetés is (és az is látszik belőle, hogy nem számít a következtetés szempontjából, hogy segítenek-e vagy sem), az input pedig Horn alakú.

12.3. Feladat. Adjon a Hanoi Tornyai problémára PROLOG programot.
Érdemes tudni, hogy a `write` unáris predikátumhoz mikor a PROLOG motor ér, kiüti és kiírja az argumentumot mellékhatásként. Újsor karaktert kiíratni az `nl` predikátummal lehet.

12.3. Feladat megoldása.

```
hanoi(f(0),X,Y,Z) :- write("Mozgatás:  "), write(X), write("->"),
write(Y), nl.
hanoi(f(f(N)),X,Y,Z) :- hanoi(f(N),X,Z,Y), hanoi(f(0),X,Y,Z),
hanoi(f(N),Z,Y,X).
```

`?-hanoi(f(f(...f(0)...)),"egyes","kettes","hármas").`
Megjegyzés: PROLOGban jobb megoldás, ha a törzsben nem használt változókat (mint az első programklózban a Z) a fejben _ jelzi, úgy a motor tudja, hogy az SLD rezolvensképzéskor az értéket nem kell használni és nem jegyzi meg, gyorsabban fut.
Másfelől itt az N értékét unárisan adjuk meg – a PROLOG számokat is tud kezelni, értékadni pedig az `X is V` predikátummal lehet, ahol az X változó megkapja a V értéket. Ezen a ponton fontos, hogy V már bírjon ground értékkel, X pedig tényleg egy változó legyen.

13. fejezet

A kompaktsági tétel és következményei. Eldönthetetlenségi eredmények

Elméleti összefoglaló

A kompaktsági tétel az egyik leghatékonyabb eszközünk volt ítéletkalkulusban, és fennáll az elsőrendű logikában is:

Tetszőleges Σ formulahalmaz pontosan akkor kielégíthető,
ha bármely **véges** részhalmaza kielégíthető.

Vagyis, ha egy input végtelen nagy Σ-ról sikerül legalább azt belátni, hogy minden **véges** részhalmaza kielégíthető, akkor az egész is az. A tételt a következő két formában szoktuk alkalmazni (mert általában nem kielégíthetőséget, hanem pont hogy kielégíthetetlenséget akarunk bizonyítani):

Ha Σ kielégíthetetlen,
akkor van **véges** kielégíthetetlen részhalmaza is.

és

Ha $\Sigma \models F$,
akkor már Σ egy véges részhalmazának is következménye F.

A fenti három alak nyilván ekvivalens (a második és harmadik variánsban csak a tétel lényegi részét mondtuk ki, a másik irány nyilvánvaló). Egyfelől ez jó, mert ez biztosítja azt, hogy működhet a rezolúciós algoritmus, mely futása során egyszerre természetesen csak véges sok klózt fog tárolni, így ha egy végtelen Σ

halmaz végtelen sok eleme *kellene* ahhoz, hogy Σ kielégíthetetlenségét bizonyítani tudjuk, nem lenne ezt algoritmikusan félig se esélyünk eldönteni.
Másfelől a kompaktsági tétel egyfajta ,,gyengeség" is, mégpedig abból a szempontból, hogy számos tulajdonság nem fejezhető ki elsőrendű logikában, akár végtelen nagy formulahalmazzal sem.
Tegyük fel, hogy szeretnénk elsőrendű logikai formulákat felírni, mondjuk a 0, 1 konstansok, $+$, $*$, bináris függvényjelek, a $<$ és $=$ bináris predikátumjelek és esetleg még más jelek (bevehetjük kedvenc függvényeinket, predikátumainkat, bármit) használatával, de úgy, hogy amit felírunk, az (izomorfizmus erejéig) **kizárólag** a természetes számok struktúrájára legyen igaz! (Pl. felírhatunk formulákat az előző fejezet Prolog példaprogramjához hasonlóan az összeadás természetéről, vagy a Peano axiómákat és még számos másikat a műveletekről stb., az mind igaz a természetes számok struktúrájára, de vajon van még olyan lehetséges interpretáció mondjuk ugyanezen az alaphalmazon, melyre szintén igaz?)
A kompaktsági tétel miatt ezt nem tudjuk megtenni, ugyanis:

- Tegyük fel, hogy a természetes számok struktúrája, $\mathcal{N}$ kielégíti a Σ formulahalmazt.
- Meg fogjuk mutatni, hogy van egy Σ-nál bővebb Δ halmaz, amit $\mathcal{N}$ nem elégít ki, de ami viszont kielégíthető, tehát van egy $\mathcal{B}$ modellje, ami nem izomorf $\mathcal{N}$-nel.
- Ez a $\mathcal{B}$ modell modellje Σ-nak is, tehát *nem standard* modell.
- Ha még az alaphalmazzal kapcsolatosan is vannak igények (pl. hogy legyenek ott is a természetes számok az objektumok), az nem baj: Herbrand tételének egy következménye azt mondja, hogy kielégíthető formulahalmaznak van megszámlálható modellje. Nem nehéz olyan Σ formulahalmazt felírni, melynek csak végtelen modellje van: ekkor Δ-nak van megszámlálhatóan végtelen modellje, ha ezt választjuk $\mathcal{B}$-nek, és az univerzumát tetszőleges módon bijekcióba hozzuk a természetes számok halmazával, akkor az alaphalmaz is ugyanaz lesz, mint $\mathcal{N}$-nek.
- A ,,megmutatjuk" rész: jelölje az $((1+1)+1)+1\ldots+1)$ termet az $\underline{n}$ jelölés, ahol n-szer adtuk össze az 1-et. A standard modellben persze ennek az értéke épp n lesz. Ha $n=0$, legyen ez a 0 term.
- Vegyünk még egy új konstansjelet is be a nyelvbe, legyen ez mondjuk c.
- Jelölje F_n a $c \neq \underline{n}$ (avagy $\neg = (c, \underline{n})$) formulát.
- Nézzük a $\Delta = \Sigma \cup \{F_i : i \geq 0\}$ formulahalmazt.

- Ennek akármelyik véges részhalmazát nézzük, kielégíthető: egy véges részhalmaz véges sok Σ-beli és véges sok F_i alakú formulát tartalmaz. Akkor van egy maximális N érték, hogy F_N benne van ebben a részhalmazban. Akkor ezt a véges formulahalmazt kielégíti az $\mathcal{N}$ struktúra azzal, hogy legyen az új c konstansjel mondjuk az $N+1$-gyel interpretálva.

- Tehát mivel Δ minden véges részhalmaza kielégíthető, így Δ is.

- Viszont Δ-nak $\mathcal{N}$ semmiképp nem lesz modellje, bárhogy is interpretáljuk benne c-t! Hiszen $\mathcal{N}$-ben minden lehetséges értéke c-nek előáll $\underline{n}$ alakban, ami ellene megy F_n-nek. Azaz van Δ-nak egy $\mathcal{B}$ modellje, melynek „c nélküli" része nem izomorf $\mathcal{N}$-nel.

- Ez a $\mathcal{B}$ tehát Σ-nak is egy modellje, mely nem izomorf $\mathcal{N}$-nel.

Számos hasonló következmény ismeretes, hasonlóan megmutatható például, hogy:

- Ha Σ-nak van akármekkora nagy véges modellje, akkor van végtelen modellje is.

- Így pl. a „véges csoportok" nem axiomatizálhatóak.

- Ha egy struktúraosztály axiomatizálható *egyáltalán* végesen, akkor bármely axiómarendszerének van véges axióma-részrendszere.

- Emiatt pl. a „0 karakterisztikájú testek" (azok a testek, melyben a 0 nem áll elő az 1 ismételt összeadása során – pl. a $\mathbb{Z}_p$ testek nem ilyenek) szintén nem axiomatizálhatóak.

A feladatgyűjtemény során már többször utaltunk rá, hogy a kielégíthetetlenség csak félig eldönthető, azaz van algoritmus (pl. az alap- vagy az elsőrendű rezolúció), mely kielégíthetetlen Σ formulahalmazt kapva előbb-utóbb valóban jelzi, hogy Σ kielégíthetetlen, azonban a kielégíthető formulahalmazokra minden ilyen algoritmus szükségképpen vagy hibás választ kell adjon, vagy végtelen ciklusba kell essen (a kettő közül inkább a végtelen ciklus...)
Ennek oka a Post Megfelelkezési Probléma (PCP, Post Correspondence Problem) eldönthetetlensége (pontosabban, félig eldönthetősége). A problémának inputja véges sok *dominó-típus*, minden dominó alsó ill. felső felén egy-egy a bináris $\{0,1\}$ ábécé fölötti szó szerepel. A kérdés: le tudunk-e rakni valahány dominót egymás mellé, legalább egyet, minden dominó-típusból akármennyit felhasználva közben úgy, hogy a lerakott dominó-sorozat alsó és felső sorában ugyanazt a szót lássuk?
Meg lehet adni egy konstrukciót, egy ún. rekurzív visszavezetést avagy választartó inputkonverziót, mely egy input dominótípus-szekvenciából elkészít egy elsőrendű

formulát úgy, hogy a dominók a PCP egy IGEN példányát pontosan akkor adják, ha az elkészített elsőrendű formula tautológia.

Feladatok

13.1. Feladat. Van-e a PCP alábbi példányainak megoldása? Ha igen, adjon egyet, ha nem, miért nincs?

a) $\binom{0}{100}$, $\binom{01}{00}$, $\binom{110}{11}$.

13.1. Feladat megoldása.

a) $\binom{0}{100}$, $\binom{01}{00}$, $\binom{110}{11}$-nak van: a

$$\binom{110}{11}\binom{01}{00}\binom{110}{11}\binom{0}{100}$$

sorozat (dominó-indexekkel: 3., 2., 3., 1.) alul is és felül is az 110011100 szó áll elő.

14. fejezet

Másodrendű és heterogén elsőrendű logika

Elméleti összefoglaló

Mint azt a kompaktsági tétel esetében láttuk, bizonyos tulajdonságok kifejezésére az elsőrendű logika nem képes, ilyen például a természetes számok struktúrájának izomorfizmus erejéig egyértelmű axiomatizálása.
Az elsőrendű logika egyféle erősítése az ún. *másodrendű* logika. A másodrendű logikában ugyanazokat a konstrukciókat használhatjuk (elsőrendű változók, függvény- és predikátumjelek, logikai konnektívák stb), mint az elsőrendűben, plusz:

- van egy újabb, *Var*-tól diszjunkt halmaza a változóknak, melyet $PVar$-nak nevezünk;
- a $PVar$-beli változók ún. *predikátumváltozók* (vagy másodrendű változók) és mindegyiknek van *aritása*;
- a predikátumváltozókat is kvantálhatjuk az $\exists$ ill. $\forall$ kvantorokkal;
- ha X egy n-aritású predikátumváltozó és $t_1, \ldots, t_n$ termek, akkor $X(t_1, \ldots, t_n)$ is atomi formula.

Szintaktikailag tehát a termek halmaza ugyanaz, mint korábban, a formulák képzési szabályai pedig kiegészülnek még az $X(t_1, \ldots, t_n)$, $\exists XF$ és $\forall XF$ alakú formulákkal (konvenció: a másodrendű változókat nagy $X, Y, \ldots$ jelöli).
Szemantikailag, mint a neve is sugallja, egy n aritású predikátumváltozó egy n aritású predikátumot vesz fel értékül. Ehhez egyrészt az elsőrendű struktúra fogalmát kell kiegészítenünk: a másodrendű struktúra ugyanúgy $\mathcal{A} = (A, I, \varphi)$, ahol

A és I ugyanazok, mint eddig, de φ ezúttal minden x elsőrendű változóhoz egy $\varphi(x) \in A$ elemet, és minden X n-aritású másodrendű változóhoz egy $\varphi(X) : A^n \to \{0,1\}$ predikátumot, azaz egy $\varphi(X) \subseteq A^n$ relációt rendel.
A kiértékelés szemantikája pedig:

$$\begin{aligned} \mathcal{A} \models X(t_1,\ldots,t_n) &\Leftrightarrow \varphi(X)(\mathcal{A}(t_1),\ldots,\mathcal{A}(t_n)) = 1; \\ \mathcal{A} \models \exists X\, F &\Leftrightarrow \text{van olyan } R \subseteq A^n, \text{ melyre } \mathcal{A}_{[X\mapsto R]}(F) = 1; \\ \mathcal{A} \models \forall X\, F &\Leftrightarrow \text{minden } R \subseteq A^n\text{-re } \mathcal{A}_{[X\mapsto R]}(F) = 1. \end{aligned}$$

A másodrendű logikára pl. nem igaz a kompaktsági tétel, így bár sok minden kifejezhető benne, teljes következtető rendszerről nem is álmodhatunk benne.
Egy másik kiterjesztés a *heterogén logika*. Az elsőrendű logikában ,,minden objektum egyenlő", nincsenek objektum-típusok, az univerzum egy homogén egységet képez, a függvényekről csak annyit tudunk, hogy az aritásnak megfelelő darabszámú objektumból készít még egyet, a predikátumok szintén az aritás szerinti számú objektumból egy bitet.
Árnyalja a képet, ha típusolunk mindent, a változókat, a függvények(jel)et és a predikátum(jel)eket is: amit így kapunk, azt nevezzük heterogén logikának. Formálisabban:

- Bevezetjük típusok egy S (Sorts) halmazát (nemüres, nem is túl nagy, vagyis megszámlálható)
- Minden változót ellátunk egy típussal: formálisan, minden $s \in S$ típushoz adva van egy Var_S halmaz, megszámlálhatóan végtelen, ennek elemei az s típusú változók.
- Minden függvényjelet ellátunk egy kimeneti típussal és az aritásának megfelelő számú bemeneti típussal: formálisan egy n aritású függvényjel ,,típusa" egy $(s_1,\ldots,s_n,s)$ sorozat lesz, ahol s_i az i. argumentum elvárt típusa lesz majd, az s pedig a kimenet típusa;
- Minden predikátumjelet is ellátunk az aritásának megfelelő számú bemeneti típussal, így az n aritású predikátumjelek típusa egy $(s_1,\ldots,s_n)$ sorozat lesz, ahol s_i az i. argumentum elvárt típusa.

A nyelv ennyit változik. A szintaxisban már a termek és formulák szintén is vannak ezúttal változások: a termek kimeneti típus szerint kerülnek osztályozásra. Az s típusú termek halmaza a legszűkebb olyan halmaz, melyre a következők fennállnak:

- Minden s típusú változó egyben s típusú term is.

- Ha f egy $(s_1, \ldots, s_n, s)$ típusú függvényjel és minden $1 \leq i \leq n$-re t_i egy s_i típusú term, akkor $f(t_1, \ldots, t_n)$ egy s típusú term.

Az atomi formulák esetében pedig: a heterogén elsőrendű logika atomi formulái $p(t_1, \ldots, t_n)$ alakúak, ahol p egy $(s_1, \ldots, s_n)$ típusú predikátumjel, t_i pedig s_i típusú term minden $1 \leq i \leq n$-re.
A szintaxis ennyit változik. A szemantikában is vannak változások, hiszen már egy struktúrában is meg kell jelenniük a típusoknak. Ennek megfelelően egy struktúra most egy $\mathcal{A} = ((A_s)_{s \in S}, I, \varphi)$ hármas, ahol

- minden s-re A_s az s típusú objektumok nemüres alaphalmaza;
- I minden $(s_1, \ldots, s_n)$ típusú p predikátumjelhez egy $I(p) : A_{s_1} \times A_{s_2} \times \ldots \times A_{s_n} \to \{0, 1\}$ predikátumot és minden $(s_1, \ldots, s_n, s)$ típusú f függvényjelhez egy $I(f) : A_{s_1} \times A_{s_2} \times \ldots \times A_{s_n} \to A_s$ függvényt rendel;
- $\varphi(x)$ pedig egy A_s-beli elem minden s típusú x változóra.

A termek kiértékelése, formulák szemantikája szintén értelem szerint változik.
A heterogén logikában felírt mondatok közelebb vannak ahhoz, mint amit az emberi nyelvben felírt formalizmus tükröz (pl. a ,,mindenkinek van egy álma" mondat nem csupán annyit mond, hogy $\forall x \exists y \text{ÖVÉ}(x, y)$, hanem még annyit is sugall, hogy x ember, y pedig álom típusú változó), ily módon a benne történő formalizálás is kényelmesebb (másik példa a geometria, ahol pontokról, egyenesekről és síkokról beszélünk, ekkor ezt a három típust érdemes legalább elkülöníteni).
A heterogén logika annyival tehát előnyösebb, mint az elsőrendű logika, hogy a szakértői rendszerek kiépülését jobban segíti. Továbbá az is igaz, hogy *beágyazható* elsőrendű logikába: a típusokat unáris predikátumjelekkel ,,szimulálva" minden heterogén logikai formulát át lehet írni egy lineáris méretű elsőrendű logikai formulába oly módon, hogy a két formula modelljei egy-egy kapcsolatba legyenek állíthatóak. Ily módon számításelméleti szempontból minden tétel és állítás, ami az elsőrendű logikára igaz volt, a heterogén logikára is igaz lesz.

Feladatok

14.1. Feladat. Mutassa meg, hogy a másodrendű logikában nem igaz a kompaktsági tétel.

14.1. Feladat megoldása. A kompaktsági tétel fejezetében követett okoskodás alapján ha sikerül leírni a természetes számok $\mathcal{N}$ struktúráját izomorfizmus erejéig, azzal igazoljuk az állítást. Ehhez pedig az ottaniak szerint elég formalizálni azt,

hogy ,,az univerzum minden eleme előáll $\underline{n}$ alakban". (Emlékeztetőül: $\underline{n}$ jelölte a $(\ldots((0+1)+1)+1\ldots)+1$ termet, n darab összeadással.)
Először is megfogalmazzuk halmazelméleti okoskodással, mit is akarunk kifejezni:

,,Ha az univerzum egy részhalmazában benne van a 0 és ha n benne van, akkor $n+1$ is, akkor bizony mindenki benne van"

A ,,halmaz" pedig egy unáris predikátum, itt most épp változó. Azaz a formulánk:

$$\forall X\Big(\big(X(0)\wedge\forall x(X(x)\to X(x+1))\big)\to\forall x X(x)\Big).$$

14.2. Feladat. Írja fel a projektív síkok axiómáit heterogén elsőrendű logikai formulákkal:

- Bármely két különböző ponthoz létezik pontosan egy, rajtuk áthaladó egyenes;
- Bármely két különböző egyenesnek létezik pontosan egy metszéspontja;
- Létezik legalább négy pont úgy, hogy ezek közt nincs három kollineáris.

14.2. Feladat megoldása. Megadjuk először a nyelvet.
Típusok: a P (Pontok) és E (Egyenesek) típus.
Predikátumok: $=_P$ és $=_E$ az egyenlőség a két típuson és $\in$, ami egy (P,E) típusú predikátum (szemantikája: (p,e) igaz, ha p rajta van e-n)
Ha ezek megvannak, akkor a formulák:

- Bármely két különböző ponthoz létezik pontosan egy, rajtuk áthaladó egyenes:
$$\forall x_P\forall y_P\Big((x_P\neq_P y_P)\to\big(\exists! e_E\, x_P\in e_E\wedge y_P\in e_E\big)\Big)$$
- Bármely két különböző egyenesnek létezik pontosan egy metszéspontja;
$$\forall x_E\forall y_E\Big((x_E\neq_E y_E)\to\big(\exists! p_P\, p_P\in x_E\wedge p_P\in y_E\big)\Big)$$
- Létezik legalább négy pont úgy, hogy ezek közt nincs három kollineáris.
$$\exists x_P^1\exists x_P^2\exists x_P^3\exists x_P^4\Big(\bigwedge_{1\le i<j\le 4}(x_i\neq_P x_j)\ \wedge\ \neg\exists y_E\bigvee_{1\le i\le 4}\bigwedge_{j\neq i}x_j\in y_E\Big).$$

Mint korábban, $\exists! x F(x)$ itt is az $\exists x(F(x)\wedge\forall y(F(y)\to(x=y)))$ formulát rövidíti.

14.3. Feladat. Adjunk a gráfok nyelvén másodrendű formulákat, melyek azt fejezik ki, hogy az input gráf...

a) ... páros.

b) ... 3-színezhető.

c) ... véges.

d) ... tartalmaz Hamilton-utat.

14.3. Feladat megoldása.

a) A G gráf páros, ha csúcsai két osztályba sorolhatók úgy, hogy minden él a két osztály közt megy. Vagyis, van a csúcsoknak egy halmaza úgy, hogy minden él egyik végpontja a halmazban, a másik azon kívül van:

$$\exists X\Big(\forall x\forall y\big(e(x,y)\to(X(x)\leftrightarrow\neg X(y))\big)\Big)$$

b) A G gráf 3-színezhető, ha csúcsai három osztályba sorolhatók az előbbi módon. Ehhez már nem lesz elég egy halmaz, veszünk egy X_1, egy X_2 és egy X_3 halmazt és ezekbe szétosztjuk a csúcsokat:

$$\begin{aligned}\exists X_1\exists X_2\exists X_3\Big(&\forall x X_1(x)\vee X_2(x)\vee X_3(X)\\&\wedge\forall x(\neg X_1(x)\vee\neg X_2(x))\wedge(\neg X_1(x)\vee\neg X_3(x))\wedge(\neg X_2(x)\vee\neg X_3(x))\\&\wedge\forall x\forall y\big(e(x,y)\to\bigwedge_{1\le i\le 3}\neg(X_i(x)\wedge X_i(y))\big)\Big)\end{aligned}$$

Az első sor azt mondja, hogy minden csúcsot beteszünk a három halmaz legalább egyikébe, a második azt, hogy legfeljebb egyikébe (eddig tehát particionáltuk a pontokat), a harmadik pedig hogy bármelyik élt is nézzük, annak a két végpontja nem lesz ugyanabban a halmazban.

c) ... véges. A véges halmazok pont azok, akiket nem lehet szürjektíven saját maguk valódi részhalmazába beleképezni. Tehát, „minden szürjektív $V\to V$ leképezés injektív is". Egy leképezést egy kétváltozós funkcionális relációval váltunk ki: $P(x,y)$ akkor kéne igaz legyen, ha $x\mapsto y$.

Tehát:

$$\forall P\Big(\forall x\exists!yP(x,y)\wedge\forall y\exists xP(x,y)\to\forall x_1\forall x_2\forall y(P(x_1,y)\wedge P(x_2,y)\to x_1=x_2)\Big)$$

d) ... tartalmaz Hamilton-utat. Azt kell pl. formalizáljuk, hogy a csúcsokon lehet definiálni egy rendezést úgy, hogy minden csúcsból vezet él a közvetlen rákövetkezőjébe:

$$\exists P(\text{ordering}(P) \wedge \forall x \forall y(((P(x,y) \wedge \neg \exists z(P(x,z) \wedge P(z,y))) \rightarrow e(x,y)),$$

ahol ordering(P) azt mondja, hogy P irreflexív, tranzitív, trichotóm, ld. bevezető fejezet.

15. fejezet

Temporális logika, formális verifikáció

Elméleti összefoglaló

A formális verifikáció feladata a következő: adott egy M modell (valamilyen formalizmusban, eddig modellek az elsőrendű logika struktúrái voltak), egy F formula (valamilyen logikában, eddig láttuk az elsőrendű logikát, ennek szeletét, az ítéletkalkulust és két kiterjesztését, a heterogén és a másodrendű logikákat), a kérdés, hogy teljesül-e $M \models F$?
Mint tudjuk, elsőrendű logikában ez a kérdés (végtelen struktúrák esetén) reménytelenül elbonyolódhat, hisz már a természetes számok alaphalmazán a szorzás, összeadás függvények jelenléte mellett az egyenlőséges elsőrendű logika modell-ellenőrzési feladata eldönthetetlen. Erre alapvetően két megoldás kínálkozik:

- Adjunk meg egyszerűbb struktúrákat és egyszerűbb logikát, mely még mindig elég kifejező számunkra, ám a verifikáció feladata legalábbis eldönthető benne. A legjobb persze egy hatékony modellellenőrző algoritmus megléte.
- Adjunk meg félautomatikus módszert, mely szakértő segítségével legalább azt képes megmutatni, ha a rendszer valóban teljesíti a specifikációt (azaz a modell kielégíti a formulát).

Az első lehetőség körébe tartozik a Kripke struktúrákon értelmezett CTL (Computation Tree Logic): a modellek (egy – informatikai – rendszer absztrakt leírása) a *Kripke struktúrák*. Először is rögzítünk egy AP (Atomic Proposition) véges halmazt. Intuitíve a rendszer minden pillanatban ezen tulajdonságok némelyikét teljesíti, némelyiket pedig nem. (Az operációs rendszerekből jól ismert program-életciklus esetében pl. lehet egy-egy ilyen tulajdonság

a FUTÁSKÉSZ, FUTÓ, VÁRAKOZÓ; erőforrásonként beállíthatunk egy-egy AP-elemet, mely akkor igaz a rendszer egy állapotában, ha éppen a rendszer birtokolja az adott erőforrást, stb.)
Ezek után a rendszer viselkedésének leírása egy Kripke struktúra, mely egy $M = (S, \to, \ell)$ hármas, ahol

- S az állapotok (STATES) nemüres halmaza – most nem kötjük ki, hogy véges!
- $\to$ egy totális átmenetreláció, azaz $\to \subseteq S \times S$ úgy, hogy minden s-re van olyan s', hogy $s \to s'$;
- $\ell : S \to P(AP)$ egy címkefüggvény, azt határozza meg, hogy melyik állapotban mely atomi állítások igazak.

A CTL formulái pedig:

- a konstans $\uparrow$ és $\downarrow$ formulák;
- minden $a \in AP$ egyben formula is;
- ha F, G formulák, akkor $\neg F$, $F \vee G$ is azok;
- ha F formula, akkor $\mathrm{EX}(F)$ és $\mathrm{AG}(F)$ is formula;
- ha F, G formulák, akkor $\mathrm{E}(F \,\mathrm{U}\, G)$ (ezt néha $\mathrm{EU}(F, G)$-nek is jelöljük) is formula.

A formulák szemantikája: az $M = (S, \to, \ell)$ rendszer egy s állapotában teljesül az F formula, jelben $M, s \models F$, ha...

- $F = a \in AP$ és $a \in \ell(s)$, vagyis ha s-ben a címkefüggvény szerint teljesül a megadott atomi állítás;
- $F = (G \vee H)$, $F = \neg G$, $F = \uparrow$ és $F = \downarrow$ a szokásos módon;
- $F = \mathrm{EX}(G)$ és s-nek van olyan s' közvetlen rákövetkezője, melyben teljesül G. (EXISTS NEXT) Formálisan, ha $\exists s \to s' : M, s' \models G$.
- $F = \mathrm{EG}(G)$ és létezik egy olyan, az s-ből induló végtelen út, melynek minden állapotában igaz G. (EXISTS GLOBALLY). Formálisan, ha $\exists s = s_0 \to s_1 \to s_2 \to \ldots \forall i : \; M, s_i \models G$.
- $F = \mathrm{E}(G \,\mathrm{U}\, H)$ és létezik olyan, az s-ből egy olyan s' állapotba vezető út, melyre igaz H és az út minden korábbi pontján (s-ben is) igaz G. (EXISTS UNTIL). Formálisan, ha $\exists s = s_0 \to s_1 \to s_2 \to \ldots \exists i : \; M, s_i \models H$ és $\forall j < i : \; M, s_j \models G$.

A CTL modellellenőrző algoritmusa, ha $M = (S, \to, \ell)$ véges, bottom-up módon, alulról felfelé kiszámítja az input F formula minden egyes G részformulájára az összes olyan állapot S_G halmazát, melyekre igaz G, vagyis $S_G = \{s : M, s \models G\}$. Ennek módszere:

- Atomi állítás: $S_a = \{s : a \in \ell(s)\}$.
- Negáció: $S_{\neg F} = S - S_F$.
- Diszjunkció: $S_{F \vee G} = S_F \cup S_G$.
- Exists Next: $S_{\mathrm{EX}(F)} = \{s : \exists s \to s',\ s' \in S_F\}$.
- Exists Globally: $S_{\mathrm{EG}(F)}$ kiszámításához határozzuk meg S_F-et, majd legyen $S_0 = S$ és $S_{i+1} = \{s \in S_i : \exists s \to s', s' \in S_i\}$. Amint $S_i = S_{i+1}$, legyen $S_{\mathrm{EG}(F)} = S_i$.
- Exists Until: $S_{\mathrm{E}(F \,\mathrm{U}\, G)}$ halmazt iteratívan határozzuk meg: $S_0 = S_G$ és $S_{i+1} = S_i \cup \{s \in S_F : \exists s \to s', s' \in S_i\}$. Amint $S_i = S_{i+1}$, legyen $S_{\mathrm{E}(F \,\mathrm{U}\, G)} = S_i$.

Feladatok

15.1. Feladat. Fejezzük ki a CTL következő operátorait a fejezet elején definiáltak segítségével!

a) AX(F): igaz s-ben, ha s minden rákövetkezőjében igaz F (Always Next)

b) EF(F): igaz s-ben, ha elérhető s-ből egy olyan s', melyben igaz F (Exists Finally)

c) AF(F): igaz s-ben, ha tetszőleges, s-ből kiinduló végtelen úton létezik olyan s', melyben igaz s (Always Finally)

d) AG(F): igaz s-ben, ha F igaz tetszőleges, s-ből elérhető állapotban (Always Globally)

e) A($F \,\mathrm{U}\, G$): igaz s-ben, ha tetszőleges, s-ből kiinduló végtelen úton létezik olyan s', melyben igaz G és az összes ezt megelőző állapotban igaz F (Always Until)

15.1. Feladat megoldása.

a) AX(F): igaz s-ben, ha s minden rákövetkezőjében igaz F (Always Next)

$$\neg\mathrm{EX}(\neg F),$$

ennek informális jelentése: „nincs olyan rákövetkezője s-nek, melyben $\neg F$ igaz lenne". Ez pont azt jelenti, amit szeretnénk.

b) EF(F): igaz s-ben, ha elérhető s-ből egy olyan s', melyben igaz F (Exists Finally)

$$\mathrm{E}(\uparrow \mathrm{U}\, F),$$

ennek informális jelentése: létezik olyan s-ből induló végtelen út, melynek egy pontján igaz F, ezt megelőzően pedig mindenhol igaz $\uparrow$. Mivel a mondat második fele automatikusan teljesül, F teljesülése után pedig nem lényeges, hogy hogyan folytatjuk a végtelen utat, ez pont az, amit mondani szeretnénk.

c) AF(F): igaz s-ben, ha tetszőleges, s-ből kiinduló végtelen úton létezik olyan s', melyben igaz F (Always Finally)

$$\neg\mathrm{EG}(\neg F),$$

ennek informális jelentése: nincs olyan végtelen út, melyen F mindvégig hamis – tehát valóban, tetszőleges végtelen úton előbb-utóbb F igaz lesz.

d) AG(F): igaz s-ben, ha F igaz tetszőleges, s-ből elérhető állapotban (Always Globally)

$$\neg\mathrm{EF}(\neg F),$$

ennek informális jelentése: nincs s-ből olyan elérhető állapot, melyben F hamis lenne – vagyis, F igaz tetszőleges, s-ből elérhető állapotban. Az eredeti modalitások alkalmazásával ez a formula $\neg\mathrm{E}(\uparrow \mathrm{U}\, \neg F)$ alakba írható.

e) A(F U G): igaz s-ben, ha tetszőleges, s-ből kiinduló végtelen úton létezik olyan s', melyben igaz G és az összes ezt megelőző állapotban igaz F (Always Until)

A korábbiakhoz képest ez a formula kicsit összetettebb. Lássuk, mikor *hamis* A(F U G)!

- Egyrészt, ha van egy olyan s-ből induló végtelen út, melyen G sose lesz igaz, akkor nyilván hamis.
- Másrészt, ha van egy olyan s-ből induló végtelen út, amin ugyan G igaz valahol, de az első ilyen pontot megelőzően valahol egyszer F is hamissá válik, akkor szintén hamis. Ezt úgy is mondhatjuk, hogy az út elején igaz $\neg G$, aztán hirtelen $\neg G \wedge \neg F$ is igaz lesz.

- Egy végtelen út pont akkor *nem* teljesíti az $F \,\mathrm{U}\, G$ formulát, ha a fenti két pont valamelyikét megsérti.

Így a végső formula:

$$\neg\big(\mathrm{EG}(\neg G) \vee \mathrm{E}(\neg G \,\mathrm{U}\, (\neg G \wedge \neg F))\big).$$

15.2. Feladat. Mit fejeznek ki a következő CTL formulák? Az atomi tulajdonságokat próbáljuk intuitíve, értelem szerint jelentéssel felruházni.

a) AG EFrestart, ahol a reset atomi tulajdonság;

b) AF AGrunning, ahol a running atomi tulajdonság;

c) AG(request $\rightarrow$ AFacknowledge)

d) AG AFenabled

15.2. Feladat megoldása.

a) AG EFrestart

Informálisan: ,,a jövő tetszőleges pontján igaz, hogy eljuthatok restart állapotba" – vagyis, akárhova is viszem el a rendszert, fogom tudni utána restartolni valahogyan

b) AF AGrunning

Informálisan: ,,tetszőleges úton előbb-utóbb eljön egy olyan pont, amikor igaz, hogy onnantól a jövő minden pontjában running igaz lesz" – vagyis, bármit is teszek, a rendszer előbb-utóbb futni fog, és onnan kezdve mindig fut

c) AG(request $\rightarrow$ AFacknowledge)

Informálisan: ,,a jövő tetszőleges pontján igaz, hogy HA request igaz, akkor onnan tetszőleges úton előbb-utóbb eljön egy olyan pont, amikor acknowledge igaz lesz" – vagyis, a jövőben bármikor lesz egy request, azt előbb-utóbb mindenképp követi egy acknowledge

d) AG AFenabled

Informálisan: ,,a jövő tetszőleges pontján igaz, hogy onnan tetszőleges úton előbb-utóbb eljön egy olyan pont, mikor enabled igaz lesz." Ha ezt jobban

meggondoljuk, ez igaz az enabled első beállása utáni állapoton is. Továbbvive ezt a gondolatmenenet azt kapjuk, hogy a formula akkor igaz, ha minden végtelen úton enabled végtelen sokszor igaz, vagyis bármi is történjék a jövőben, a rendszer sosem áll le véglegesen.

15.3. Feladat. Ebben a feladatban olyan CTL modellellenőrző algoritmust fejlesztünk ki, mely lineáris időigényű mind a formula, mind a struktúra méretében, vagyis $O(|\varphi| \cdot |M|)$ időigényű. Modalitásonként tesszük mindezt, vagyis: adjunk olyan módszereket, melyek $O(|M|)$ időben kiszámítják S_F-et,...

a) ...ha $F = \uparrow$ vagy $F = \downarrow$!

b) ...ha $F = \neg G$, feltéve, hogy S_G-t már kiszámítottuk!

c) ...ha $F = G \vee H$, feltéve, hogy S_G-t és S_H-t már kiszámítottuk!

d) ...ha $F = \mathrm{EX}(G)$, feltéve, hogy S_G-t már kiszámítottuk!

e) ...ha $F = \mathrm{EG}(G)$, feltéve, hogy S_G-t már kiszámítottuk!

f) ...ha $F = \mathrm{E}(G \,\mathrm{U}\, H)$, feltéve, hogy S_G-t és S_H-t már kiszámítottuk!

15.3. Feladat megoldása. A megoldásunkban egyrészt feltesszük, hogy a halmazokat bitvektorokkal reprezentáljuk, $O(1)$ időben queryzve membershipet, $O(|M|)$ időben inicializálva üres halmazra, $O(1)$ időben beszúrva elemet. Továbbá, rendelkezésre áll M gráfként és M^T, M transzponáltja szintén gráfként reprezentálva, éllistás reprezentációval, valamint M erősen összefüggő komponenseinek gráfja, így hogy két állapot ugyanabban az erősen összefüggő komponensben van-e, illetve a tartalmazó komponens mérete szintén konstans időben elérhető. Mindezt $O(|M|)$ időben elő tudjuk állítani az S_F halmazok kiszámítása előtt, a formulától ezek az értékek nem függnek.

a) ...ha $F = \uparrow$ vagy $F = \downarrow$!

Adjuk vissza a konstans 1 ill. konstans 0 bitvektorokat, ezeket $O(|M|)$ idő előállítani.

b) ...ha $F = \neg G$, feltéve, hogy S_G-t már kiszámítottuk!

Adjuk vissza S_G komplementerét, ezt $O(|M|)$ idő előállítani.

c) ...ha $F = G \vee H$, feltéve, hogy S_G-t és S_H-t már kiszámítottuk!

Adjuk vissza S_G és S_H koordinátánkénti diszjunkcióját $O(|M|)$ időben.

d) ... ha $F = \mathrm{EX}(G)$, feltéve, hogy S_G-t már kiszámítottuk!

Inicializáljuk S_F-et üres vektorra. Eztán menjünk végig S_G elemein, és minden $s \in S_G$-re szúrjuk be S_F-be az M^T-ben s-sel szomszédos csúcsokat. Így M^T minden élét legfeljebb egyszer járjuk be, tehát ez $O(|M|)$ idő.

e) ... ha $F = \mathrm{EG}(G)$, feltéve, hogy S_G-t már kiszámítottuk!

Állítsuk elő az M gráfjának S_G által feszített részgráfját, vagyis az $\Gamma_G = (S_G, \to \cap S_G^2)$ gráfot, ez $O(|M|)$ időben megy. Azokat a csúcsokat kell beletegyük S_F-be, melyekből ebben a gráfban kiindul végtelen hosszú séta. Ehhez állítsuk elő Γ_G komponensgráfját $O(|M|)$ időben, és a komponensgráfban (ami egy irányított, körmentes gráf) target-source irányú bejárással jelöljünk meg minden komponenst, ami nemtriviális (azaz vagy több csúcs van benne, vagy egy hurokéllel ellátot csúcs), vagy vezet belőle él megjelölt komponensbe. Ez is megy $O(|M|)$ időben. Végül S_F-be tegyük az összes, megjelölt komponens-beli csúcsokat.

f) ... ha $F = \mathrm{E}(G \,\mathrm{U}\, H)$, feltéve, hogy S_G-t és S_H-t már kiszámítottuk!

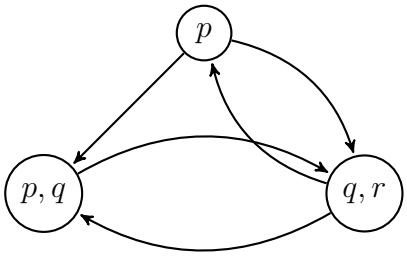
Az előzőhöz hasonló módon, először állítsuk elő az $S_G \cup S_H$ által feszített Γ részgráfját M-nek, $O(|M|)$ időben, valamint ennek a komponensgráfját. Jelöljük meg az összes olyan komponenst, melyben szerepel S_H-beli állapot. (Úgy, hogy S_H-n iterálunk végig és az ő komponenseiket jelöljük meg). Majd a komponensgráfon target-source irányú bejárással jelöljük meg még minden komponenst, melyből vezet él megjelölt komponensbe. Végezetül, S_F-be tegyük be az összes, megjelölt komponens-beli csúcsokat.

15.4. Feladat. Mutassuk meg, hogy CTL-re *nem* igaz a kompaktsági tétel.

15.4. Feladat megoldása. Legyen p egy atomi állítás, F_n pedig az $\mathrm{AX}\,\mathrm{AX}\,\ldots\,\mathrm{AX}p$ formula, ahol az AX modalitás n-szer szerepel. Tehát F_n akkor igaz s-ben, ha pontosan n lépésben csupa olyan állapot érhető el benne, melyekben igaz p.
Akkor a $\Sigma = \{F_n : n \geq 0\} \cup \{\mathrm{EF}\neg p\}$ formulahalmaz kielégíthetetlen, hisz azt mondja, hogy nulla, egy, kettő, ... lépésben csupa olyan állapot érhető el s-ből, melyekben igaz p, és még hozzávesszük, hogy emellett elérhető s-ből valahány lépésben egy olyan állapot is, melyben p hamis – ez nyilván egyszerre nem teljesülhet.
Ugyanakkor, Σ tetszőleges véges Σ_0 részhalmazát ha vesszük, abban az F_n formulákból csak véges sok van, tehát van olyan N, melyre F_N nem szerepel Σ_0-ban. Akkor a $0 \to 1 \to 2 \to \ldots$ Kripke struktúrában, ahol $p \notin \ell(N)$ és $p \in \ell(n)$ minden

$n < N$-re, Σ_0 teljesül az $s = 0$ állapotban. Tehát Σ minden véges részhalmaza kielégíthető, Σ mégsem az.

15.5. Feladat. Tekintsük a következő Kripke struktúrát:



Mely állapotaiban igaz...

a) p?

b) $p \to q$?

c) $\mathrm{EG}(p)$?

d) $\mathrm{EG}(q)$?

e) $\mathrm{AG}(p)$?

f) $\mathrm{AF}(p \to \mathrm{EG}q)$?

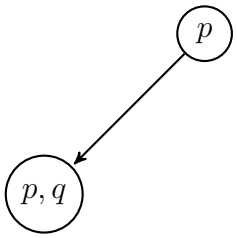
g) $\mathrm{E}(p \,\mathrm{U}\, q)$?

15.5. Feladat megoldása. Jelölje a felső állapotot s_0, a bal alsót s_1, a jobb alsót s_2.

a) $S_p = \{s_0, s_1\}$, az állapotokba beleírt címke függvény alapján.

b) $S_{p \to q} = (S - S_p) \cup S_q = (\{s_0, s_1, s_2\} - \{s_0, s_1\}) \cup \{s_1, s_2\} = \{s_1, s_2\}$.

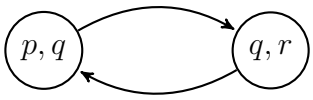
c) $S_{\mathrm{EG}(p)}$ meghatározásához meghatározzuk az $S_p = \{s_0, s_1\}$ által feszített részgráfot:



Ennek komponensgráfja önmaga, és előbb s_1-et, majd s_0-t bejárva nem címkézünk meg senkit, mert akkor tennénk, ha nemtriviális komponensben lenne vagy mutatna belőle él címkézett csúcsba. Tehát $S_{\mathrm{EG}(p)} = \emptyset$.

A fixpontiterációs módszerrel: $S_0 = S_p = \{s_0, s_1\}$, $S_1 = \{s \in \{s_0, s_1\} : \exists s' \in \{s_0, s_1\} s \to s'\} = \{s_0\}$, mert s_0 és s_1 közül csak s_0-ból megy él $\{s_0, s_1\}$-be, majd $S_2 = \emptyset$, mert s_0-ból sem megy él $\{s_0\}$-ba, végül $S_3 = \emptyset$ és így $S_{\mathrm{EG}(p)} = \emptyset$.

d) EG(q)-hoz meghatározzuk az $S_q = \{s_1, s_2\}$ által feszített részgráfot:



Ennek egyetlen komponense a $\{s_1, s_2\}$, mely nemtriviális, hisz van benne él. Megjelöljük ezt a komponenst, több komponens nincs, és visszaadjuk az összes $\{s_1, s_2\}$-beli csúcsot: $S_{\mathrm{EG}q} = \{s_1, s_2\}$.

e) AG(p)? Azon csúcsokat kell visszaadjuk, melyekből minden elérhető állapotban igaz p. Ilyen csúcs nincs: $S_{\mathrm{AG}(p)} = \emptyset$.

Módszeresen a $\neg\mathrm{EF}\neg p$ formulát kifejtve $S_{\neg p} = \{s_2\}$, $S_{\mathrm{EF}\neg p} = \{s_0, s_1, s_2\}$ (a transzponált gráfban egy elérhetőség $S_{\neg p}$-ből, majd $S_{\neg\mathrm{EF}\neg p} = S - \{s_0, s_1, s_2\} = \emptyset$.

f) AF($p \to \mathrm{EG}q$)-hoz $S_p = \{s_0, s_1\}$, $S_{\mathrm{EG}q} = \{s_1, s_2\}$, $S_{p\to\mathrm{EG}(q)} = \{s_1, s_2\}$ és $S_{\mathrm{AF}(p\to\mathrm{EG}q)}$-ben azok a csúcsok vannak benne, akikből bármelyik úton is megyek, előbb-utóbb $\{s_1, s_2\}$-be jutok: mind, S.

g) E($p\,\mathrm{U}\,q$)-hez $S_q = \{s_1, s_2\}$, $S_p = \{s_0, s_1\}$, így az $S_p \cup S_q$ által feszített részgráf maga a struktúra, ennek transzponáltjában kell S_q-ból egy elérhetőséget kiszámítani, ami $\{s_0, s_1, s_2\}$ lesz.