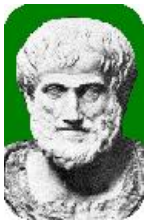
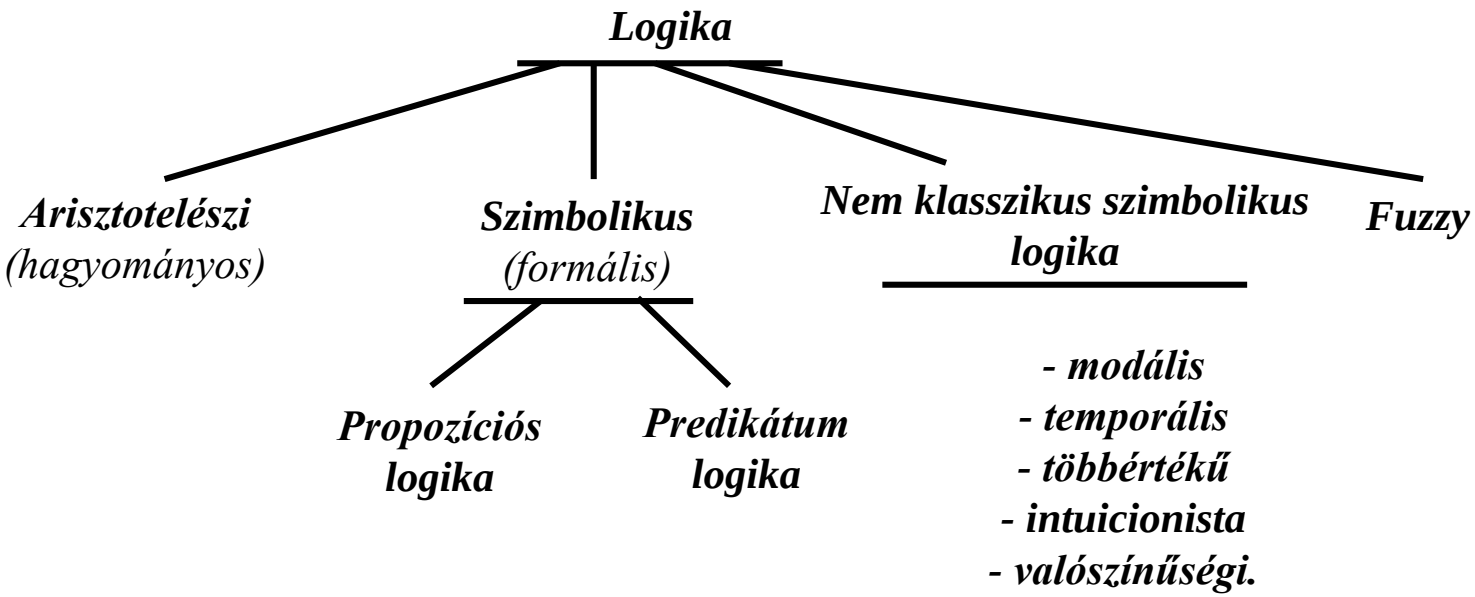
Tudásszemléltetés formális logikával

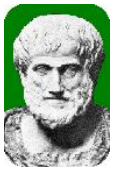


- **A logikáról**

  A logika a bölcselés tudománya, a helyes gondolkodás művészetének tana.

  - **A logika osztályozása**





A szimbolikus logika kifejlődése

- **Petrus Ramus** (1515-1572)
  - Bírálta az arisztotelészi logikát.
  - "A logika: a helyes érvelés eszköze."

1.

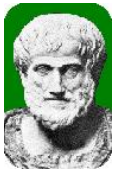
- **Francis Bacon** (1561-1626)
  - "A logika az új ismeretek megszerzésének eszköze."
  - Kísérletek, mérések fontossága.

2.

- **René Descartes** (1596-1650)
  - *"Értekezés a módszerről"* c. műve
    - A problémákat osszuk független, vagy kevéssé összefüggő részekre
    - Az egyszerűtől haladjunk a bonyolultabb felé
    - Csak olyan dolgokat fogadjunk el igaznak, melyek igazsága megkérdőjelezhetetlen
    - Törekedjünk a teljes felsorolásokra.

3.

1. www.phil-fak.uni-duesseldorf.de/ .../neuzeit/ramus.htm 2. "**www.amorc.no/images/bacon.jpg**"
3. www.homeoint.org/seror/ articles/raison.htm



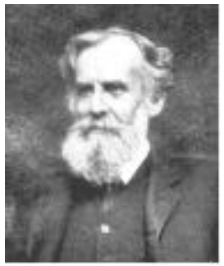
- **A szimbolikus logika kifejlődése ..**

1.

  - **Gottfried Wilhelm Leibniz (1646-1716)**
    - A szimbolikus logika előfutára.
    - "Logika $\cong$ matematika."
    - **Azonosság** definíciója**:**
      "Azok a terminusok tekintendők azonosnak, vagy egybeesőnek, amelyek egymással tetszés szerint felcserélhetők anélkül, hogy ezáltal bármelyik állítás igazsága megváltozna."

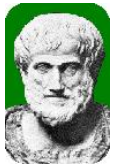
2.

  - **George Boole** (1815-1864)
    - A szimbolikus logika megalapítója.
    - *"A logika matematikai analízise"* című műve.
    - A **logikai szorzás** és **logikai összeadás** műveletének bevezetése.


3.

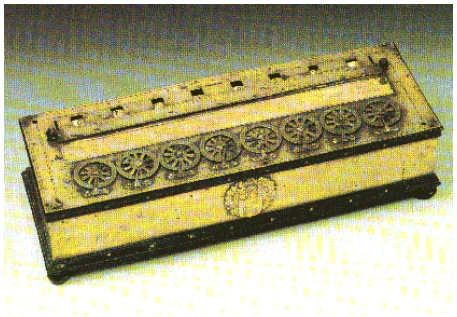
  - **John Venn** (1834-1923)
    - *"Symbolic Logic"* című műve
    - Diagramok: teljes esemény = téglalap; körök: mindenféle egymásbametszés

1. http://usuarios.lycos.es/Cantemar/Leibniz.html 2. powerreporting.com/ altavista.htm
3. lawrence.lancour.students.noctrl.edu/ mathematicians.html



A szimbolikus logika kifejlődése ..

- **William S. Jevons** (1835-1882)
  - *"megengedő vagy"*
  - Első logikai gép (a logikai zongora)



1.

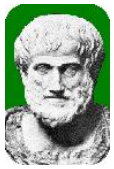
- **Charles S. Peirce** (1839-1914)
  - Boole összes logikai művelete kiváltható *egyetlen művelettel*, a ‘sem...., sem…‘, NOR művelettel.

2.

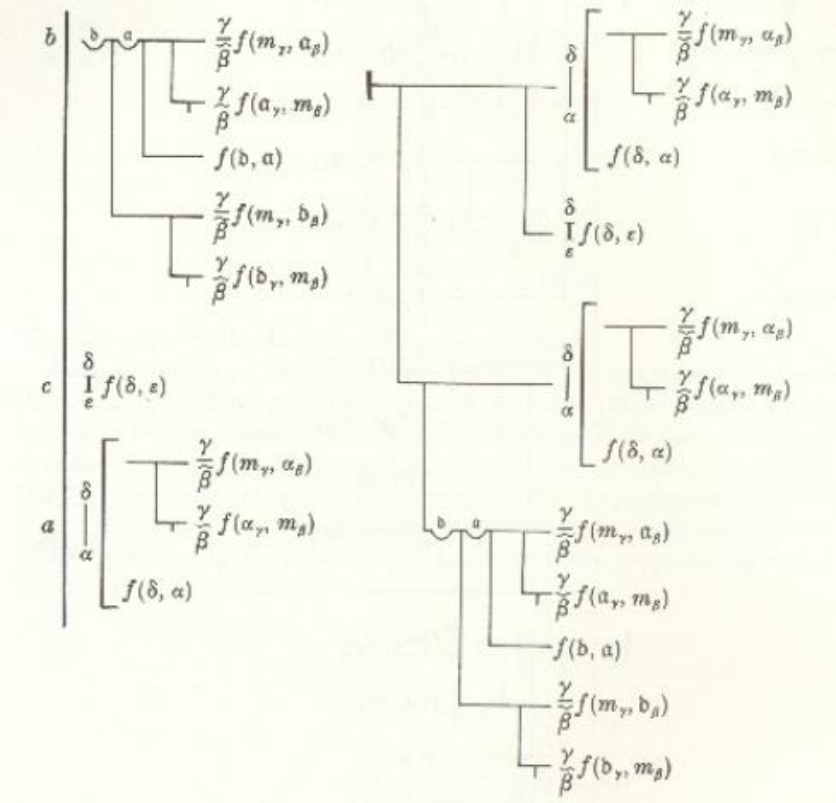
- **Augustus de Morgan** (1806-1871)
  - Reláció-azonosságok.

3.

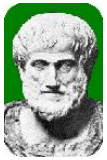
*1. "**pierre.mf.free.fr/tpe/ partie2/pasclin2.gif**" 2. www.macalester.edu/~warren/ courses/P50-01_timeln.htm*
3. www.din.uem.br/ia/precursores/ morgan.html



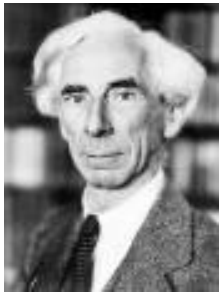
- **A szimbolikus logika kifejlődése ..**
  - **Gottlob Frege** (1848-1925)
    - Kimagasló teljesítménye
    - ***"A szimbolikus logika atyja"***
    - *"Fogalomírás, a tiszta gondolkodás aritmetika mintájára megalkotott formanyelve" c. műve*



Részlet a Begriffschrift-ből

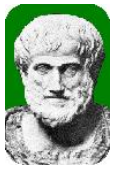


- **A szimbolikus logika kifejlődése ..**
  - **Bertrand Russell** (1872-1970)
    - Frege-től függetlenül ugyanoda eljutott
    - Hézagok az elődök munkáiban: logikai **paradoxonok**.
    - *"Az osztályok osztálya eleme-e önmagának ?"*
    - **Típuselmélete**: magyarázat a paradoxonokra.



1.

1. www.univie.ac.at/bvi/photo-gallery/ photo_gallery.htm



Új irányzatok a logikában

- *A **modális** logikákkal* olyan terminusok is kifejezhetők, mint a "lehetséges", "szükségszerű" és ezek változatai. A modális logika a klasszikus logika kiterjesztése.
- *A **deontikus** logika* olyan etikai és jogi kifejezések elemzésével foglalkozik, mint a "Megtiltott, hogy...", "Megengedett, hogy...", stb.

- **Jan Lukasiewicz** (1878-1956)
  - ***Háromértékű logika:* 0, 1/2, 1**
  - Az 1/2 érték a bizonytalanságot fejezi ki.
  - Eljutott *a végtelenértékű logikáig*.

1.

- **E.L.Post** (1897-1954)
  - ***Többértékű*** logika, Lukasievicz-csel egyidőben.

- ***J.M.Keynes*** (1883-1946)
  - *Ch.S.Peirce-szel együtt a **valószínűségi** logika* megalkotójának tartják.
  - Logikai valószínűség: a racionális hit (ésszerű meggyőződés) foka, 0...1 közötti érték.

2.

1. www.fmag.unict.it/PolPhil/ Lukas/Lukas.html 2. www.ssc.upenn.edu/~ues/

Propozíciós logika

- Egyéb elnevezések: kijelentéskalkulus, ítéletkalkulus, nulladrendű predikátum-kalkulus.

- *Propozíció*: egy kijelentő mondat formában megadott állítás, mely az adott kontextusban egyértelmű *igaz*, vagy *hamis* logikai értékkel bír.

- A propozíciós logika olyan egyszerű nyelvtani kapcsolatokat alkalmaz a legegyszerűbb "atomi mondatok" között, mint az ***és***, ***vagy***, ***nem***.

- *Atomi mondatok* :
  A= " A víz száz fokon forr."
  B= " A macskák ugatnak."

  Az ilyen atomi összetevők (objektumok) **egyértelműen megítélhetők** az igaz, vagy hamis értékek valamelyikével. (Általános tapasztalat szerint A igaz, B hamis).

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

Propozíciós logika ..

- ***A propozíciós logika tárgya, célja*** összetett szerkezetek kiértékelésére formális szabályt adni. A kiértékelésben az atomi összetevők (objektumok) igazságértékeit kell csak figyelembe venni, jelentésüktől eltekintünk.

- Atomi kijelentések helyett szimbólumok:

  Esik az eső. $\Rightarrow$ A

- Eltekintünk a természetes nyelvi jelentéstől
  Nem természetes átalakítás, pl.:
  A: Jóska megbetegedett.
  B: Jóska elment az orvoshoz.

  Hétköznapi értelemben **A és B** jelentése eltér **B és A** összetett kijelentésétől.

  Csak az igaz, vagy hamis értéke fontos számunkra A-nak és B-nek.

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

A propozíciós logika szintaxisa

- **Jelkészlet**

- **Elválasztójelek:** , ( ) [ ]

- **Logikai operátorok** (műveleti jelek): {csökkenő precedenciasorrendben}

¬	∧	∨	→	≡
negáció	konjunkció	diszjunkció	implikáció	ekvivalencia
'nem'	'és'	'vagy'	'ha ... akkor ...' 'következik'	'akkor és csak akkor' 'azonos'

- **Logikai változók** (ítéletváltozók): ***p, q, r, …***

- **Logikai konstansok** (ítélet-, predikátumkonstansok) : ***T, F (True, False), (igaz, hamis)*.**

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

A propozíciós logika szintaxisa ..

- A propozíciós logika (kijelentéskalkulus, ítéletkalkulus) ***formuláit*** a jelkészlet elemeiből a ***szintaxis szabályai*** szerint építjük fel:

- *Atomi formula* (atom)
  - Minden ítéletkonstans atom : **T, F**
  - Minden ítéletváltozó atom : **p, q, r, …**

- *Formula* (jól formált formula):
  - Minden atomi formula egyben formula
  - Ha **A** és **B** formulák, akkor a

$$\neg A,\ \ A \wedge B,\ \ A \vee B,\ \ A \rightarrow B,\ \ A \equiv B$$

kifejezések is formulák.

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

A propozíciós logika szemantikája

- Egy logikai formulának az igazságértéke ad ***jelentést***, mely igazságértéket a *szemantika szabályai* szerint kapja meg.

- **A jelentésadás lépései:**
  1. Interpretálás
  2. Kiértékelés

- **Interpretálás**
  A formula *interpretálása:* minden egyes ítéletváltozójához az *igaz*, vagy a *hamis* értéket rendeljük *minden lehetséges módon*. Egy hozzárendelés egy interpretációt ad.
  Az interpretációk száma a propozíciós logikában véges: 2^n ahol n a változók száma.

  pl.: $A{:=}(p \wedge q) \rightarrow (r \equiv (\neg s))$

  két lehetséges interpretációja:

  I_1: $(p,q,r,s) \leftarrow (T,T,F,F)$

  I_2: $(p,q,r,s) \leftarrow (F,T,T,F)$

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

A propozíciós logika szemantikája ..

- **Kiértékelés**
  Az interpretált formula kiértékelését a logikai **műveleti jelek szemantikája alapján** végezzük:

A	B	$\neg A$	$A \wedge B$	$A \vee B$	$A \rightarrow B$	$A \equiv B$
T	T	F	T	T	T	T
T	F	F	F	T	F	F
F	T	T	F	T	T	F
F	F	T	F	F	T	T

Pl.: Ha süt a nap, nem tanul Péter.

süt a nap : p

tanul Péter : q

$A := p \rightarrow (\neg q)$

A lehetséges 4 interpretáció egyike: $p = T$, $q = F$,
a kiértékelt formula: $T \rightarrow (\neg F)$; $T \rightarrow T$; T, az A állítás igaz.

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

Kielégíthetetlen és érvényes kifejezések

A kielégíthetetlen logikai kifejezések és az érvényes kifejezések fontos szerepet játszanak a **tételbizonyításban**. A **tétel** azt jelenti, hogy állításokból logikailag következik a konklúzió.

- **Tautologikus törvény:** $A \vee B = A$, ha A tautológia,
  $A \wedge B = B$, ha A tautológia.

  A logikai kifejezés tautológia, azaz **érvényes** kifejezés akkor, ha **minden lehetséges helyettesítése igaz.**

- **Kielégíthetetlenségi törvény:** $A \vee B = B$, ha A kielégíthetetlen,
  $A \wedge B = A$, ha A kielégíthetetlen.

  A logikai kifejezés **kontradikció**, azaz kielégíthetetlen kifejezés akkor, ha minden lehetséges helyettesítésre hamis értéket ad.

Természetesen a formulák többsége egyes interpretációkban hamis, más interpretációkban igaz, azaz kielégíthető.

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

Tételbizonyítás a propozíciós logikában

- Egy propozíciós logikai formula **érvényességének igazolására** alkalmas módszerek:
  - Igazságtáblás
  - Formális levezetés
  - Quine algoritmusa
  - Wang algoritmusa
  - Rezolúció.

- A rezolúció szabálya:

$$\frac{\alpha \vee \beta,\ \neg\beta \vee \gamma}{\alpha \vee \gamma}$$

  Mivel β nem lehet egyszerre igaz és hamis, ezért valamelyik premisszában, előfeltételben a másik tagnak (α, vagy γ) igaznak kell lennie, tehát az $\alpha \vee \gamma$ igaz.

- A rezolúció módszere kiemelkedik a többi közül, mivel ez a bizonyítási módszer alkalmazható a predikátum logikában is.

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

Tételbizonyítás a propozíciós logikában ..

- A **tétel** igaz állításokból és egy, azokból állítólag következő konklúzióból áll.

- A **tételbizonyítás** feladata formális szabályok gépies alkalmazásán keresztül igazolni, hogy a konklúzió a premisszák következménye.

- A rezolúciós módszer **lépései**:
  - A bizonyítandó tétel **tagadott alakját** vesszük alapul azáltal, hogy feltesszük, hogy a konklúzió ellentettje igaz a premisszák igaz értékei mellett. Ha az ily módon kapott összetett formula kielégíthetetlen voltát sikerül belátni, akkor az eredeti tétel igaz.
  - Ehhez konjunktív normál forma alakra (KNF) hozzuk az összetett formulát és a **rezolúció ismételt alkalmazásával**, fokozatos egyszerűsítéseken keresztül **jutunk el az ellentmondáshoz**.

- A rezolúció alkalmazása a propozíciós logikában (ítéletkalkulus) korlátozott, mivel bonyolultabb problémák leírására ez a logika nem ad eszközöket.

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

Tételbizonyítás a propozíciós logikában ..

- A konjunktív normálforma
  - Klózok konjunkciója, ÉS kapcsolata
  - Az eredeti formulával ekvivalens
  - Egyszerű felépítése miatt gépiesen kezelhető.

- A klóz literálok diszjunkciója (VAGY kapcsolata), vagy egyetlen literál.
- A literál egy ítéletváltozó (logikai változó), vagy annak negáltja.

Példa:

$$\overbrace{\underbrace{(p \vee \neg q)}_{\text{klóz}} \wedge \underbrace{(\neg p \vee r)}_{\text{klóz}} \wedge \underbrace{(\neg r \vee \neg s)}_{\text{klóz}} \wedge \underbrace{q}_{\text{klóz}} \wedge \underbrace{s}_{\text{klóz}}}^{\textit{konjunktív normál forma}}$$

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

Tételbizonyítás a propozíciós logikában ..

- Formulák konjunktív normálformára hozásának lépései:
  1. Azonosságok kiküszöbölése: $A \equiv B = (A \rightarrow B) \wedge (B \rightarrow A)$
  2. Implikációk kiküszöbölése: $A \rightarrow B = \neg A \vee B$
  3. A negálás hatáskörének redukálása: $\neg (A \vee B) = \neg A \wedge \neg B$
     $\neg (A \wedge B) = \neg A \vee \neg B$
  4. Klózok konjunkciójának létrehozása: $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$.

- A diszjunktív normál formából elhagyva a konjunkció operátorokat, a formula klózokra esik szét, melyek halmazára alkalmazzuk a rezolúció szabályát ciklusban. Ily módon a rezolúció egy olyan könnyen automatizálható eljárás, melynek segítségével egy klózhalmaz, illetve a neki megfelelő konjunktív normál forma kielégíthetetlenségét belátjuk.

- A klózhalmaz egyszerűsítése rezolválható klózpárokon keresztül történik.

- Rezolválható klózpár: egyetlen ellentett literálpárt tartalmaznak, pl:
  $$p \vee A, \quad \neg p \vee B$$
  ahol A és B közvetlenül tovább már nem rezolválható klózok.

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

Tételbizonyítás a propozíciós logikában ..

- A rezolúciós folyamat során a rezolválható párok rezolválása eredményeként adódó rezolvens klózt a halmazhoz adjuk.
- A klózhalmaz kielégíthetetlenségét az jelzi, hogy végül üres klózt kapunk.
- A rezolúció algoritmusa:

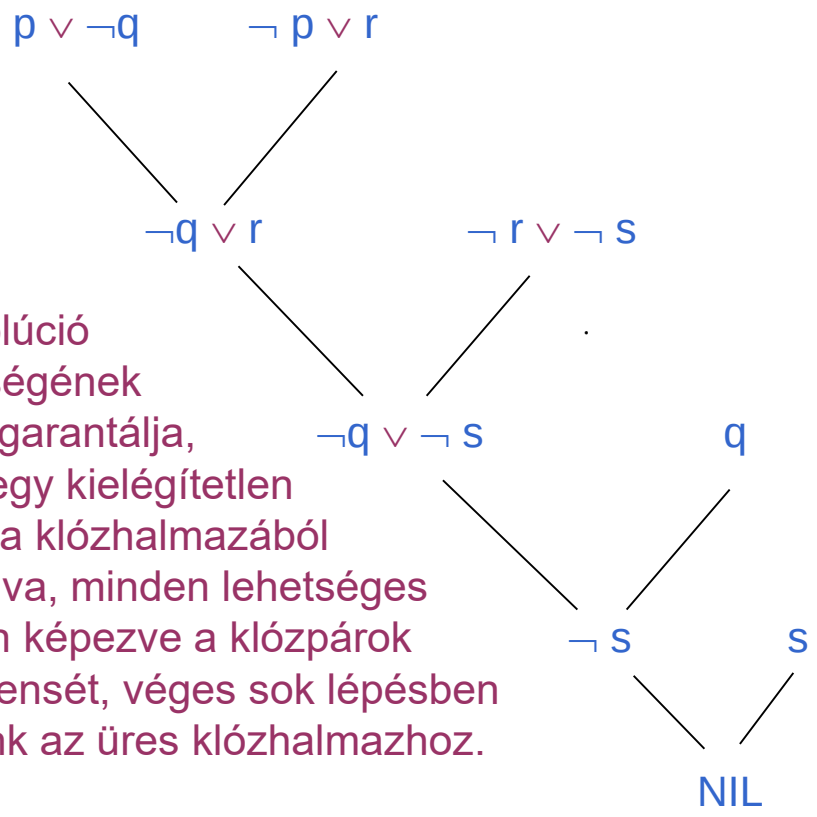
Procedure Rezolúció
1. KLÓZHALMAZ feltöltése
2. do
3. Rezolválható klózpár Ci, Cj kiválasztása a KLÓZHALMAZ-ból
4. Rezolválás: Cij $\Leftarrow$ R(Ci,Cj)
5. KLÓZHALMAZ bővítése Cij rezolvenssel
6. while Cij $\neq$ NIL
end

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

- **Tételbizonyítás a propozíciós logikában ..**

- Példa:

KNF: $(p \vee \neg q) \wedge (\neg p \vee r) \wedge (\neg r \vee \neg s) \wedge q \wedge s$



A rezolúció teljességének tétele garantálja, hogy egy kielégítetlen formula klózhalmazából kiindulva, minden lehetséges módon képezve a klózpárok rezolvensét, véges sok lépésben eljutunk az üres klózhalmazhoz.

*Forrás: Fekete I.-Gregorics T.-Nagy S.: **Bevezetés a mesterséges intelligenciába** LSI Oktatóközpont, Budapest, 1990. p290.*

Predikátum logika

További elnevezések: elsőrendű predikátumkalkulus, vagy egyszerűen predikátumkalkulus.

- *Predikátum:* objektumok sajátságainak és az objektumok közötti kapcsolatok, viszonyok, relációk szimbolikus formában való megadására alkalmas állítás, mely egy adott interpretációban egyértelmű igaz, vagy hamis minősítéssel bír.

- **A predikátum logika jellemzői:**
  - MI feladatok reprezentálására alkalmas (ld.: PROLOG).
  - *Kijelentéseink **tartalmát** is leírja*, mivel az állítások az individuumkonstansok, individuumváltozók és individuumfüggvények révén egy tartomány (, alaphalmaz, domain) elemeire vonatkozhatnak.
  - Az ítélet-, vagy logikai függvények, azaz a ***predikátumok*** használatával indivíduum paraméterektől függő igazságú állítások fogalmazhatók meg.
  - Új logikai operátorok, a **kvantorok** segítségével kifejezhetjük a *"minden"* és a *"létezik"* fordulatokat is.
  - Az *interpretáció módja eltér* a propozíciós logikáétól, és ebből eredően a predikátum logikában egy formulának **végtelen sok interpretációja** lehetséges az alaphalmaz, és az azon értelmezett függvények és predikátumok tetszőleges választhatósága miatt.

- **Predikátum logika ..**

Bonyolultabb tételek bizonyíthatók, mint a propozíciós logikában:

Pl.:

Premisszák, ismert tények:

(1) Van olyan hallgató, aki minden tárgyat szeret.

(2) A szócséplést egyik hallgató sem szereti.

Igaz-e a fenti állítások ismeretében a következő állítás?

Bizonyítandó állítás:

(4) Egyik tárgy sem szócséplés.

A predikátum logika szintaxisa

- ***Jelkészlet:***
  - **Elválasztójelek:** , ( ) [ ]
  - **Logikai operátorok** (műveleti jelek): $\neg \wedge \vee \rightarrow \equiv$
  - **Kvantorok:** univerzális kvantor ("minden"): $\forall$
    egzisztenciális kvantor ("létezik" vagy "van olyan"): $\exists$.
    (precedencia: $\forall \exists \neg \wedge \vee \rightarrow \equiv$)
  - **Elemkonstansok** (individuumkonstansok): *a, b, c,...*,vagy nevek (például: *János, asztal*).
  - **Elemváltozók** (individuum változók): *x, y, z...*
  - **Függvényszimbólumok** (individuumfüggvények): *f, g, h,...* , vagy
    például: *testvére(Józsi)* $\Rightarrow$ *Laci*
  - **Itéletkonstansok** (logikai konstansok): *True, False*.
  - **Itéletváltozók** (logikai változók): *p, q, r...*
  - **Predikátumszimbólumok** (logikai függvények): *P, Q, R,...* , vagy
    például: *testvérek(x, y)*,
    P(x,y).

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

- **A predikátum logika szintaxisa ..**

  - **A jelkészlet elemeiből a *kifejezések* 3 osztálya alkotható meg:** Term-ek, Atomi formulák (Atomok) és Jólformált formulák (Formulák).

  - ***Term*** (individuum konstans, individuum változó, individuum függvény):

    - Minden elemkonstans (individuum konstans) term. Pl.: *a,b,...,asztal, szék*, stb.

    - Minden elemváltozó (individuum változó) term. Pl.: *x, y*.

    - Ha *f* egy *n*-argumentumú függvényszimbólum és $t_1,...,t_n$ term-ek, akkor $f(t_1,...,t_n)$ is term. Pl.: *f(a,x)*, *gyereke(Pál,x)*,
      színe(haj,időskorban)$\Rightarrow$*ősz*,
      színe(haj,fiatalkorban) $\Rightarrow$ *szőke*.

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

- **A predikátum logika szintaxisa ..**

- ***Atomi formula*** (kiértékelve *True*, vagy *False*):
  - **Minden ítéletkonstans atomi formula.** (*True, False*)
  - **Minden ítéletváltozó atomi formula.** (*p, q, r,…*)
  - **Ha *P egy n*-argumentumú predikátumszimbólum (logikai függvény) és $t_1,...,t_n$ term-ek, akkor $P(t_1, ...,t_n)$ atomi formula.**
    Pl.: *házaspár(Pál,Irén)* $\Rightarrow$ *True*.

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

A predikátum logika szintaxisa ..

- ***Formula*** (jólformált formula):
  - **Minden atomi formula egyben formula.** (*True, p, P(a,b)*)
  - **Ha *A* és *B* formulák, akkor a**
    (¬A), (A ∧ B), (A ∨ B), (A → B), (A ≡ B) **kifejezések is formulák.**
  - **Ha az *A* egy formula és az *x* egy elem (individuumváltozó), akkor a**
    (∀x)A, (∃x)A **kifejezések is formulák.** Pl.: *(∀x) testvére (x,Pál)*

    Precedenciasorrend: ∀∃¬ ∧ ∨ → ≡, pl.: (((∃x)A) ∨ (B) helyett: (∃x)A ∨ B

    Példa: Vagy van egy részeg, vagy nem volt már ital.
    A:= részeg(x)=R(x),
    C= volt ital..
    (∃x)R(x) ∨ ¬C.

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

A predikátum logika szintaxisa ..

- **Kötött változó:**
  Minden előfordulása a formulában kvantor hatása alatt áll.

  $(\forall x)\ P(x,y)$ x kötött változó, y nem kötött változó.

  $(\forall x)\ [P(x,y) \wedge (\exists y)\ Q(x,y)]$ mindkét x kötött *[]* miatt, y első előfordulása nem kötött, második előfordulása kötött.

- **Mondat:** olyan formula, melyben minden változó összes előfordulása kötött. Csak mondatokkal foglalkozunk.

  Példa mondatra: $(\forall x)\ [P(x) \vee (\exists y)\ Q(x,y)]$

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

Példa a formulázásra

Feladat:

Állítások: **(1) Van olyan hallgató, aki minden tárgyat szeret.**

(2) A szócséplést egyik hallgató sem szereti.

---

(4) Egyik tárgy sem szócséplés

Formalizálás:

Alkalmazott predikátumok:

H(x): *az x egy hallgató*

T(y): *az y egy tárgy*

C(y): *az y egy szócséplés*

S(x,y): *az x szereti y-t.*

A formalizált állítások:

F1: $(\exists x)[H(x) \wedge (\forall y)(T(y) \rightarrow S(x,y))]$

F2: $(\forall x)[H(x) \rightarrow (\forall y)(C(y) \rightarrow \neg S(x,y))]$

F3: $(\forall x)[T(x) \rightarrow \neg C(x)]$

dr.Dudás László

A predikátumkalkulus szemantikája

- Az elsőrendű predikátumkalkulus formuláinak is hasonlóan adunk **jelentést**, mint az ítélet-kalkulusban:
  - először ***interpretáljuk*,** majd
  - ***kiértékeljük*** a formulát. (Eredmény: True, vagy False).

- ***Interpretáció:***
  - ***Individuumtartomány*** *(*az értelmezés alaphalmazának) **megválasztása:** jele *D*, nem üres.
  - **Hozzárendelések:**
    - Minden *individuumkonstans-szimbólumnak* feleltessük meg *D* egy elemét. (*a:= asztal*)
    - Minden *n*-argumentumú *individuumfüggvényhez* (pl.: *f(a,b)*) rendeljünk hozzá egy $D^n \Rightarrow D$ leképezést. (*gyermeke(a,b)* $\Rightarrow$ *c, gyermeke(Pál,Irén)* $\Rightarrow$ *Jutka*).
    - Minden *n*-argumentumú *predikátumfüggvénynek* feleltessünk meg egy $D^2 \Rightarrow \{T,F\}$ leképezést, ahol *T* a logikai igaz, *F* pedig a logikai hamis értéket jelöli. (Pl.: *házaspár(Pál,Irén)* $\Rightarrow$ *True*).

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

- **A predikátumkalkulus szemantikája ..**

  - ***Kiértékelés:***

    - Ha az *A* és *B* formulák igazságértéke ismert, akkor a ***(¬A), (A ∧ B), (A ∨ B), (A → B), (A ≡ B)*** formulák *igazságértékét* a logikai műveleti jelek propozíciós logikánál megadott igazságtáblái alapján *határozzuk meg*.

    - A ***(∀x)A*** formula igazságértéke pontosan akkor *T* az adott interpretációban, ha az *A* formula igazságértéke *minden x ∈ D* esetén *True*, egyébként *(∀x)A* értéke *False*.

    - A ***(∃x)A*** formula igazságértéke pontosan akkor *True* az adott intrepretációban, ha az *A* formula igazságértéke *legalább egy x ∈ D* esetén *True*; egyébként *(∃x)A* értéke *False*.

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

Predikátum-logikai formulák kielégíthetősége

- ***Kielégíthető*** az a formula, amelyik valamely interpretációban igaz.
- ***Érvényesnek*** nevezünk egy formulát, ha minden interpretációban igaz.
  Például a $(\forall x)P(x) \vee (\exists y) \neg P(y)$.
- ***Kielégíthetetlen*** *az a* formula, amely minden interpretációban hamis értékű.
  Például a $(\forall x)P(x) \wedge (\exists y) \neg P(y)$.
- **Az *érvényes*, illetve a *kielégíthetetlen* formulák a fontosak** számunkra, ugyanis a tételbizonyítást egy formula érvényességének, illetve kielégíthetetlenségének a belátására lehet visszavezetni.
- **Probléma:** az elsőrendű predikátum kalkulus egy formuláját *nem tudjuk az összes lehetséges interpretációban kiértékelni*, mivel végtelen sok alaphalmazt lehet megadni.
- **Megoldás***: a rezolúciós eljárás, mellyel egy formula* érvényessége, vagy kielégíthetetlensége belátható.

Forrás: Fekete I.-Gregorics T.-Nagy S.: ***Bevezetés a mesterséges intelligenciába*** *LSI Oktatóközpont, Budapest, 1990. p290.*

Rezolúció a predikátum logikában

Bonyolultabb tételek bizonyíthatók, mint a propozíciós logikában, ugyanakkor a bizonyítás lépései is jóval összetettebbek.
A következőkben egy egyszerű példát mutatunk, amely rámutat a predikátum logikán belüli rezolúció egyik erősségére: A bizonyítás folyamán individuumváltozó - individuumkonstans hozzárendeléseket is végez, amely révén arra is választ kapunk, hogy milyen feltételek mellett igaz a tétel.

Igazi erejét megtapasztalhatjuk a Prolog programozási nyelvben.

Pl.: Feladat: Értik-e a hallgatók a rezolúciót?

Premisszák, ismert tények:

(1) A hallgatók értelmesek.

(2) Az értelmesek mindent értenek, ami egyszerű.

(3) Az értelmeseknek a rezolúció egyszerű.

Igaz-e a fenti állítások ismeretében a következő állítás?

Bizonyítandó állítás:

(4) A hallgatók értik a rezolúciót.

Rezolúció a predikátum logikában ..

Formulázás:

- Alkalmazott predikátumok:

  hallgató(x) - az x az egy hallgató

  értelmes(x) - az x értelmes

  érti(x,y) - az x érti az y-t

  egyszerű(y) - az y egyszerű

- A premisszák és a negált következmény formalizált alakban:

  (1) $\neg$ hallgató(x) $\vee$ értelmes(x)

  (2) $\neg$ értelmes(x) $\vee$ érti(x,y) $\vee$ $\neg$egyszerű(y)

  (3) $\neg$ értelmes(x) $\vee$ egyszerű (Rezolúció)

  (4) hallgató(x) $\wedge$ $\neg$ érti(x,Rezolúció) $\Leftarrow$ tagadása $\Leftarrow$

  $\neg$ hallgató(x) $\vee$ érti(x,Rezolúció)

dr.Dudás László

Rezolúció a predikátum logikában ..

A klózhalmaz:

¬ hallgató(x) ∨ értelmes(x) ¬ értelmes(x) ∨ érti(x,y) ∨ ¬egyszerű(y)

¬ értelmes(x) ∨ egyszerű (Rezolúció) hallgató(x) ¬ érti(x,Rezolúció)

A rezolúciós folyamat lépései:

¬ hallgató(x) ∨ értelmes(x) hallgató(x)

értelmes(x) ¬ értelmes(x) ∨ érti(x,y) ∨ ¬egyszerű(y)

¬ érti(x,Rezolúció) érti(x,y) ∨ ¬egyszerű(y)

↓ y ⇐ Rezolúció

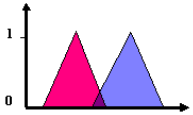
¬egyszerű(Rezolúció) ¬ értelmes(x) ∨ egyszerű (Rezolúció)

¬ hallgató(x) ∨ értelmes(x) ¬ értelmes(x)

¬ hallgató(x) hallgató(x)

NIL

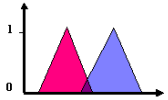
dr.Dudás László



A Fuzzy logika alapjai

A Fuzzy logika Lofti Zadeh nevéhez fűződik. (1965)

- A hagyományos kétértékű logika helyett végtelenértékű, **folytonos változókat** és a hamis és igaz közötti folytonos átmenetet használ a fogalmak, jellemzők igazságértékének megadására.
- Hiányos adatok elfogadása, zajtűrő képessége, kollektív döntést utánzó működése a mesterséges neurális hálózatokhoz hasonlóan **robusztus**, megbízható alkalmazások készítését teszi lehetővé bizonytalan adatok esetére is.
- Szimbolikus szinten működik, **nyelvi változókat** használ, ezáltal közel áll az emberi gondolkodáshoz.
- Alkalmazása különösen a szabályozástechnikában igen elterjedt.



A Fuzzy logika alapjai ..

Fuzzy halmazok

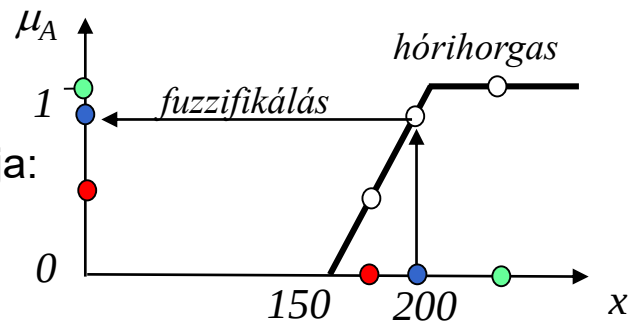
- Egy elemnek a fuzzy halmazba való tartozásának foka, 0-1 közötti mértéke van, nem merev igen/nem.

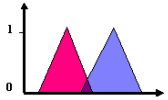
- Értelmezés:
  Ha **X** az x jellemzők, értékek egy csoportja, akkor az **X**-en értelmezett **A** fuzzy halmaz a következő rendezett párok halmaza:

$$\mathbf{A} = \{ (x, \mu_A(x)) : x \in \mathbf{X} \}$$

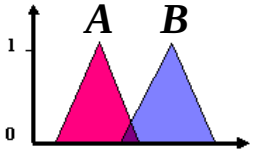
A $\mu_A(x)$ neve *tagsági függvény*, értéke az x tagsági aránya, beletartozásának mértéke az **A** fuzzy halmazba, ill. az **A** tagsági függvényhez társított nyelvi érték igazságának foka. A $\mu_A(x)$ tagsági függvény az **X** értékhalmazt leképezi az **M** tagsági térbe ($\mu_A(x) \in [0;1]$).

Példa: **X** testmagasságértékek [cm].
A halmaz: a „hórihorgas” nyelvi értéknek megfelelő x, $\mu_A(x)$ számpárok halmaza, ábrája:
fizikai változó: x= 200 ;
fuzzy változó: testmagasság = hórihorgas igazságértéke= 0.9





A Fuzzy logika alapjai ..



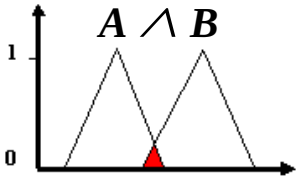
Fuzzy logikán alapuló következtető rendszer

- Produkciós, szabályalapú működés, de:
- a szabályok feltételoldalán fuzzy ÉS, ill. fuzzy VAGY logikai műveletekkel összekötött

  fuzzy változó = nyelvi érték

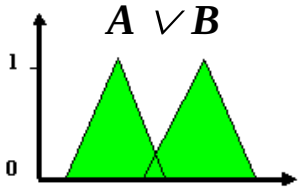
  beletartozási relációk vannak, melyek értékét a fuzzy változóhoz tartozó fizikai változó (x) aktuális értéke és a nyelvi értékhez tartozó tagfüggvény együttesen határozzák meg.

- A feltételoldali logikai kifejezés eredő érvényességi, igazsági értékét a fuzzy ÉS, ill, VAGY logikai műveletek értelmezésének felhasználásával számítjuk és ez lesz a következmény oldal érvényessége is egyben.



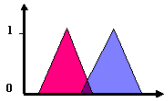
- Fuzzy ÉS művelet értelmezése (**C**= **A** $\wedge$ **B**):

  $$\mu_C(x)= \min\{ \mu_A(x), \mu_B(x) \}, x \in \mathbf{X}$$



- Fuzzy VAGY művelet értelmezése (**C**= **A** $\vee$ **B**):

  $$\mu_C(x)= \max\{ \mu_A(x), \mu_B(x) \}, x \in \mathbf{X}$$



A Fuzzy logika alapjai ..

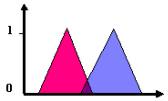
Fuzzy logikán alapuló következtető rendszer..

- A következmény oldalon egyetlen
  fuzzy változó = nyelvi érték
  reláció áll, melynek érvényessége, igazságértéke megegyezik a feltételoldalra meghatározott értékkel.

- Az összes szabály egyszerre működik a fizikai változók, mint bemenő tények felhasználásával és kollektív döntést hoz. A kollektív döntés eredményét a fuzzy nyelvi értékek tagfüggvényeinek felhasználásával, a fuzzy változó terében az érvényességi értékekről a fizikai változóra való áttéréssel, ún. defuzzifikálással nyerjük.

- A defuzzifikálás módszerei:
  - Területközéppont meghatározással (legelterjedtebb eljárás)
  - Tagsági függvény maximumok súlyozott átlagának meghatározásával
  - A legnagyobb érvényességű következmény-tagfüggvény maximumával.

Forrás: Erdélyi Ferenc: A technológiai menedzsment informatikai eszközei; információ rendszerek Miskolc, 1997.

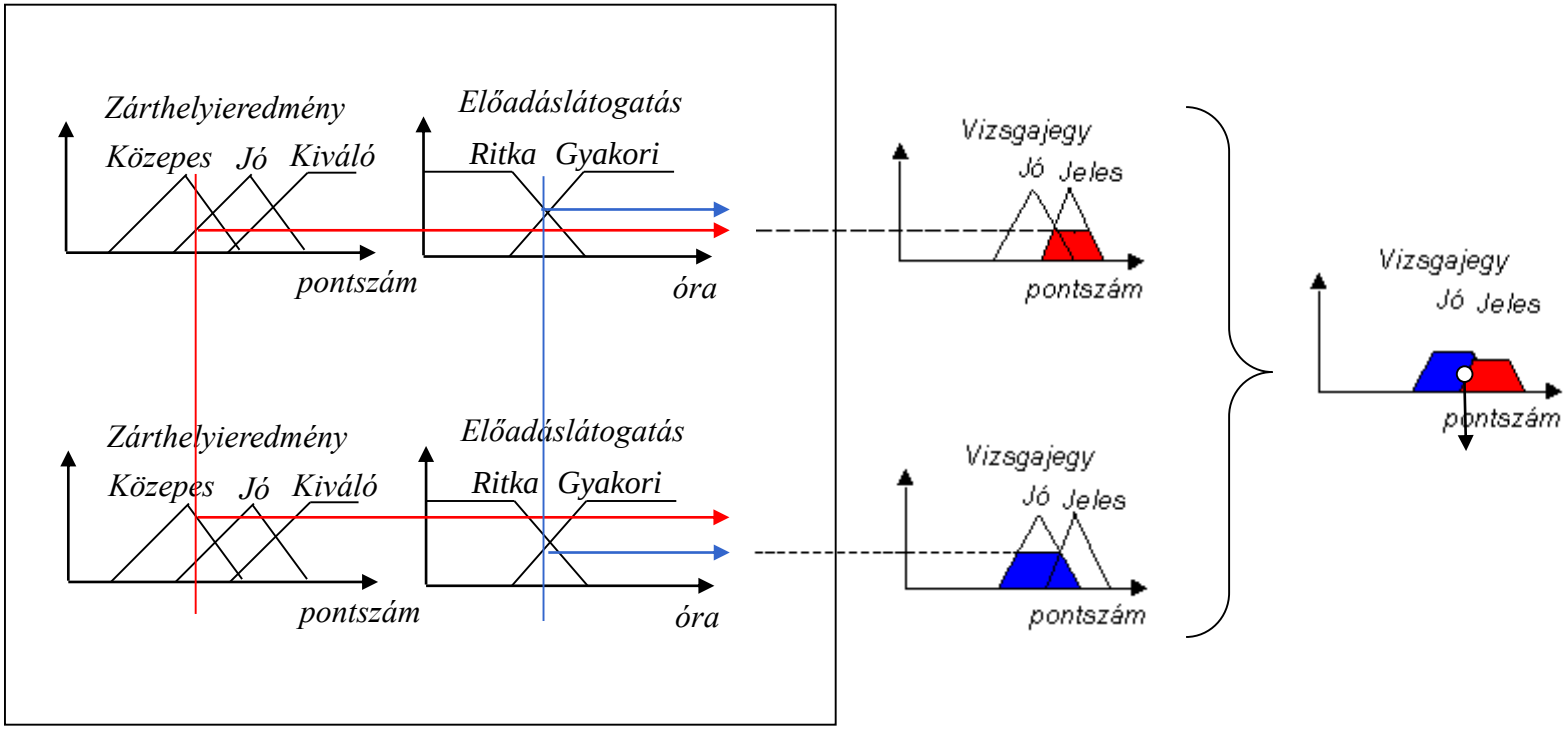


- **A Fuzzy logika alapjai ..**

Fuzzyfikálás, defuzzifikálás

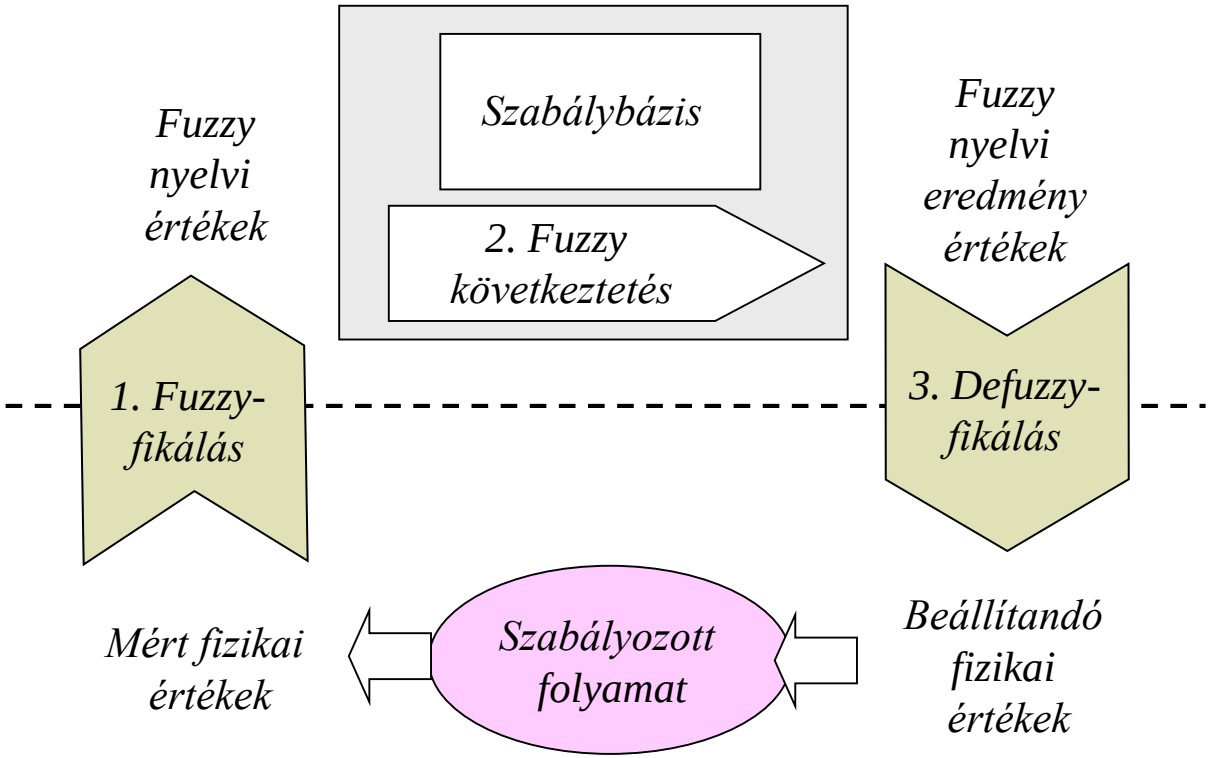
If Zárthelyieredmény = Jó ÉS Előadáslátogatás = Ritka then Vizsgajegy = Jeles

If Zárthelyieredmény = Közepes ÉS Előadáslátogatás = Gyakori then Vizsgajegy = Jó



- **A Fuzzy logika alapjai ..**

Fuzzy szabályozórendszer vázlata



Mesterséges intelligencia-orientált programozási nyelvek

- A speciális programozási nyelvek kifejlődését egy-egy **szakterület sajátos elvárásai**, a szakterület feladatait jellemző közös tulajdonságok motiválják.
- A mesterséges intelligencia kutatások is kiérleltek több olyan programozási nyelvet, ill. keretrendszert, amelyben a tématerület alkalmazásait könnyebb elkészíteni, hiszen a fejlesztői környezetek már **fél megoldást** jelentő eszközöket nyújtanak.
- A speciális segédeszközöktől az általános felhasználhatóság igényével fellépő programnyelvekig terjed a kínálat.
- Az MI nyelvek a szimbólumkezelés eszközei.
- Az utóbbi időben a klasszikus MI nyelvek mellett előtérbe kerül a C++ is.

Forrás: Yoshiaki Shirai - Jun-ichi Tsujii: Mesterséges intelligencia Alapelvek és alklamazások NOVOTRADE Rt. 1987.

Klasszikus mesterséges intelligencia nyelvek

- Legismertebb és legrégebbi a **LISP** (*John Mc Carthy*,1957).
- Számtalan LISP dialektus van, a legismertebbek: a Common LISP, az INTERLISP, a MacLISP, valamint a Standard LISP. Mivel a LISP-re nincs szabvány, ezek eltérései az alkalmazói programok hordozhatóságát megnehezítik.
- Erős hatást gyakorolt a későbbi nyelvekre.
- **QA1, QA2, QA3** programnyelvek, 1960-as évek. Általános problémamegoldó rendszerek, illetve **tételbizonyítás** céljára. **Szimbolikus logikát, rezolúciót** alkalmaztak.
- **QA4,** 1968. A PLANNER programozási nyelv hatása érezhető rajta. Szimbolikus reprezentáció mellett procedurális reprezentációra is képes volt.

Klasszikus mesterséges intelligencia nyelvek ..

1.

- A **PLANNER** programozási környezetet *C.Hewitt*, a MIT hallgatója dolgozta ki 1968-ban.
- A PLANNER-nél már megjelent a deklaratív nyelvek fő jellemvonása: csak a problémakört deklaráló, objektumokra és eljárásokra vonatkozó **tudást**, valamint a **megoldandó problémát** kellett megadni, a probléma megválaszolásához vezető utat, a szükséges tudás- és eljáráselemeket az út bejárásához maga a rendszer keresi meg és használja fel.
- Az ábrázolt kétféle tudásforma a PLANNER-nél fizikailag is elkülönült: a deklaratív tudást a **tényadatbázis**ban, a procedurális tudást a **procedure** (eljárás)-**adatbázis**ban tárolta.

1. http://www.ai.mit.edu/people/hewitt/hewitt.html

- **Klasszikus mesterséges intelligencia nyelvek ..**

  - A PLANNER előnyei:
    - **Mintaillesztés:** ha adott az atomok változókat tartalmazó listája, akkor az illeszkedő tények megkeresése, és ezzel a változók értékeinek meghatározása automatikus.
    - **Az eljárásokat a minták hívják.** Egy adott célhoz illeszkedő eljárás hívását egyszerűen a cél megadása váltja ki automatikusan, anélkül, hogy az eljárás nevének meghatározása szükséges lenne.
    - **A keresés automatikus.** A PLANNER az adatbázisokból alkalmas tényeket és eljárásokat választ ki a cél eléréséhez.
    - **Visszalépés**: a kiválasztott tételek nem feltétlenül helyesek. Ha a választás a keresés sikertelenségéhez vezet, a PLANNER visszatér a közvetlenül megelőző döntési ponthoz, és újraválaszt. (Depth first)
    - **Dinamikus tényadatbázis**: automatikusan is bővül a kikövetkeztetett tényekkel, tények törölhetők.

- **Klasszikus mesterséges intelligencia nyelvek ..**
  - **CONNIVER** programnyelv, *G.J.Sussman*, MIT kutató, 1972.
    - A CONNIVER a PLANNER kiterjesztett változata, használja a tényadatbázist és az eljárás-adatbázist, valamint a minta alapján történő keresést.
    - **Újdonsága** abban volt, hogy **elhagyta az automatikus keresést**, és a problémamegoldás vezérlését a programozóra bízta. Ezáltal a keresés **hatásfoka** megnőtt.
    - További beépített függvények.
  - **POP-x** programok: európai fejlesztés.
    - Legújabb verziójuk, a POP-11 négy MI nyelvet integrál egy fejlesztői környezetbe, közte: LISP, PROLOG, **POPLOG**. A nyelvek közül egy feladaton belül is szabadon választhatunk.
    - A POPLOG nyelv a POP-11 készítőinek teljesen saját fejlesztése.
    - Erőteljes grafikai lehetőségek, többféle hardver- platform.
    - Interpreteres üzemmód, inkrementális fordítás.

dr.Dudás László

A PROLOG

- **PROLOG** (Programming in Logic, Logika alapú programozás) *Alain Colmerauer, Bob Kowalski,* 1970.

1.

1.

- **Predikátum konstanson alapul**ó általános célú programnyelv, amely azonban legelőnyösebben logikai problémák megoldására alkalmazható.

- A PROLOG létrehozására kihatott az a tapasztalat, amelyet a korábbi MI programozási nyelvekkel szereztek: minél több a rendszerbe épített lehetőség, a rendszer annál inkább lelassul.

- **Deklaratív** programnyelv: elegendő megadnunk csak a problématerületre vonatkozó tényeket, szabályokat és a megválaszolandó kérdést, ill. minősítendő állítást, és a **program generálja a megoldáshoz vezető lépéseket** és szolgáltatja a választ.

1. http://www.york.ac.uk/depts/lang/webstuff/people/sjh/L333intro_to_prolog/creators.html

Bevezetés a PROLOG programozási nyelvbe

- Jellemzői:

  Hasonló a PLANNER-hez a következőkben:

  - Az eljárások és a tények a **mintaillesztés** során kerülnek elő. Eltér viszont abban, hogy betartja a bemenőállítások **sorrendjét**.
  - A keresést automatikus **visszalépéssel** (back track) hajtja végre, amelyet azonban a programozó a ! (cut, **vágás**) operátorral módosíthat. Ez a lehetőség **ellentmond a logikai programozás ideális koncepciójának**: a programozó feladata csak a logikai probléma megadása, a problémaleírás algoritmikus kiértékelését pedig a gépnek kell elvégeznie.

    Kowalski képlete:

    algoritmus = logika + vezérlés

    Jelentése a következő: egy algoritmus mindig két komponenst tartalmaz implicit formában:

    – a logikai összetevőt, amely megadja a kérdéses problémára vonatkozó tudást,

    – a vezérlési összetevőt, mely megtestesíti a megoldási stratégiát az adott probléma esetére.

dr.Dudás László

A PROLOG jellemzői

A Prolog további jellemzői:

- Bevezetésre került a **symbol** típus (domain), hogy a predikátumlogika individuumkonstansai, -változói és -függvényei a valóság egy részletét leképezhessék.
- Fontos adatszerkezet a **lista**. A listafeldolgozás a **mintaillesztésen** és a **rekurzión** alapul.
- A függvények a predikátumfüggvényeken alapulnak, és **nincs különbség a bemenő és a kimenő változók között.**
- A bemeneti állítások (**szabályok**, clauses) formátuma korlátozott:

$$\underbrace{\mathbf{P}}_{\textit{fej}} \quad \mathbf{if} \quad \underbrace{\mathbf{Q\ and\ R\ and\ S.}}_{\textit{törzs}}$$

A fej egyetlen predikátum, a törzs pedig atomi formulák (többnyire predikátumok, azaz logikai függvények) negációt nem tartalmazó konjunkciója (csak *and* lehet).

A PROLOG jellemzői ..

A Prolog további jellemzői .. :

- Mivel a törzs atomi formulák konjunkciója, azaz Horn-halmaz, így azt mondhatjuk, hogy a PROLOG **a Horn halmazokra a rezolúció elvét alkalmazza.**

- **A felhasználóra hagyja a rezolúciós stratégia megvalósítását**: néhány megszorítást tett a bemenőállításokra, és az illesztés a bemenőállítások (clauses, tények és szabályok) megadott sorrendjében hajtódik végre. A rendszer így tömörebb lett és gyorsabb végrehajtást tesz lehetővé.

- A predikátum logikára épül, de vannak eltérések:
  - Csak az elemváltozók kezdődnek **nagybetűvel**.
  - Ítéletkonstansok közül a False (fail) **ritkán**, a True és ítéletváltozók még ritkábban fordulnak elő.
  - **Rendszerpredikátumok** léteznek, melyek valamit elvégeznek és eközben igaz értéket vesznek fel. Pl.: read(). write(), circle(), stb.
  - **Infix** (közbeírt) alakú relációoperátorok (<,>, <=,>=,<>,=,><) és aritmetikai operátorok (+,-,*,/) a predikátumos alak helyett. Pl.: kisebb(X,Y) helyett X<Y írható.

dr.Dudás László

A PROLOG jellemzői ..

A Prolog további eltérései a predikátum logikától:

- Egyenlőséges reláció egyetlen hiányzó értékű változójának képes **kiszámítani** az értékét. Pl.: 5*7= X $\Rightarrow$ X=35
- Aritmetikai kifejezéseket, műveleti operátorokat végrehajtja.
- A szabályokban szereplő ***if*** (:-) logikai szimbólumot úgy kezeli, mint egy infix alakban használt speciális rendszer-predikátumot, amely a PROLOG-tól egy speciális kiértékelő algoritmust kíván.
- A szabályok és az atomok (főként predikátumok) ugyanolyan módon kezelendő objektumoknak minősülnek.
- Megengedi **ítéletváltozók** alkalmazását a szabályokban és megengedi **helyettesítésüket szabályokkal**.
- Képes manipulálni saját "adatbázisát" (az **assert** és a **retract** rendszerpredikátumok útján.)
- Kényelmes **[Fej | Törzs]** listaalak. [ ] az üres lista jele.
- A megoldási folyamat programozó általi vezérlésére alkalmazza a **! jelet**, mely formailag egy atomi formula. Pl.: a if b and c and ! and d. A ! (**cut**) megváltoztatja, lerövidíti a PROLOG Depth-First (mélységi bejárás) algoritmusát a felesleges ágak bejárásának letiltásával, az ágak levágásával.

A PROLOG jellemzői ..

A PROLOG programok felépítésének és eszköz-rendszerének részletezése előtt tekintsünk egy egyszerű feladatot és PROLOG programját!

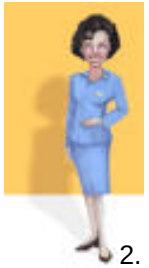
Ismertek az alábbi **tények**:

Cleopátra szebb Ginánál.

Gina szebb Ursulánál.

Kérdés: *Ki sokkal szebb Ursulánál ?*

1.



2.

3.

- A feladat megválaszolására alkalmas **PROLOG program**:

```
domains    hölgy = symbol
predicates     szebb(hölgy, hölgy)
               sokkal_szebb(hölgy,hölgy)
goal           sokkal_szebb(Valaki, ursula)
clauses    szebb(cleopátra,gina).
           szebb(gina, ursula).
           sokkal_szebb(A,C ) if szebb(A,B ) and szebb(B,C ).
```

1. www.dollsinmini.com/ showcase.htm 2. www.meetingadvice.com/columnists/ index.php3?Columnist=2
3. www.theatlantic.com/issues/ 98may/ursula.htm

A PROLOG működése

A működés két alapvető összetevője az

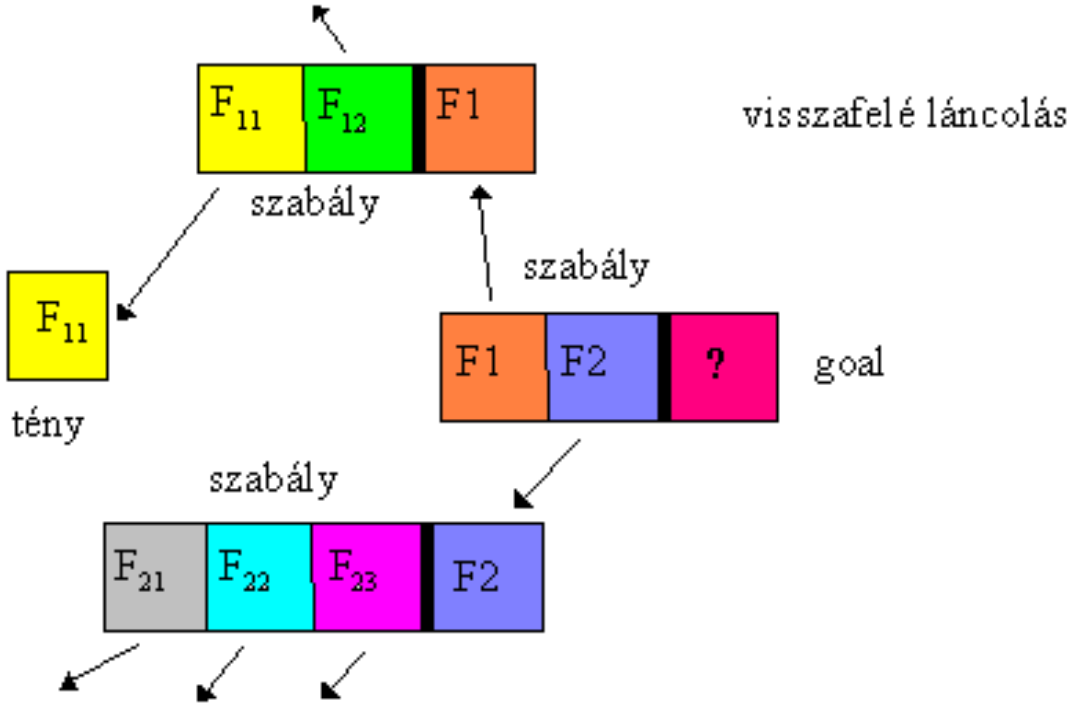
- **illesztés** (matching) és a
- **visszalépés** (backtracking).

- Lényege: a goal-ban feltett kérdést megpróbálja **tényállításokra visszavezetni**. Amennyiben nem létezik illeszkedő - nevében és a megadott paramétereiben egyező - tény, akkor illeszkedő szabályt keres. Ha van illeszkedő szabály, akkor az előbbi lépések ismétlésével megpróbálja a szabály Törzsében lévő feltételek kiértékelésével meghatározni a szabály Fejének igazságértékét (Hátraláncolás).

- Az **összes** lehetőség végigpróbálásával megállapítja, hogy a kérdés teljesülhet-e, és ha igen, hányféle módon.

- Ha próbálkozás közben zsákutcába jut - nincs illeszkedő tény, vagy szabály -, **visszalép** a legutolsó elágazási pontig, ahol a több illesztési lehetőség közül a következőt választja.

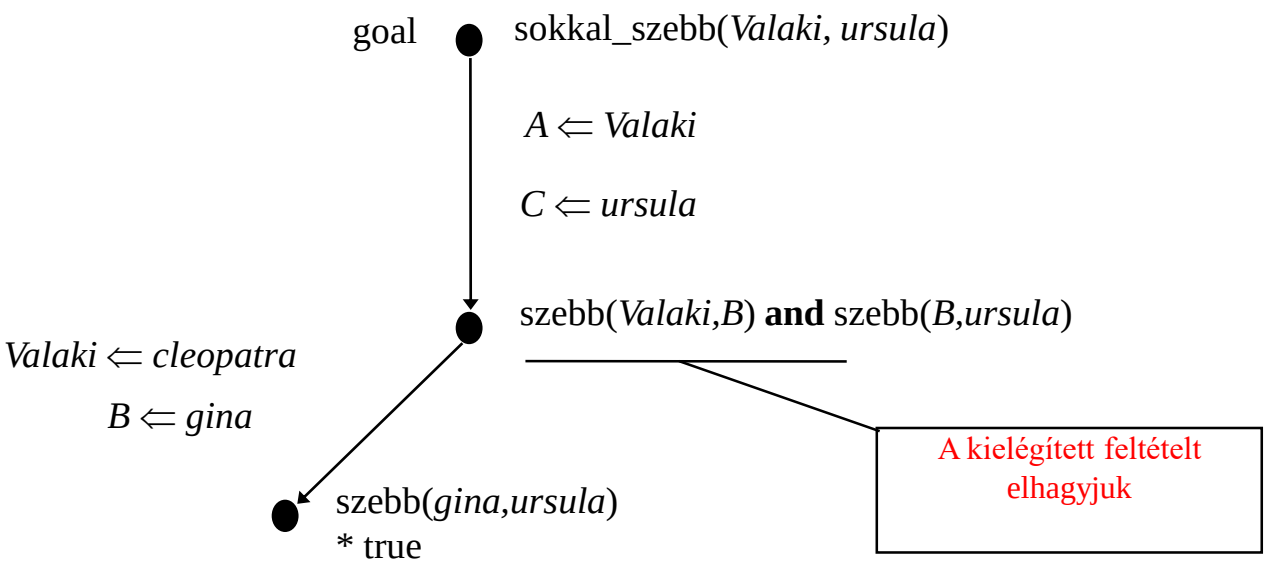
- **A PROLOG működése ..**



A PROLOG működése ..

Szemléltetés fagráffal:

domains hölgy = **symbol**
predicates szebb(*hölgy, hölgy*)
sokkal_szebb(*hölgy, hölgy*)
goal sokkal_szebb(*Valaki, ursula*)
clauses szebb(*cleopátra,gina*).
szebb(*gina, ursula*).
sokkal_szebb(*A,C*) **if** szebb(*A,B*) **and** szebb(*B,C*).



A PROLOG működése ..

Szemléltetés fagráffal:

domains hölgy = **symbol**

predicates szebb(*hölgy, hölgy*)

sokkal_szebb(*hölgy, hölgy*)

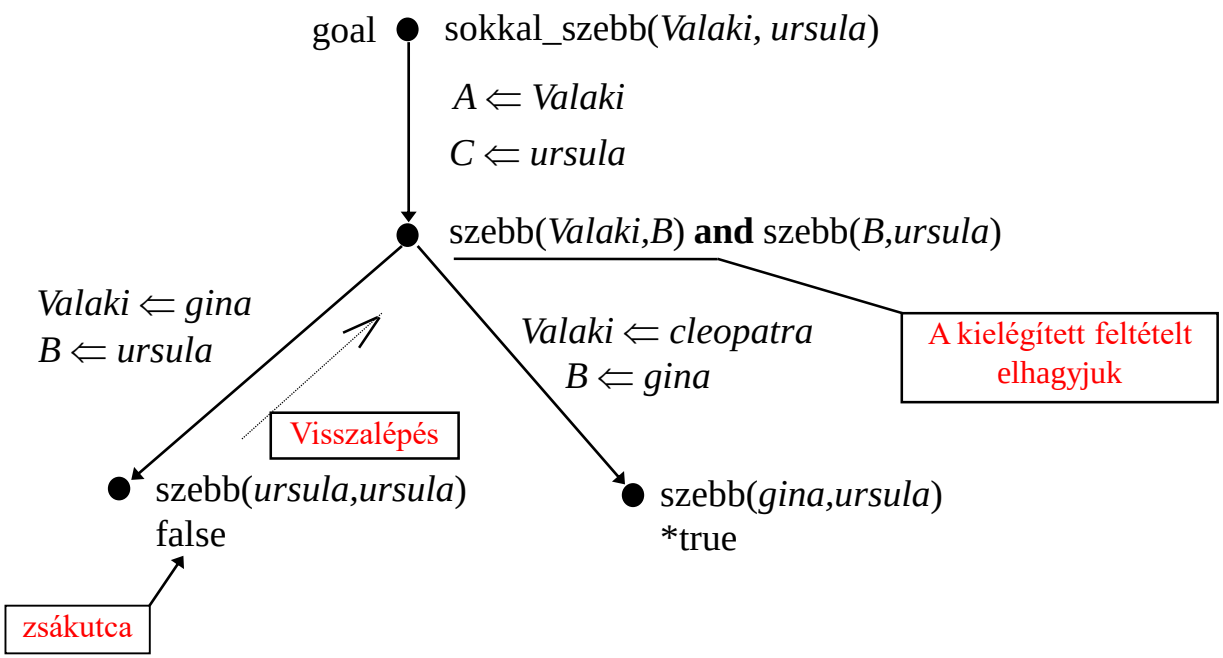
goal sokkal_szebb(*Valaki, ursula*)

clauses szebb(*gina, ursula*).

szebb(*cleopátra,gina*).

sokkal_szebb(*A,C*) **if** szebb(*A,B*) **and** szebb(*B,C*).

A tényeket felcseréltük



- **Szemantikus háló alapú tudásszemléltetés**
  Keret és forgatókönyv alapú tudásszemléltetés
  Eset alapú rendszerek

- **A szemantikus háló** (Ross Quillian, 1968)

- **Biológiai ihletés**: az agy fogalomábrázolásának feltételezett formája
  - objektumok, koncepciók**,** fogalmak - csomópontok
  - viszonyaik, kapcsolataik - gráfélek.

- **Hierarchikus modell** oka: az ember kognitív (megismeréssel kapcsolatos) működésére vonatkozó kísérletek az objektumok specifikus jellemzőinél gyorsabb válaszidőket eredményeztek, mint az általános, magasabb szintű kategóriákhoz tartozó jellemzőknél (*Kanári - énekel? Kanári - repül? Kanári - bőre van?*)
  Következtetés: hierarchikus egyed - alosztály - osztály kapcsolat valószínű.

- **Cél**: Az emberi információtárolás és visszakeresés modellezése.

geocities.yahoo.com.br/peixe_betta/ FotosCanari.html

Szemantikus háló alapú tudásszemléltetés

- Szoftveralkalmazás, célja: a természetes nyelvek megértésének gépi modellezése.

- **Quillian:** ***"egy szó jelentését meg lehet kapni a szóhoz társított szövegek halmazaként"***

- Programjával az ember gondolkodási folyamatának azt a részét modellezte, amelyet az ember akkor végez, amikor egy **lexikon két szava között** keres kapcsolatot. Egy szemantikus háló jól definiált eljárásainak segítségével képes volt összeállítani egy választ bármelyik, a szótárban megtalálható szópár esetében azok összevetésére és szembeállítására.

- Megkereste a szavakhoz kötődő ismeretek közös részeit, kapcsolódási pontjaikat. Emiatt szokták ***asszociatív hálónak*** is nevezni a szemantikus hálót.

Szemantikus háló alapú tudásszemléltetés ..

- Példák a Quillian által elkészített program működésére szavak közötti kapcsolat keresésekor:

1.Összevetés: *Sírni, Kényelem*

A. Kapcsolat: *Szomorú*

(1) A *sírás* azon dolgok közül való, amelyek *szomorú* hanggal párosulnak.

(2) A *kényelem* érdekében valamit kevésbé *szomorúvá* tehetünk.

2.Összevetés: Növény, Élő

A. 1.Kapcsolat: Élő

(1) A *növény élő* szervezet.

B. 2.Kapcsolat: Élő

(2) A *növény* olyan szervezet, amely a levegőből is vesz magához táplálékot.

Ez a táplálék olyan dolog, amely kell az *életéhez*.

3.Összevetés: Növény, Ember

A. Kapcsolat: Emlős

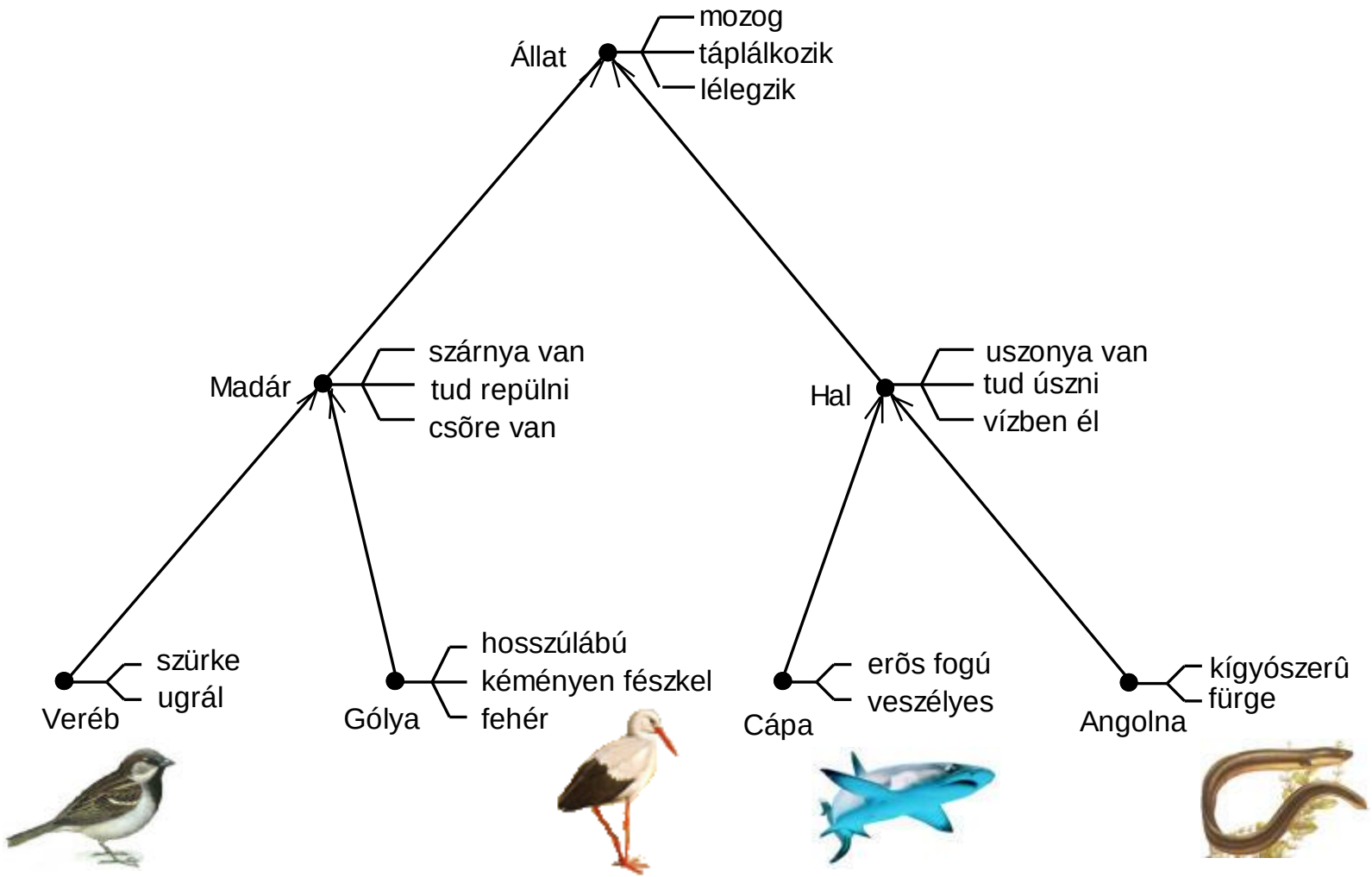
(1) A *növény* nem *emlős* szervezet.

(2) Az *ember* egy *emlős*.

Christopher F. Chabris: **Artificial Intelligence & Turbo Pascal** Dow Jones-Irwin, Illinois,1987. p395.

Szemantikus háló alapú tudásszemléltetés ..

- Tekintsünk egy konkrét példát a Quillian-féle szemantikus hálóra!



www.rainieraudubon.org/bib/nestbox/ sparrow-starling-info.html www.earthlife.net/birds/ ciconiiformes.html
www.calacademy.org/geninfo/ hours_admissions.html www.myspace.co.uk/nessie/ nessie/fish.html

Szemantikus háló alapú tudásszemléltetés ..

- A nyilak az **egyed-alosztály** (instance_of) és az **alosztály - osztály** (is_a) kapcsolatot jelzik, a többi gráfél a **tulajdonsága** (featured__by) kapcsolat megadására szolgál.
  További lehetséges kapcsolatformák:
  típusa; van_neki; tud; azonos_vele; stb.

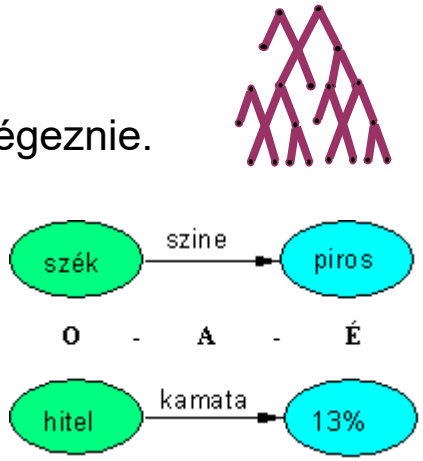
- Léteznek gazdagabb kapcsolatkészlettel dolgozó szemantikus háló reprezentációk is, lásd CD elmélet.

- **Következtetés, feladatmegoldás szemantikus hálóval**
  Egy szemantikus hálóval ábrázolhatjuk egy tématerület ismereteit. A tématerülettel kapcsolatos ismeretek alapján megválaszolható kérdést szintén egy neki megfelelő szemantikus hálóval adjuk meg (**célháló**), majd ezt a kisebb hálót illesztjük a tématerület hálójának azonos csomópontokat tartalmazó részére. A kérdésre a választ a tématerület hálójának illeszkedő része hordozza.
  Általában a hálókezelő algoritmusra ennél több feladat hárul, elő kell állítania például az öröklött tulajdonságokat is.

dr.Dudás László

Az egyed - alosztály - osztály kapcsolat

- **Egyed**: van olyan tulajdonsága, amely csak rá igaz
- **Osztály**: olyan kategória, melynek jellemzői több egyedre, vagy alsóbb osztályra is igazak, így ezek az egyedek, vagy alosztályok közös tulajdonságaik alapján egy magasabbszintű osztályba sorolhatók.
- A közös tulajdonságot csak az osztálynál kell tárolni, az egyedekre, vagy alosztályokra **öröklődik**.
- A tulajdonságörökítés (inheritance) általánosan értendő: nemcsak a tulajdonság jellegű kapcsolatok öröklődnek, hanem például a birtoklást, valamilyen érzelem irányulását, stb. is beleértjük.
- A tulajdonságörökítést a hálókezelő programnak kell végeznie.
- A taxonomikus kapcsolat: osztálybatartozás.
- Másik fő kapcsolati forma: tulajdonság hozzárendelés, objektum-attribútum-érték hármas.



Szemantikus háló előnyei

- Az **osztályhierarchia** a tulajdonságok hatékony tárolását is segíti: az osztály minden alosztályára, illetve egyedére egyaránt érvényes tulajdonságok a legmagasabb, legáltalánosabb szinten kerülnek tárolásra, azonban érvényesek a becsatlakozó alsóbb osztályokra és egyedekre is - működik az **öröklődés**. Kiküszöböli a redundanciát, ellentmondás-mentességet eredményez.

- A **grafikus ábrázolás** szemléletes és könnyű érthetőséget jelent és az emberi gondolkodáshoz közel áll.

- **Gyors** számítógépi reprezentációt tesz lehetővé: a csomópontok memóriaterületekre, az élek **mutatókra** képezhetők le. Elmarad a listák elemeinek kimerítő illesztése, mely a szabály- és logika alapú ismeretszemléltetést jellemezte. Az objektumok megtalálása ún. *hash* táblák segítségével gyorsan megtörténhet, a kapcsolatoknak megfelelő mutatók pedig meghatározzák a kapcsolódó ismeretelemek helyét.

- **Rugalmas** tudásszemléltetési eszköz: könnyen bővíthető új objektumokkal és viszonylatokkal, a módosítás és a törlés egyszerű.

A jelentés tárolása

- A szemantikus háló **önmagában nem hordozza** a teljes jelentést. A kapcsolatok értelmezése a szemantikus hálót szemlélő emberre, ill. a hálót kezelő **algoritmusra** hárul.

- Például mondhatjuk azt, hogy a Cápa a Hal osztály egyede, de jelenthetné a kapcsolat azt is, hogy a Cápa a Hal objektum mellett él.

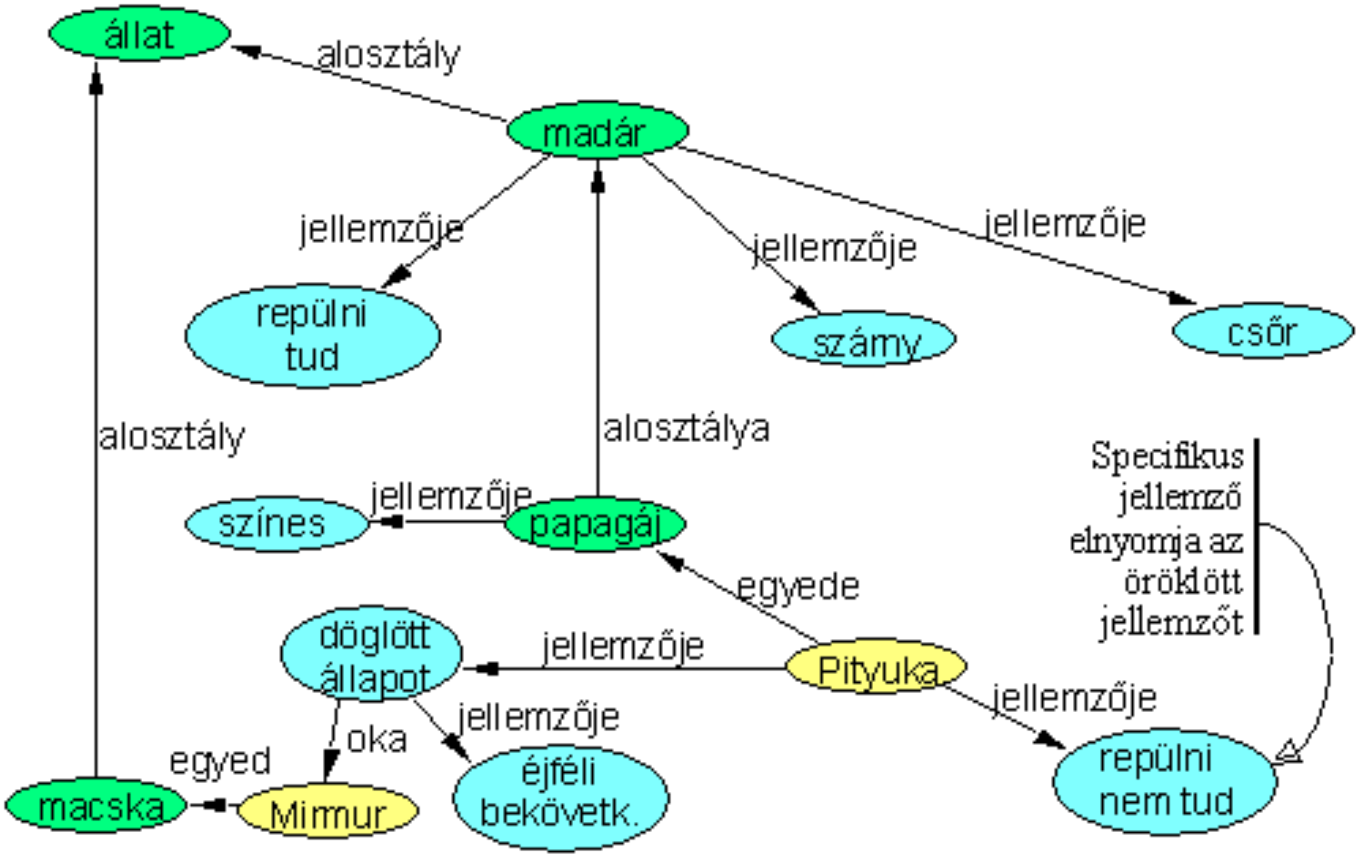
- Az ábra által sugallt értelmezéstől eltérő értelmezés alkalmazása az ember számára **a fogalmakhoz kötődő tudása miatt erőltetettnek tűnhet**, de a számítógép számára csak annyi, az objektumokhoz kötődő ismeret áll rendelkezésre, amennyit a háló kapcsolatrendszere és az azt kezelő algoritmus megtestesít.

- Előnyös lenne az algoritmus szempontjából, ha **az összes kapcsolatot**, amely egy szemantikus hálóval történő tudásszemléltetésben előfordulhat, az algoritmus elkészítése előtt néhány csoportba sorolhatnánk. Ez lehetővé tenné a szemantikus háló működtető, értelmező részének a hálótól független elkészítését, és számtalan olyan háló értelmezését, melyek csak egy adott tématerület fogalmait (objektumait) és viszonylataikat tartalmazzák.

dr.Dudás László

Az öröklés problémái

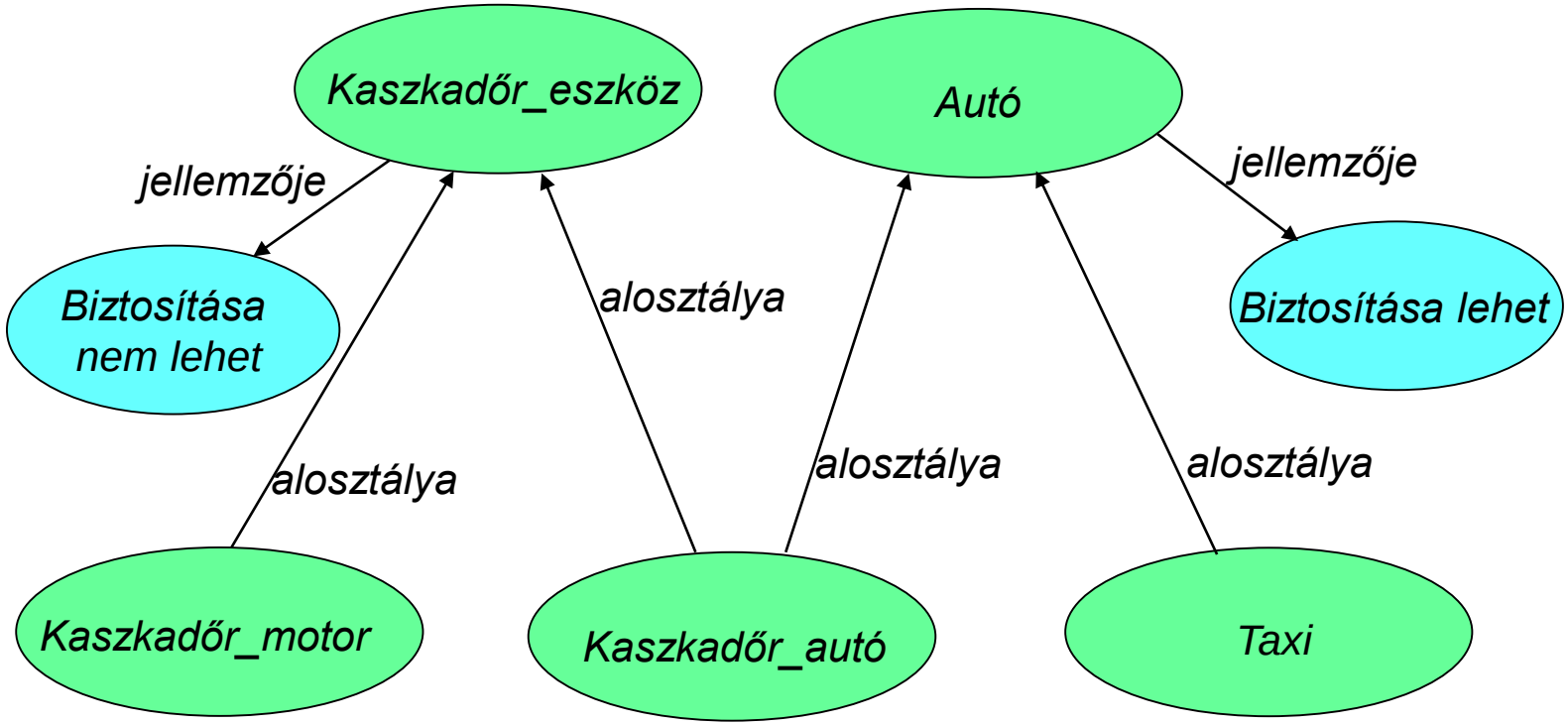
- Kivétel: Pityuka repülni nem tud.



dr.Dudás László

- **Az öröklés problémái ..**

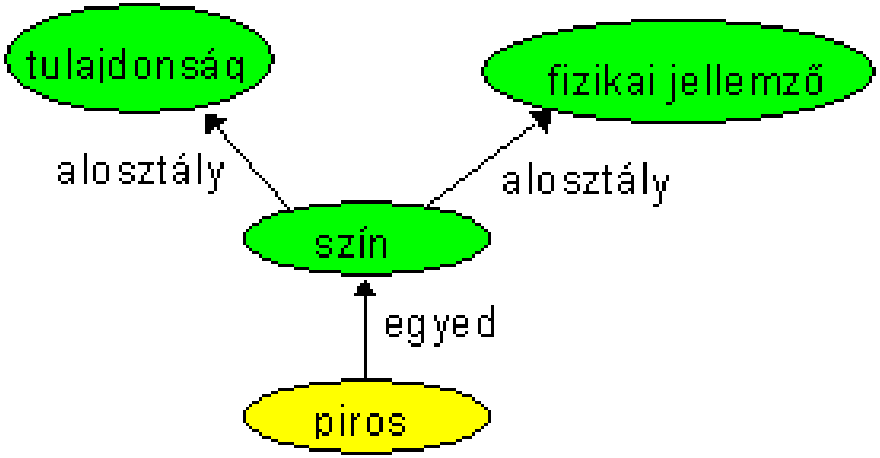
  - Ellentmondás: Kaszkadőr autóknak lehet, vagy sem biztosítása?



Feloldási módszerek: alapértelmezés; prioritás alkalmazása.

A szemantikus háló egyéb problémái

- A *típus/egyed megkülönböztetés* szükségessége (Type/Token Distinction).



A piros egy szín
A szín egy fizikai jellemző.

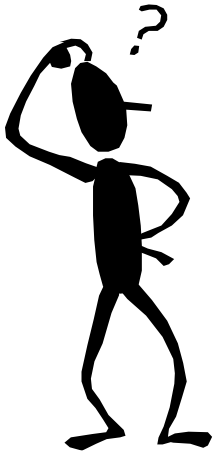
A piros egy fizikai jellemző?

A probléma ott érezhető, hogy a piros esetében az örökölt fizikai jellemző mivolta nem tűnik helyesnek. Ennek magyarázata abban van, hogy a fizikai jellemző tulajdonság a színhez, mint osztályhoz (type) kötődik, és nem az osztály egyedéhez (token).

A szemantikus háló egyéb problémái ..

- Másik érzékenységet igénylő probléma az **intencionális és az extencionális** értelmezés eltérése ugyanazon dolognál. Ugyanazon dolog intencionális leírása a lényeget ragadja meg, míg az extencionális megadás a dolog leírása által megadott valós objektumok halmaza. Pl. a *kutya* fogalom extencionális jelentése az összes kutya, intencionális jelentése mindazok a gondolatok, melyek a kutyákhoz kapcsolódnak, azaz a "kutyaság" maga.

- Egyes szerzők a szemantikus háló hibájául rótták fel, hogy a logikai szemléltetéssel ellentétben nem tud olyan fogalmakat kezelni, mint: legalább egy nem specifikált objektum, összes objektum, stb.



Egy szemantikus hálón alapuló sikeres szakértőrendszer

- A **PROSPECTOR** egy geológusokat segítő konzultációs rendszer. Bár alapvetően szabályalapú rendszer, de az **adatszerkezete egy szemantikus hálón alapult**. A Stanford Research Institute hozta létre 1978-ban. Feladata volt a geológusok segítése az érclelőhelyek behatárolásában, és nagy területek ásványvagyonának kiértékelésében.

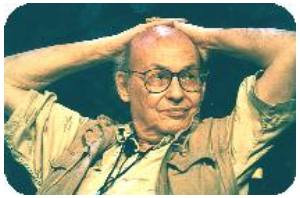
  Párbeszédes rendszer, amely több alrendszert foglalt magába: egy uránfeldúsulásokat leíró modellt, egy rézérc és egy molibdén modellt. A rendszernek nagyszámú, és jelentőségüket relatívan mutató tényezőket kellett kezelnie.

  A gyakorlati eredmények azt mutatták, hogy a rendszer 7%-os pontossággal volt képes megbecsülni az ércvagyon helyét és mennyiségét.

Peter S. Shell: **Expert Systems - A Practical Introduction** John Wiley & Sons, New York, 1986. p99.
www.disasterrelief.org/Disasters/ 020410chinamines/

Keret és forgatókönyv alapú tudásszemléltetés

A *keret* és a *forgatókönyv* tudásábrázolási forma újdonsága abban van a szemantikus hálóhoz képest, hogy a tudáselemeket **sztereotip egységekbe**, keretekbe, forgatókönyvekbe szervezi. Ezek az egységek *objektumoknak*, *tevékenységek-nek*, vagy *eseményeknek* felelhetnek meg.


1.

A **keretek (frames)** *Marvin Minsky* nevéhez fűződnek (1975).

A **forgatókönyveket (script-**ek**)** *Roger Schank* publikálta (1977).

2.

Mind a keretek, mind a forgatókönyvek kulcsgondolata az, hogy az elméletekre, eseményekre, szituációkra vonatkozó ismereteink mind ezen dolgokra vonatkozó **elvárásaink** körül szerveződnek

1. http://www.discover.com/99awards/awpic1.html 2. www.edge.org/documents/ archive/edge59.html

Keret alapú tudásszemléltetés

- **A keretek**
  A keretek a valós világra vonatkozó ismereteket oly módon reprezentálják, hogy **egyesítik** az objektumokra, tevékenységekre és eseményekre vonatkozó **deklaratív leírást** azon információk előállítására vonatkozó **eljárások megadásával**, melyek célok elérésének, információk megszerzésének módjára vonatkoznak, ilymódon túllépve több olyan problémán, mely a szemantikus hálót jellemezte.

- A keret tudásábrázolási forma bevezeti a ***prototípus*** fogalmát, mely azt a felismerést tükrözi, hogy az emberi ismerettárolás sok **sztereotip leképezést** tartalmaz.

- A valós, vagy elvont objektumokat, fogalmakat reprezentáló keretek **hierarchikus keretrendszerré** kapcsolódnak össze az ismeretábrázolás során.

- A keret alapú ismeretszemléltetés a szemantikus háló továbbfejlesztésének tekinthető.

Peter Jackson : **Introduction to Expert Systems** Addison-Wesley, New York, 1990. p526.

- **Keret alapú tudásszemléltetés ..**

Neve:	**Bútor**
Magasság:	< 4m
Magasság:	> 0.2m
. .	
Anyag:	fa
Súlya:	IF-NEEDED: Térfogat* fajsúly

Alosztály

Neve:	**Szék**
Szín:	barna
Lábszám:	4
Funkció:	ülőhely

Alosztály

Neve:	**Asztal**
Funkció:	étkezés
Magasság: default:	< 1.4m 1m
Magasság:	> 0.4m
Lábszám:	4

Alosztály

Neve:	**Bárszék**
Magasság:	1.2m
Lábszám:	3
Anyag:	alumínium

Egy keretrendszer részlete

Keret alapú tudásszemléltetés ..

A szemantikus hálókkal **megegyező** tulajdonságok:

- Hierarchikus egyed - alosztály - osztály szerkezet.
- Tulajdonság örökítés, mely kiterjed a procedurális tulajdonságokra is. Konfliktusok feloldása specifikusság, prioritás, vagy alapértelmezés figyelembe vételével.
- Hasonló számítógépes reprezentáció: keretek - memóriahelyek; kapcsolatok - mutatók. Gyors működés.
- Keretkezelő program a következtetés, problémamegoldás kivitelezésére, de jóval gazdagabb feladatkörrel.
- Grafikus ábrázolás használható, de a grafika inkább a keretleíró nyelvek támogatója. (Lásd KappaPC szoftvert.)
- Rugalmas tudás bővítés, módosítás, törlés.

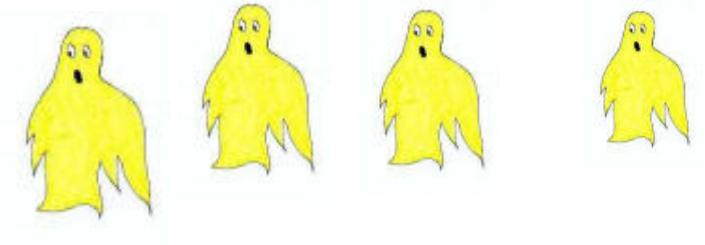
Keret alapú tudásszemléltetés ..

A szemantikus hálókon túlmutató tulajdonságok:

- Egységbefoglalás: objektum, attribútumok, értékek, deklaratív és procedurális összetevők. Slot - filler, attribútum - érték párok, speciális attribútum a keret neve.
- Az attribútumok és attribútum-értékek megadása más keretekre való utalással, többszörös egymásba ágyazással is lehetséges.
- Default, alapértelmezett értékek szolgálják a kérdések megválaszolását.
- A procedurális ismeretszemléltetés részeként értékeket előállító függvények, az értékváltozásokra működésbe lépő mechanizmusok, eseményvezérelt démon rendszer működik.

  IF_NEEDED
  IF_ADDED
  IF_MODIFIED
  IF_DELETED

  démonok működésének eredményeként dinamikus, élő rendszerek alakíthatók ki.



Keret alapú tudásszemléltetés ..

A szemantikus hálókon túlmutató tulajdonságok .. :

- Az attribútumok értékkészletére, értéktartományára, alap (default) értékére adhatunk meg előírásokat.

- A keret tudásábrázolás sokkal elterjedtebb, mint a szemantikus háló, mivel gyakorlatilag annak összes tulajdonságát magába foglalja. Speciális **keretkezelő nyelveket** hoztak létre a keretek használatának megkönnyítésére (FRL,KRL,OWL,NETL,KL-ONE, ART, stb).
  Ezenkívül több **hibrid**, azaz többféle tudásszemléltetési módszert egyesítő rendszerben is alkalmazásra került (KappaPC, Level5 Object, Nexpert Object/Smart Elements, Aion Development System, CBR Express, stb.).

dr.Dudás László

- **Példa eseményt leíró keretre**

Általános Előadás keret

Megnevezés: előadás

Terem:

Lehetőségek: római számos előadótermek, kb.20db,
arab számozású kistermek, kb. 200 db,
laborok, kb. 40 db.

Kezdési idő: 8:00, 9:00, ... , 18:00.

Időtartam: 40 perc - 180 perc.

Default: 50 perc.

Befejezés időpontja:

Ha szükséges: Kezdési idő + Időtartam.

Eszközök:

Lehetőségek: krétás tábla, filctollas tábla, számítógép, írásvetítő, diavetítő, projektor, video, TV, film, modell, laboreszközök.

dr.Dudás László

Példa eseményt leíró keretre

ME Alkalmazott Informatikai Tanszék MI előadás keret

Megnevezés: MI előadás
Terem:
    Lehetőségek: I, II, XXX.
    Default: I.
Kezdési idő: szerda, kb.14:00.
Időtartam: 160 perc - 175 perc.
    Default: 170 perc.
Befejezés időpontja (öröklött függvény):
        Ha szükséges: Kezdési idő + Időtartam.
Eszközök:
    Lehetőségek: krétás tábla, számítógép, írásvetítő, projektor.

http://www.lincoln.ac.nz/about/profile.htm

A forgatókönyvek

A *Schank*-féle *forgatókönyv* *koncepcionális primitíveket* és azok *kapcsolatait* rögzíti. A koncepcionális primitívek magasabb szintű elvonatkoztatásoknak felelnek meg.

Példa:

Előadás forgatókönyv

www.cf.ac.uk/international/ study/teaching.html

Feltételezések (díszletek): előadóterem, tábla, kréta, írásvetítő, projektor, transzparensek, filctollak.

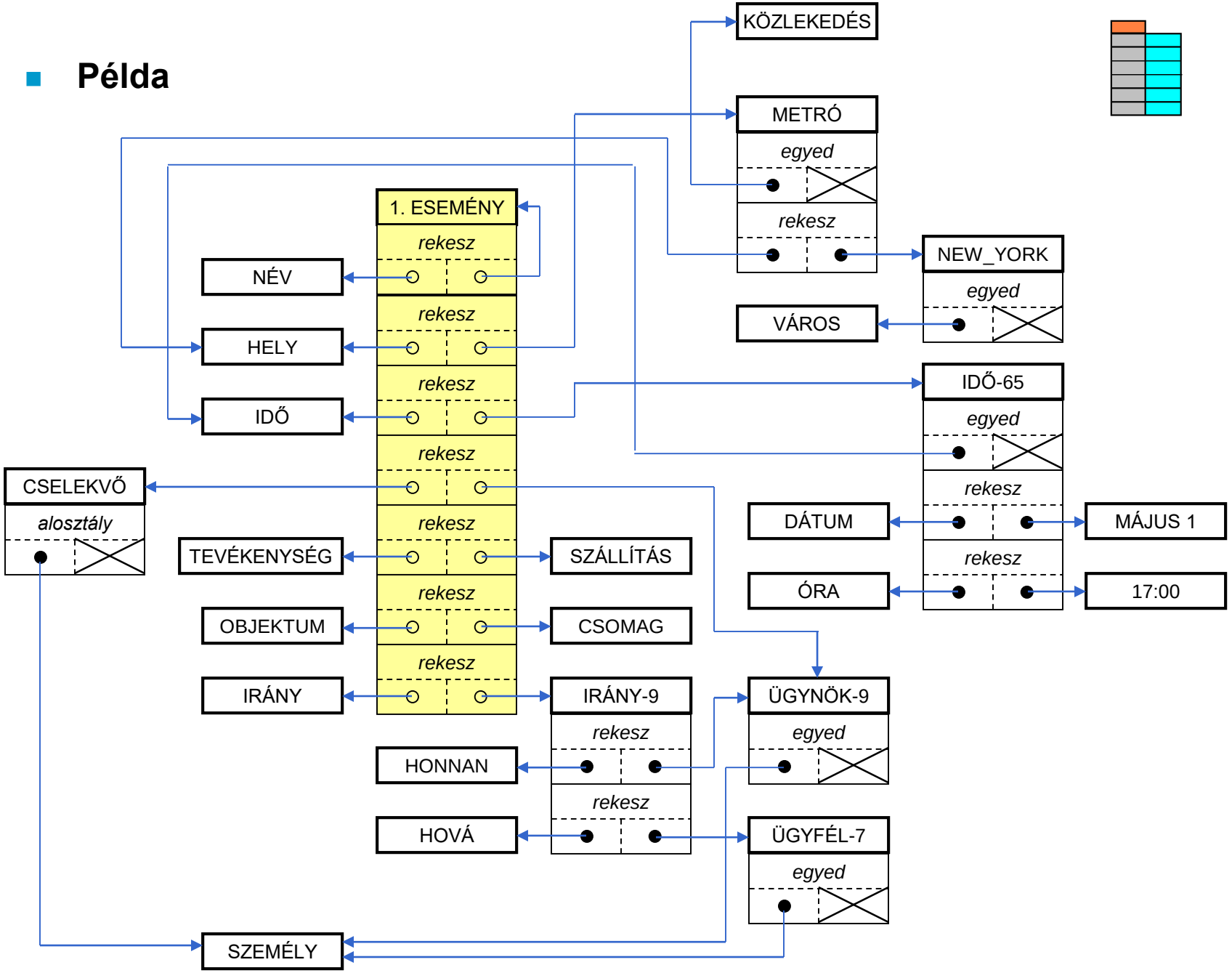
Szereplők (szerepek): diákok, tanár.

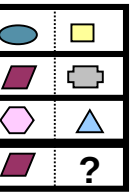
Nézőpont: tanár.

Eseménysorrend: 1. Belép a terembe
2. Hozzákészül, kivéve, ha nincs diák, mert akkor elmegy
3. Megtartja az előadást
4. Összeszedelődzködik
5. Elmegy.

Fő esemény: 3. Megtartja az előadást.

- **Példa**


KÖZLEKEDÉS
METRÓ
egyed
rekesz
NEW_YORK
egyed
VÁROS
1. ESEMÉNY
rekesz
NÉV
rekesz
HELY
rekesz
IDŐ
rekesz
CSELEKVŐ
alosztály
rekesz
TEVÉKENYSÉG
SZÁLLÍTÁS
rekesz
OBJEKTUM
CSOMAG
rekesz
IRÁNY
IRÁNY-9
rekesz
HONNAN
rekesz
HOVÁ
IDŐ-65
egyed
rekesz
DÁTUM
MÁJUS 1
rekesz
ÓRA
17:00
ÜGYNÖK-9
egyed
ÜGYFÉL-7
egyed
SZEMÉLY



Esetalapú rendszerek

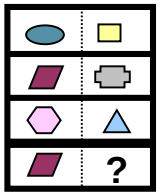
- Cél: Régebbi feladatok megoldásakor szerzett tapasztalatok hasznosítása **hasonló** aktuális feladatok megoldásához.

- Egy **eset** összetevői:
  - A **probléma** leírása
  - A probléma **megoldásának** leírása
  - A megoldás jóságának/rosszaságának minősítése.

- Az eset leírása történhet bármilyen ismeretreprezentációs módszerrel, leggyakoribb a keretalapú szemléltetés.

- A **probléma** leírásánál olyan formalizmust kell alkalmazni, amely olyan **metrikát** értelmez, amely révén az esetek problémaleírásai egymással számszerű eredménnyel összehasonlíthatók (Közelség). (10cm - 20cm; piros színű - narancs színű; szép - gyönyörű. Eltérő adattípusokra nem egyformán könnyű metrikát találni.)

- Az eseteket esetbázisban tároljuk.



Az esetalapú következtetés működése

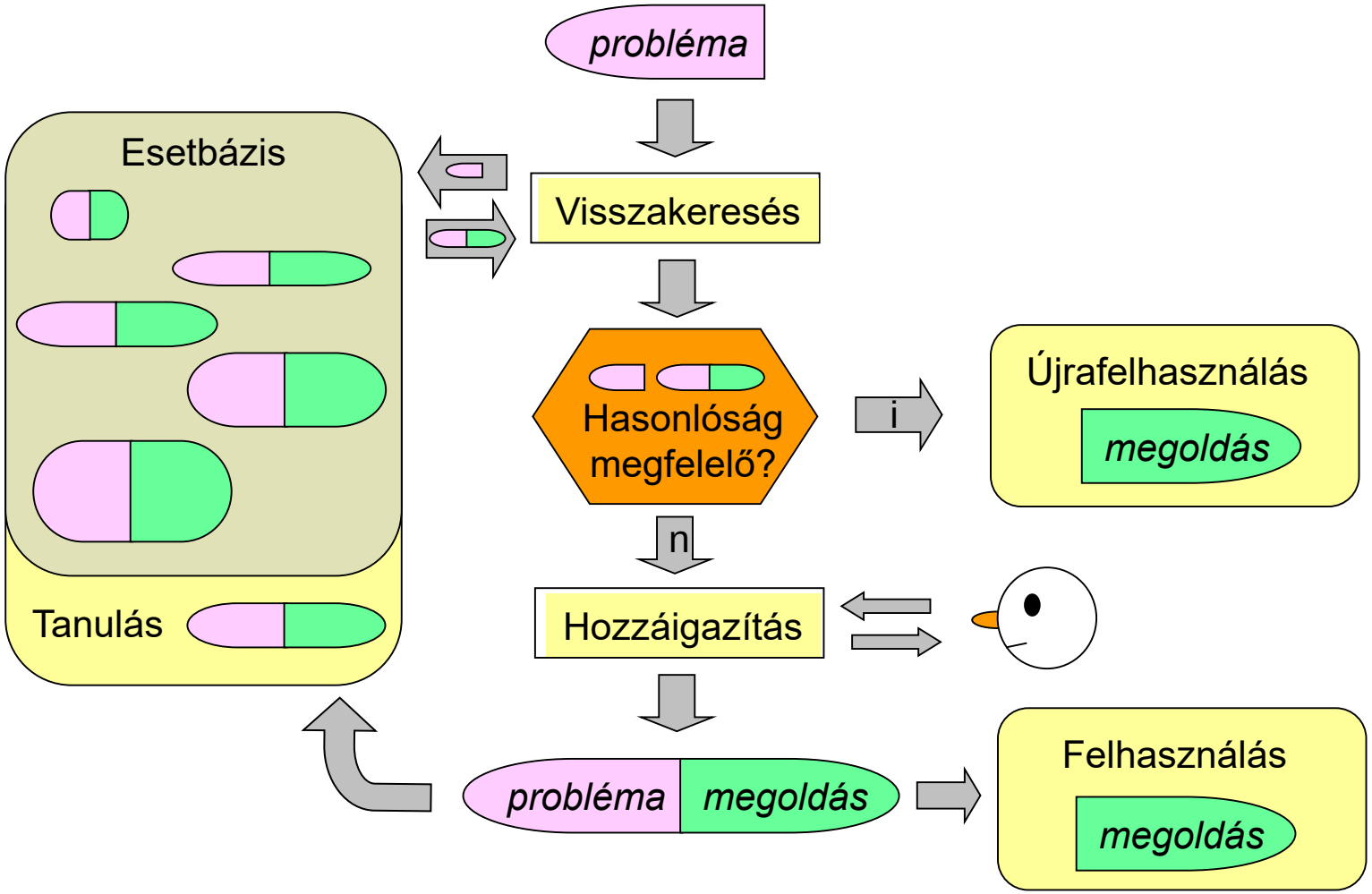
1. **Visszakeresés**: Az esetbázisban megkeressük a megoldandó aktuális problémához legjobban hasonlító, az alkalmazott metrika szerint legközelebbi korábbi problémaleírást.

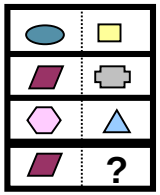
2. **Újrafelhasználás**: amennyiben a hasonlóság egy megadott nagy értéket elér, a korábbi eset megoldását használjuk fel az aktuális probléma megoldására.

3. **Hozzáigazítás**: Amennyiben a legközelebbi eset hasonlósága nem éri el a kívánt szintet, a rendszer **interaktív** módon hozzáigazítja az eset problémaleírását az aktuális problémához, eközben természetesen az eset megoldás oldalát is módosítva. Az ily módon előállt megoldást használjuk fel az aktuális probléma megoldására.

4. **Tanulás**: A 3. pontban előállt hozzáigazított esetet az esetbázishoz adja, a megoldás jóságának/rosszaságának minősítésével együtt.

- **Az esetalapú következtetés működése ..**





Az esetalapú következtetés tulajdonságai

Előnyök:

- A probléma modelljének előzetes kidolgozása nélkül is alkalmazható
- Használat közben fejlődik, könnyen bővíthető
- Robusztus: hiányos, vagy rosszul definiált fogalmakkal is megadhatók esetek
- Nem algoritmizálható problémák esetén is alkalmazható
- Képes támogatni a korábbi hibás megoldások elkerülését is.

Hátrányok:

- Emberi interakciót igényel az esetek többségében
- Minősége romolhat az eltérő felhasználók eltérő igényszintje miatt a tanulás során.