

Prolog 1.

Készítette: Szabó Éva

Prolog

- Logikai, deklaratív nyelv.
- „Egy logikai program egy modellre vonatkoztatott állítások halmaza, melyek a modell tulajdonságait, és az azok között fellépő kapcsolatokat adják meg. Egy adott relációt meghatározó állítások halmazát predikátumnak nevezzük. A predikátumokat **tények** és **szabályok** alkotják, amelyeket a Prologban **ponttal zárunk le**. Hasonlít a magyar nyelvhez, mert az állításokat ott is ponttal zárjuk le.”

SWI-Prolog interpreter

- indítás: ***prolog*** parancs
- fájlok betöltése, újratöltése: ***[fájlnév kiterjesztés nélkül]***.
- Pl. elso.pl fájl betöltése: [elso].
- Kiterjesztés: **.pl**
- egy predikátumhoz tartozó tények és szabályok listája: ***listing(predikátumnév)***.
- nyomkövetés, végrehajtási lépések kiírása: ***trace, cél***.
- A cél egy kérdés, célklóz. Tovább lépés ***enterrel***.
- kilépés: ***halt***.
- kommentezés: egysoros kommentek, % jellel bevezetve

- Miután sikeresen betöltöttük a forrásfájlunkat, megjelenik egy '|?-' prompt, és itt tehetjük fel a Prolognak szánt kérdéseket.

Kérdés lehet...

- **Eldöntendő** kérdést ugyanúgy tehetünk fel, mintha állítást írnánk a kódba. A Prolog tudja, hogy ha ezt a promptba írjuk, akkor azt kérdésként kell kezelni.
- **Általános** kérdéseket szabad változók használatával tudunk feltenni mégpedig úgy, hogy a kérdésben azt jelöljük változóval, amire rá szeretnénk kérdezni.

Példa (halmazok.pl)

reszhalmaz (n, z). % tény

reszhalmaz (z, q). % tény

reszhalmaz (q, r). % tény

reszhalmaz (r, c). % tény

|?- reszhalmaz (q,r). → 'yes'

|?- reszhalmaz (z,n). → 'no'

subset(X,Y) :- reszhalmaz(X,Y). % szabály

subset(X,Y) :- reszhalmaz(X,Z), subset(Z,Y).

subset(n,r).

subset(z,X). % cél

Termek

- **Egyszerű** termék fajtái:
 - **Boolean** – értékei: **true**, **false**
 - **Integer** – egész számok.
 - **Real** – valós számok
 - **Variable** – Változók: nagybetűvel vagy **_**-sal kezdődne, értékadás = -jellel
 - X
 - **_var**
 - Atom – karaktersorozatok (idézőjelek közti vagy kisbetűvel kezdődő szöveg)
- **Összetett** termék
 - Lista – nincsenek indexelve az elemek, rekurzióval fogjuk bejárni.
pl.: [1,2,3,4,5]

Kiértékelés

- Kifejezések kiértékelésére a beépített, infix **is** operátort használhatjuk.
- Általános alakja:
 $\langle \text{szabad változó} \rangle \text{ is } \langle \text{kifejezés} \rangle$
vagy
 $\langle \text{szabad változó} \rangle \text{ is } \langle \text{kötött változó} \rangle$
- Pl.: $X \text{ is } (3+5)$. Kimenete: $X = 8$.

Aritmetikai operátorok

+	<i>összeadás</i>
-	<i>kivonás</i>
*	<i>szorzás</i>
/	<i>törtosztás</i>
//	<i>egészosztás</i>
mod	<i>egész osztás maradéka</i>
**	<i>hatványozás</i>

Az infix operátorok az alábbi alakban is írhatók:

- 3+4 +(3,4)
- 3**4 **(3,4)
- 3 mod 4 mod(3,4)
- stb...

- a logikai ÉS (konjunkció) kapcsolatot a \wedge karakter jelenti.
- A logikai VAGY kapcsolatot pedig a \vee karakterrel jelöljük.

Változók állapotai

Változóknak 3 állapota van:

- szabad
- szabad-megosztott
- kötött

Illeszkedési szabályok

- Kötött változó az értékével vesz részt.
- Konstans kizárólag önmagával illeszkedik.
- Kötött változó illeszkedik a szabad változóval, és a szabad változó is kötétté válik.
- Két szabad változó illeszkedik, és utána mindkettő szabad-megosztott lesz.
- Összetett term akkor illeszkedik, ha a funktor és a résztermek száma megegyezik, és a résztermek is illeszkednek.

$$\text{fgv}(A,3) = \text{fgv}(B,3).$$

Illeszkednek, mert ugyanaz a funktor, mindkét oldalon 2-2 paraméterrel, továbbá a második paraméter mindkét oldalon a konstans 3, amely önmagával illeszkedik, az első paraméter pedig mindkét oldalon szabad változók, amelyek szintén illeszkednek.

$$\text{fgv}(A,A) = \text{fgv}(4,5).$$

Nem illeszkednek.

Az első paraméterek illeszkednek, mert a szabad változó illeszkedik a konstanssal, és utána kötött lesz ($A=4$). De a második paraméter illesztésénél ugyanez lenne a helyzet, de az A ekkor már kötött változó, amely az értékével vesz részt az illesztésben, tehát azt nézi, hogy $4=5$, amely nyilván hamis.

$$f(X, g(Z, h(X, 4))) = f(Y, g(3, h(5, 4))).$$

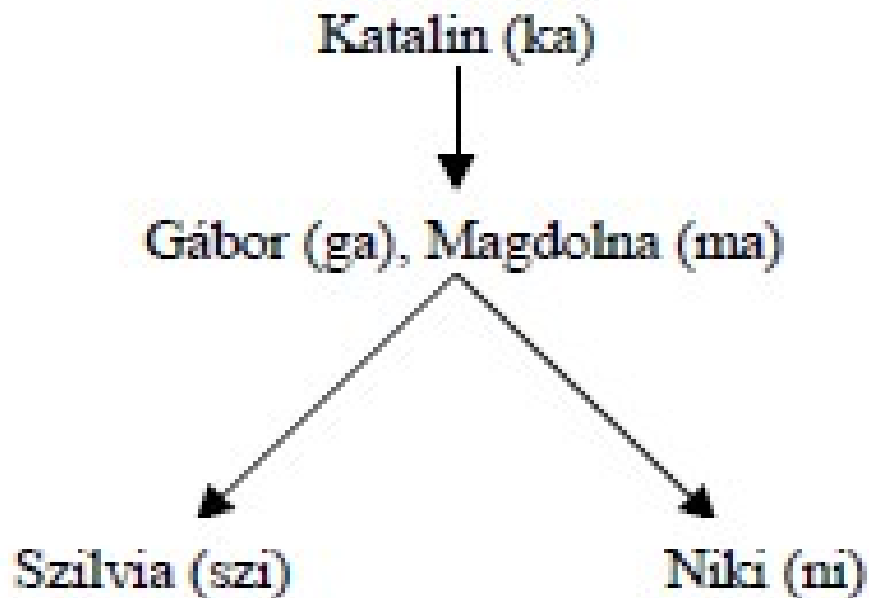
Induljunk el a legbelső zárójelből. Az X illeszkedik az 5-tel, az X kötött lesz ($X=5$), a 4 is illeszkedik a 4-gyel.

Az eggyel kintebbi zárójel első paraméterei illeszkednek, a Z kötött lesz ($Z=3$). A második paraméterek szintén illeszkednek, az előbb már beláttuk.

Most nézzük a legkülső zárójelet. Már beláttuk, hogy a második paraméterek illeszkednek, tehát elegendő az első paramétereket vizsgálni.

Az kell, hogy X illeszkedik-e Y -nal. Könnyen látjuk, hogy igen, mert bár X már kötött, de Y még szabad. Minden részterm illeszkedik és a funktorok is megegyeznek, tehát az egész illeszkedik.

Családfa példa (csfa.pl)



Ennek megfelelően a családot leíró tények halmaza a következő:

```
szulo(ka, ma) .  
szulo(ka, ga) .  
szulo(ga, szi) .  
szulo(ma, szi) .  
szulo(ga, ni) .  
szulo(ma, ni) .
```

Kérdések

- Magdolna szülője-e Szilvinek?

szulo(ma,szi). → 'yes'

- Szilvi szülője-e Magdolnának?

szulo(szi,ma). → 'no'

- Van-e olyan X, akinek Gábor az apja?

szulo(ga,X).

→ $X = \text{szi}$, ha itt **Entert** ütünk, akkor nem listázza tovább a válaszokat. Ha ;-t, akkor a következő alternatívával tér vissza. Nyomjunk ;-t!

→ $X = \text{ni}$

- Katalin nagyszülője-e Szilvinek? (Szabályt kell hozzá definiálnunk!)

$nsz(X,Y) :- szulo(X,Z), szulo(Z,Y).$

- Azaz, X az Y nagyszülője, ha X szülője Z-nek és Z szülője Y-nak.
- Mivel módosítottuk a tényeket (és most már szabályt is) tartalmazó állományunkat, újra kell töltsük, amit a betöltésnél beírt utasítás megismétlésével tehetünk meg. ([csfa].)
- Katalin nagyszülője-e Szilvinek?
 $nsz(ka,szi).$