Úvod
Atomické formuly

1. prednáška · Matematika (4): Logika pre informatikov

Ján Kľuka, Jozef Šiška

Letný semester 2020/2021

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky
Katedra aplikovanej informatiky

Obsah 1. prednášky

Úvod

Úvod

O logike

Čo je logika

Logika je vedná disciplína, ktorá študuje usudzovanie.

Správne, racionálne usudzovanie je základom vedy a inžinierstva.

Vyžaduje rozoznať
správne úsudky z predpokladaných princípov a pozorovania
od chybných úvah a špekulácií.

Správnosť úsudkov, zdá sa, nie je iba vec konvencie a dohody.

Logika skúma, **aké** sú zákonitosti správneho usudzovania
a **prečo** sú zákonitosťami.

Ako logika študuje usudzovanie

Logika má dva hlavné predmety záujmu:

Jazyk zápis pozorovaní, definície pojmov, formulovanie teórií

Syntax pravidlá zápisu tvrdení

Sémantika význam tvrdení

Usudzovanie (inferencia)

odvodzovanie nových logických dôsledkov
z doterajších poznatkov.
Aký má vzťah s jazykom, štruktúrou tvrdení?

Jazyk, poznatky a teórie

Jazyk slúži na formulovanie tvrdení, ktoré vyjadrujú
poznatky o svete (princípy jeho fungovania aj pozorované fakty).

Súboru poznatkov, ktoré považujeme za pravdivé, hovoríme ***teória***.

Príklad 0.1 (Party time!)

Máme troch nových známych — Kim, Jima a Sarah.
Organizujeme párty a P0: chceme na ňu pozvať niekoho z nich.
Od spoločných kamarátov sme sa ale dozvedeli o ich požiadavkách:

P1: Sarah nepôjde na párty, ak pôjde Kim.

P2: Jim pôjde na párty, len ak pôjde Kim.

P3: Sarah nepôjde bez Jima.

Možné stavy sveta a modely

Jedna z otázok, ktoré si o teórii o party môžeme položiť, je:
„V akých zostavách môžu noví známi prísť na párty tak,
aby boli všetky podmienky splnené?“

Priamočiaro (aj keď prácne) to zistíme tak, že:

1. vymenujeme **všetky možné stavy sveta** (účasti nových známych),
2. zistíme, v ktorých sú všetky podmienky splnené.

K	J	S	P0	P1	P2	P3
n	n	n				
n	n	p				
n	p	n				
n	p	p				
p	n	n				
p	n	p				
p	p	n				
p	p	p				

P0: Niekto z Kim, Jima, Sarah príde na párty.

P1: Sarah nepôjde na párty, ak pôjde Kim.

P2: Jim pôjde na párty, len ak pôjde Kim.

P3: Sarah nepôjde bez Jima.

Možné stavy sveta a modely

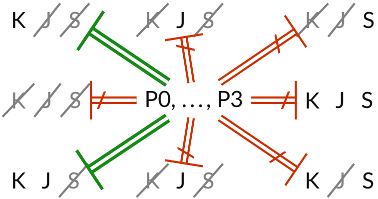
Teória rozdeľuje **možné stavy sveta** (interpretácie) na:

- ⊨ stavy, v ktorých je pravdivá — ***modely*** teórie,
- ⊭ stavy, v ktorých je nepravdivá.

Tvrdenie aj teória môžu mať viacero modelov, ale aj žiaden.

Príklad 0.2

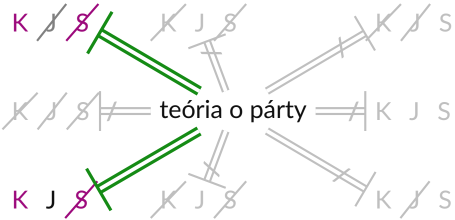
Modelmi teórie P0, P1, P2, P3 sú dve situácie:
keď Kim príde na párty a ostatní noví známi nie,
a keď Kim a Jim prídu na párty a Sarah nie.



Logické dôsledky

Často je zaujímavá iná otázka o teórii —
musí byť nejaké tvrdenie pravdivé **vždy, keď** je pravdivá teória?

V našom príklade:
Kto **musí** a kto **nesmie** prísť na párty,
aby boli podmienky P0, . . . , P3 splnené?



Logické dôsledky

Logickými dôsledkami teórie sú tvrdenia,
ktoré sú pravdivé vo všetkých modeloch teórie.

Príklad 0.3

Logickými dôsledkami teórie P0, P1, P2, P3 sú napríklad:

- Kim príde na párty.
- Sarah nepríde na párty.

Logických dôsledkov je nekonečne veľa, môžu nimi byť ľubovoľne zložité tvrdenia:

- Na party príde Kim alebo Jim.
- Ak príde Sarah, tak príde aj Jim.
- Ak príde Jim, tak nepríde Sarah.

$\vdots$

Logické usudzovanie

Preskúmať všetky stavy sveta je často nepraktické až nemožné.

Logické dôsledky ale môžeme *odvodzovať* **usudzovaním** (*inferovať*).

Pri odvodení vychádzame z ***premís*** (predpokladov)
a postupnosťou **správnych úsudkov** dospievame k ***záverom***.

Príklad 0.4

Vieme, že ak na párty pôjde Kim, tak nepôjde Sarah (P1),
a že ak pôjde Jim, tak pôjde Kim (P2).

1. Predpokladajme, že na párty pôjde Jim.
2. Podľa 1. a P2 pôjde aj Kim.
3. Podľa 2. a P1 nepôjde Sarah.

Teda podľa uvedenej úvahy:
Ak na párty pôjde Jim, tak nepôjde Sarah.

Dedukcia

Úsudok je správny (*korektný*) vtedy, keď **vždy**, keď sú pravdivé jeho premisy, je pravdivý aj jeho záver.

Ak sú všetky úsudky v odvodení správne,
záver **je logickým dôsledkom** premís
a odvodenie je jeho *dôkazom* z premís.

Dedukcia je usudzovanie, pri ktorom sa používajú iba správne úsudky.

Logika študuje dedukciu, ale aj niektoré nededuktívne úsudky,
ktoré sú *vo všeobecnosti* nesprávne,
ale sú správne v *špeciálnych* prípadoch alebo sú *užitočné*:

- indukcia — zovšeobecnenie;
- abdukcia — odvodzovanie možných príčin z následkov;
- usudzovanie na základe analógie (podobnosti).

Kontrapríklady

Ak úsudok nie je správny, vieme nájsť ***kontrapríklad*** —
stav sveta, v ktorom sú predpoklady pravdivé,
ale záver je nepravdivý.

Príklad 0.5

Nesprávny úsudok:
Ak platia tvrdenia teórie o party, na party príde Jim.

Kontrapríklad:
Stav, kedy príde Kim, nepríde Jim, nepríde Sarah.

Teória je pravdivá, výrok „na party príde Jim" nie je pravdivý.

Ťažkosti s prirodzeným jazykom

Prirodzený jazyk je problematický:

- Viacznačné slová: Milo je v posluchárni A.
- Viacznačné tvrdenia: Videl som dievča v sále s ďalekohľadom.
- Ťažko syntakticky analyzovateľné tvrdenia:

  Vlastníci bytov a nebytových priestorov v dome prijímajú rozhodnutia na schôdzi vlastníkov dvojtretinovou väčšinou hlasov všetkých vlastníkov bytov a nebytových priestorov v dome, ak hlasujú o zmluve o úvere a o každom dodatku k nej, o zmluve o zabezpečení úveru a o každom dodatku k nej, o zmluve o nájme a kúpe veci, ktorú vlastníci bytov a nebytových priestorov v dome užívajú s právom jej kúpy po uplynutí dojednaného času užívania a o každom dodatku k nej, o zmluve o vstavbe alebo nadstavbe a o každom dodatku k nim, o zmene účelu užívania spoločných častí domu a spoločných zariadení domu a o zmene formy výkonu správy; …

  — *Zákon č. 182/1993 Z. z. SR v znení neskorších predpisov*
- Výnimky a obraty so špeciálnym ustáleným významom:
  Nikto nie je dokonalý.

Formálne jazyky

Problémy prirodzených jazykov sa obchádzajú použitím umelých **formálnych** jazykov.

- Presne definovaná, zjednodušená syntax (pravidlá zápisu tvrdení) a sémantika (význam).
- Niekoľko formálnych jazykov už poznáte: aritmetika, jazyky fyzikálnych a chemických vzorcov, programovacie jazyky, …
- Problémy z reálneho sveta opísané v prirodzenom jazyku musíme najprv **formalizovať**, a potom naň môžeme použiť logický aparát.
- Formalizácia vyžaduje cvik, trocha veda, trocha umenie.

Formalizácia poznatkov

S formalizáciou ste sa už stretli — napríklad pri riešení slovných úloh:

Karol je trikrát starší ako Mária.
Súčet Karolovho a Máriinho veku je 12 rokov. $\rightsquigarrow$
Koľko rokov majú Karol a Mária?

$$k = 3 \cdot m$$
$$k + m = 12$$

Stretli ste sa už aj s formálnym jazykom výrokovej logiky.

Príklad 0.6

Sformalizujme náš párty príklad:

P0: Niekto z trojice Kim, Jim, Sarah pôjde na párty.

P1: Sarah nepôjde na párty, ak pôjde Kim.

P2: Jim pôjde na párty, len ak pôjde Kim.

P3: Sarah nepôjde bez Jima.

Jazyk logiky prvého rádu (FOL) je jeden zo základných formálnych jazykov, ktorým sa logika zaoberá.

Do dnešnej podoby sa vyvinul koncom 19. a v prvej polovici 20. storočia — Gottlob Frege, Guiseppe Peano, Charles Sanders Peirce.

Výrokové spojky + **kvantifikátory** $\forall$ a $\exists$.

Dá sa v ňom vyjadriť veľa zaujímavých tvrdení, bežne sa používa v matematike.

$$\forall \varepsilon > 0 \, \exists \delta > 0 \ldots$$

Logika prvého rádu a informatika

Informatika sa vyvinula z logiky
(John von Neumann, Alan Turing, Alonzo Church, …)

Prvky logiky prvého rádu obsahuje väčšina **programovacích jazykov**:

- `all(x > m for x in arr)`,
- `select T1.x, T2.y from T1 inner join T2 on T1.z = T2.z where T1.z > 25`,

niektoré (Prolog) sú priamo podmnožinou FOL.

Vo FOL sa dá **presne špecifikovať**, čo má program robiť,
popísať, čo robí, a **dokázať**, že robí to, čo bolo špecifikované.

Vo **výpočtovej logike** a umelej inteligencii sa FOL používa na riešenie rôznych ťažkých problémov (plánovanie, rozvrh, hľadanie a overovanie dôkazov matematických tvrdení, …)
simulovaním usudzovania.

Kalkuly — formalizácia usudzovania

Pre mnohé logické jazyky sú známe **kalkuly** —
množiny usudzovacích pravidiel, ktoré sú

korektné — odvodzujú iba logické dôsledky,
úplné — umožňujú odvodiť všetky logické dôsledky.

Kalkuly sú bežné v matematike

- na počítanie s číslami, zlomkami (kalkul elementárnej aritmetiky),
- riešenie lineárnych rovníc (kalkul lineárnej algebry),
- derivovanie, integrovanie, riešenie diferenciálnych rovníc (kalkul matematickej analýzy)

$\vdots$

Sú korektné, ale nie vždy úplné.

Poznáte už aj jeden logický kalkul — ekvivalentné úpravy.

Úvod

O kurze

Čím sa budeme zaoberať v tomto kurze

Teoreticky

- Jazykmi logiky prvého rádu (FOL), jeho syntaxou a sémantikou
- Správnymi úsudkami v ňom a dôvodmi ich správnosti
- Korektnosťou a úplnosťou logických kalkulov
- Automatizáciou usudzovania

Prakticky

- Vyjadrovaním problémov vo FOL
- Dokazovaním konkrétnych logických dôsledkov
- Automatizovaním riešenia problémov
- Manipuláciou symbolických stromových štruktúr (výrazov — formúl a termov)
- Programovaním vlastných jednoduchých automatických dokazovačov

Filozoficky

- Zamýšľanými a nezamýšľanými významami tvrdení
- Obmedzeniami vyjadrovania a usudzovania

Prístup k logike na tomto predmete

Stredoškolský prístup príliš **neoddeľuje** *jazyk* výrokov od jeho *významu* a vlastne ani jednu stránku **nedefinuje jasne**.

V tomto kurze sa budeme snažiť byť **presní**.

- *Zdanlivo* budeme o jednoduchých veciach hovoriť zložito

Pojmy z logiky budeme **definovať matematicky**

- ako množiny, postupnosti, funkcie, atď., ←-- Matematika (1), (3)

na praktických cvičeniach aj **programami**

- ako reťazce, slovníky, triedy a metódy. ←-- Programovanie (1), (2)

Budeme sa pokúšať **dokazovať** ich vlastnosti.

Budeme teda hovoriť **o formálnej logike** pomocou matematiky — *meta* matematika logiky, matematika **o** logike.

Organizácia kurzu — rozvrh, kontakty, pravidlá

Organizácia predmetu — rozvrh, kontakty a pravidlá absolvovania — sú popísané na oficiálnej webovej stránke predmetu:

https://dai.fmph.uniba.sk/w/Course:Mathematics_4

Atomické formuly

Jazyky logiky prvého rádu

Logika prvého rádu je trieda (rodina) formálnych jazykov.

Zdieľajú:

- časti abecedy — **logické symboly** (spojky, kvantifikátory)
- pravidlá tvorby **formúl** (slov)

Líšia sa v **mimologických symboloch** — časť abecedy, pomocou ktorej sa tvoria najjednoduchšie — **atomické formuly** (**atómy**).

Atomické formuly a výroky v prirodzenom jazyku

Atomické formuly logiky prvého rádu zodpovedajú **pozitívnym jednoduchým vetám** o vlastnostiach, stavoch, vzťahoch a rovnosti **jednotlivých pomenovaných** objektov.

Príklady 1.1

- Milo beží.
- Jarka vidí Mila.
- Milo beží, ale Jarka ho nevidí.
- Jarka vidí všetkých.
- Jarka dala Milovi Bobíka v sobotu.
- Jarka nie je doma.
- Niekto je doma.
- Súčet 2 a 2 je 3.
- Prezidentkou SR je Zuzana Čaputová.

Atomické formuly a výroky v prirodzenom jazyku

Atomické formuly logiky prvého rádu zodpovedajú **pozitívnym jednoduchým vetám** o vlastnostiach, stavoch, vzťahoch a rovnosti **jednotlivých pomenovaných** objektov.

Príklady 1.1

- ✓ Milo beží.
- ✓ Jarka vidí Mila.
- ✗ Milo beží, ale Jarka ho nevidí.
- ✗ Jarka vidí všetkých.
- ✓ Jarka dala Milovi Bobíka v sobotu.
- ✗ Jarka nie je doma.
- ✗ Niekto je doma.
- ✓ Súčet 2 a 2 je 3.
- ✓ Prezidentkou SR je Zuzana Čaputová.

Indivíduové konštanty

Indivíduové konštanty sú symboly jazyka logiky prvého rádu, ktoré pomenúvajú jednotlivé, pevne zvolené objekty.

Zodpovedajú *približne* vlastným menám, jednoznačným pomenovaniam, niekedy zámenám; konštantám v matematike a programovacích jazykoch.

Príklady 1.2

$\mathtt{Jarka}$, 2, $\mathtt{Zuzana\_Čaputová}$, $\mathtt{sobota}$, π, $\ldots$

Indivíduové konštanty a objekty

Indivíduová konštanta

- vždy pomenúva skutočný, existujúci objekt

  (na rozdiel od vlastného mena *Zeus*);
- nikdy nepomenúva viac objektov

  (na rozdiel od vlastného mena *Jarka*).

Objekt

- môže byť pomenovaný aj viacerými indivíduovými konštantami

  (napr. `Prezidentka_SR` a `Zuzana_Čaputová`);
- nemusí mať žiadne meno.

Predikátové symboly

Predikátové symboly sú symboly jazyka logiky prvého rádu, ktoré vyjadrujú vlastnosti alebo vzťahy.

Jednoduché vety v slovenčine majú *podmetovú* (*subjekt*) a *prísudkovú* časť (*predikát*):

Jarka	vidí	Mila.
podmet	prísudok	predmet
podmetová časť	prísudková časť	

Do logiky prvého rádu prekladáme takéto tvrdenie pomocou predikátového symbolu `vidí`, ktorý má dva ***argumenty*** („podmety“): indivíduové konštanty `Jarka` a `Milo`.

Úloha argumentu v predikáte je daná jeho poradím (podobne ako pozičné argumenty funkcií/metód v prog. jazykoch).

Arita predikátového symbolu

Predikátový symbol má pevne určený počet argumentov — ***aritu***.

Vždy musí mať práve toľko argumentov, aká je jeho arita.

Dohoda 1.3

Aritu budeme *niekedy* písať ako horný index symbolu.
Napríklad $\mathtt{beží}^1$, $\mathtt{vidí}^2$, $\mathtt{dal}^4$, $<^2$.

Zamýšľaný význam predikátových symbolov

Unárny predikátový symbol (teda s aritou 1) zvyčajne označuje vlastnosť, druh, rolu, stav.

Príklady 1.4

$\texttt{pes}(x)$	x je pes
$\texttt{čierne}(x)$	x je čierne
$\texttt{beží}(x)$	x beží

Binárny, *ternárny*, … predikátový symbol (s aritou 2, 3, …) zvyčajne označuje vzťah svojich argumentov.

Príklady 1.5

$\texttt{vidí}(x, y)$	x vidí y
$\texttt{dal}(x, y, z, t)$	x dal(a/o) objektu y objekt z v čase t

Kategorickosť významu predikátových symbolov

V bežnom jazyku často nie je celkom jasné, či objekt má alebo nemá nejakú vlastnosť — kedy je niekto *mladý*?

Predikátové symboly predstavujú ***kategorické*** vlastnosti/vzťahy — pre každý objekt sa dá **jednoznačne rozhodnúť**, či má alebo nemá túto vlastnosť/vzťah s iným objektom či inými objektmi.

Význam predikátového symbolu preto často zodpovedá rovnakému slovenskému predikátu iba približne.

Príklad 1.6

Predikát $\mathtt{mladší}^2$ môže označovať vzťah „x je mladší ako y“ presne.

Predikát $\mathtt{mladý}^1$ zodpovedá vlastnosti „x je mladý“ iba približne.

Nekategorickými vlastnosťami sa zaoberajú *fuzzy* logiky.
Predikáty v nich zachytávajú význam týchto vlastností presnejšie.

Atomické formuly

Atomické formuly majú tvar

$$predik\acute{a}t(argument_1, argument_2, \dots, argument_k),$$

alebo

$$argument_1 \doteq argument_2,$$

pričom k je arita *predikát*u,
a $argument_1, \dots, argument_k$ sú (nateraz) indivíduové konštanty.

Atomická formula zodpovedá (jednoduchému) výroku v slovenčine,
t.j. tvrdeniu, ktorého pravdivostná hodnota (pravda alebo nepravda)
sa dá jednoznačne určiť,
lebo predikát označuje kategorickú vlastnosť/vzťah

a indivíduové konštanty jednoznačne označujú objekty.

Formalizácia jednoduchých výrokov

Formalizácia je preklad výrokov z prirodzeného jazyka do formálneho logického jazyka.

Nie je to jednoznačný proces.

Vopred daný prvorádový jazyk (konštanty a predikáty) sa snažíme využiť čo najlepšie.

Príklad 1.7

Sformalizujme v jazyku s konštantami `Evka`, `Jarka` a `Milo` a predikátom $\texttt{vyšší}^2$ výroky:

A_1: Jarka je vyššia ako Milo.

A_2: Evka je nižšia ako Milo.

Formalizácia jednoduchých výrokov

Formalizácia je preklad výrokov z prirodzeného jazyka do formálneho logického jazyka.

Nie je to jednoznačný proces.

Vopred daný prvorádový jazyk (konštanty a predikáty) sa snažíme využiť čo najlepšie.

Príklad 1.7

Sformalizujme v jazyku s konštantami $\mathtt{Evka}$, $\mathtt{Jarka}$ a $\mathtt{Milo}$ a predikátom $\mathtt{vyšší}^2$ výroky:

A_1: Jarka je vyššia ako Milo. $\rightsquigarrow$ $\mathtt{vyšší}(\mathtt{Jarka}, \mathtt{Milo})$

A_2: Evka je nižšia ako Milo. $\rightsquigarrow$ $\mathtt{vyšší}(\mathtt{Milo}, \mathtt{Evka})$

Zanedbávame nepodstatné detaily — pomocné slovesá, predložky, skloňovanie, rod, …: x je vyšší/vyššia/vyššie ako y $\rightsquigarrow$ $\mathtt{vyšší}(x, y)$.

Formalizácia spojená s **návrhom vlastného jazyka** je **iteratívna**:
Postupne zisťujeme, aké predikáty a konštanty potrebujeme,
upravujeme predchádzajúce formalizácie.

Príklady 1.8

A_1: Jarka dala Milovi Bobíka.

Návrh jazyka pri formalizácii

Formalizácia spojená s **návrhom vlastného jazyka** je **iteratívna**: Postupne zisťujeme, aké predikáty a konštanty potrebujeme, upravujeme predchádzajúce formalizácie.

Príklady 1.8

A_1: Jarka dala Milovi Bobíka.

$\rightsquigarrow$ `dalaMiloviBobíka(Jarka)`

Návrh jazyka pri formalizácii

Formalizácia spojená s **návrhom vlastného jazyka** je **iteratívna**:
Postupne zisťujeme, aké predikáty a konštanty potrebujeme, upravujeme predchádzajúce formalizácie.

Príklady 1.8

A_1: Jarka dala Milovi Bobíka.

$\rightsquigarrow$ `dalaMiloviBobíka(Jarka)`

A_2: Evka dostala Bobíka od Mila.

Návrh jazyka pri formalizácii

Formalizácia spojená s **návrhom vlastného jazyka** je **iteratívna**:
Postupne zisťujeme, aké predikáty a konštanty potrebujeme,
upravujeme predchádzajúce formalizácie.

Príklady 1.8

A_1: Jarka dala Milovi Bobíka.

$\rightsquigarrow$ $\cancel{\mathtt{dalaMiloviBobíka}(\mathtt{Jarka})}$ $\mathtt{dalBobíka}(\mathtt{Jarka}, \mathtt{Milo})$

A_2: Evka dostala Bobíka od Mila.

$\rightsquigarrow$ $\mathtt{dalBobíka}(\mathtt{Milo}, \mathtt{Evka})$

Návrh jazyka pri formalizácii

Formalizácia spojená s **návrhom vlastného jazyka** je **iteratívna**: Postupne zisťujeme, aké predikáty a konštanty potrebujeme, upravujeme predchádzajúce formalizácie.

Príklady 1.8

A_1: Jarka dala Milovi Bobíka.

$\rightsquigarrow$ $\cancel{\texttt{dalaMiloviBobíka}(\texttt{Jarka})}$ $\texttt{dalBobíka}(\texttt{Jarka}, \texttt{Milo})$

A_2: Evka dostala Bobíka od Mila.

$\rightsquigarrow$ $\texttt{dalBobíka}(\texttt{Milo}, \texttt{Evka})$

A_3: Evka dala Jarke Cilku.

Návrh jazyka pri formalizácii

Formalizácia spojená s **návrhom vlastného jazyka** je **iteratívna**:
Postupne zisťujeme, aké predikáty a konštanty potrebujeme,
upravujeme predchádzajúce formalizácie.

Príklady 1.8

A_1: Jarka dala Milovi Bobíka.

$\rightsquigarrow$ $\cancel{\mathtt{dalaMiloviBobíka}(\mathtt{Jarka})}$ $\mathtt{dalBobíka}(\mathtt{Jarka}, \mathtt{Milo})$

A_2: Evka dostala Bobíka od Mila.

$\rightsquigarrow$ $\mathtt{dalBobíka}(\mathtt{Milo}, \mathtt{Evka})$

A_3: Evka dala Jarke Cilku.

$\rightsquigarrow$ $\mathtt{dalCilku}(\mathtt{Evka}, \mathtt{Jarka})$

Návrh jazyka pri formalizácii

Formalizácia spojená s **návrhom vlastného jazyka** je **iteratívna**:
Postupne zisťujeme, aké predikáty a konštanty potrebujeme, upravujeme predchádzajúce formalizácie.

Príklady 1.8

A_1: Jarka dala Milovi Bobíka.

$\rightsquigarrow$ $\cancel{\mathtt{dalaMiloviBobíka}(\mathtt{Jarka})}$ $\cancel{\mathtt{dalBobíka}(\mathtt{Jarka}, \mathtt{Milo})}$
$\mathtt{dal}(\mathtt{Jarka}, \mathtt{Milo}, \mathtt{Bobík})$

A_2: Evka dostala Bobíka od Mila.

$\rightsquigarrow$ $\cancel{\mathtt{dalBobíka}(\mathtt{Milo}, \mathtt{Evka})}$ $\mathtt{dal}(\mathtt{Milo}, \mathtt{Evka}, \mathtt{Bobík})$

A_3: Evka dala Jarke Cilku.

$\rightsquigarrow$ $\cancel{\mathtt{dalCilku}(\mathtt{Evka}, \mathtt{Jarka})}$ $\mathtt{dal}(\mathtt{Evka}, \mathtt{Jarka}, \mathtt{Cilka})$

A_4: Bobík je pes.

Návrh jazyka pri formalizácii

Formalizácia spojená s **návrhom vlastného jazyka** je **iteratívna**:
Postupne zisťujeme, aké predikáty a konštanty potrebujeme,
upravujeme predchádzajúce formalizácie.

Príklady 1.8

A_1: Jarka dala Milovi Bobíka.

⇝ ~~dalaMiloviBobíka(Jarka)~~ ~~dalBobíka(Jarka, Milo)~~
dal(Jarka, Milo, Bobík)

A_2: Evka dostala Bobíka od Mila.

⇝ ~~dalBobíka(Milo, Evka)~~ dal(Milo, Evka, Bobík)

A_3: Evka dala Jarke Cilku.

⇝ ~~dalCilku(Evka, Jarka)~~ dal(Evka, Jarka, Cilka)

A_4: Bobík je pes.

⇝ pes(Bobík)

Návrh jazyka pri formalizácii

Minimalizujeme počet predikátov, uprednostňujeme flexibilnejšie, viacúčelovejšie ($\texttt{dal}^3$ pred $\texttt{dalBobíka}^2$ a $\texttt{dalCilku}^2$).

Dosiahneme

- expresívnejší jazyk (vyjadrí viac menším počtom prostriedkov),
- zrejmejšie logické vzťahy výrokov.

Podobné normalizácii databázových schém.

Atomické formuly

Syntax atomických formúl

Presné definície

Cieľom logiky je uvažovať o jazyku, výrokoch, vyplývaní, dôkazoch.

Výpočtová logika sa snaží automaticky riešiť konkrétne problémy vyjadrené v logických jazykoch.

Spoľahlivé a overiteľné úvahy a výpočty vyžadujú
presnú dohodu na tom, o čom hovoríme —
definíciu logických pojmov (jazyk, výrok, pravdivosť, $\ldots$).

Pojmy (napr. *atomická formula*) môžeme zadefinovať napríklad

- matematicky ako množiny, n-tice, relácie, funkcie, postupnosti, $\ldots$;
- informaticky tým, že ich naprogramujeme, napr. zadefinujeme triedu `AtomickaFormula` v Pythone.

Matematický jazyk je univerzálnejší ako programovací — abstraktnejší, menej nie až tak podstatných detailov.

Syntax atomických formúl logiky prvého rádu

Najprv sa musíme dohodnúť na tom,
aká je syntax atomických formúl logiky prvého rádu:

- z čoho sa skladajú,
- čím vlastne sú,
- akú majú štruktúru.

Symboly jazyka atomických formúl logiky prvého rádu

Z čoho sa skladajú atomické formuly?

Symboly jazyka atomických formúl logiky prvého rádu

Z čoho sa skladajú atomické formuly?

Definícia 1.9

Symbolmi jazyka $\mathcal{L}$ atomických formúl logiky prvého rádu sú mimologické, logické a pomocné symboly, pričom:

Mimologickými symbolmi sú

- *indivíduové konštanty* z nejakej neprázdnej spočítateľnej množiny $\mathcal{C}_\mathcal{L}$
- a *predikátové symboly* z nejakej spočítateľnej množiny $\mathcal{P}_\mathcal{L}$.

Jediným *logickým symbolom* je $\doteq$ (symbol rovnosti).

Pomocnými symbolmi sú **(**, **)** a **,** (ľavá, pravá zátvorka a čiarka).

Množiny $\mathcal{C}_\mathcal{L}$ a $\mathcal{P}_\mathcal{L}$ sú disjunktné.
Pomocné symboly sa nevyskytujú v symboloch z $\mathcal{C}_\mathcal{L}$ ani $\mathcal{P}_\mathcal{L}$.
Každému symbolu $P \in \mathcal{P}_\mathcal{L}$ je priradená *arita* $\mathrm{ar}_\mathcal{L}(P) \in \mathbb{N}^+$.

Abeceda jazyka atomických formúl logiky prvého rádu

Na Úvode do teoretickej informatiky by ste povedali,
že *abecedou* jazyka $\mathcal{L}$ atomických formúl logiky prvého rádu
je $\Sigma_{\mathcal{L}} = \mathcal{C}_{\mathcal{L}} \cup \mathcal{P}_{\mathcal{L}} \cup \{\doteq, \mathbf{(}, \mathbf{)}, \mathbf{,}\}$.

V logike sa väčšinou pojem *abeceda* nepoužíva, pretože potrebujeme rozlišovať rôzne druhy symbolov.

Namiesto *abeceda jazyka* $\mathcal{L}$ hovoríme *množina všetkých symbolov jazyka* $\mathcal{L}$ alebo len *symboly jazyka* $\mathcal{L}$.

Na zápise množiny $\Sigma_{\mathcal{L}}$ však ľahko vidíme,
čím sa rôzne jazyky atomických formúl logiky prvého rádu
od seba líšia a čo majú spoločné.

Príklady symbolov jazykov atomických formúl logiky prvého rádu

Príklad 1.10

Príklad o deťoch a zvieratkách sme sformalizovali v jazyku $\mathcal{L}_{\text{dz}}$, v ktorom

$$\mathcal{C}_{\mathcal{L}_{\text{dz}}} = \{\texttt{Bobík}, \texttt{Cilka}, \texttt{Evka}, \texttt{Jarka}, \texttt{Milo}\},$$
$$\mathcal{P}_{\mathcal{L}_{\text{dz}}} = \{\texttt{dal}, \texttt{pes}\}, \quad \mathrm{ar}_{\mathcal{L}_{\text{dz}}}(\texttt{dal}) = 3, \quad \mathrm{ar}_{\mathcal{L}_{\text{dz}}}(\texttt{pes}) = 1.$$

Príklad 1.11

Príklad o návštevníkoch party by sme mohli sformalizovať v jazyku $\mathcal{L}_{\text{party}}$, kde

$$\mathcal{C}_{\mathcal{L}_{\text{party}}} = \{\texttt{Kim}, \texttt{Jim}, \texttt{Sarah}\},$$
$$\mathcal{P}_{\mathcal{L}_{\text{party}}} = \{\texttt{príde}\}, \quad \mathrm{ar}_{\mathcal{L}_{\text{party}}}(\texttt{príde}) = 1.$$

Označenia symbolov

Keď budeme hovoriť o ľubovoľnom jazyku $\mathcal{L}$, často budeme potrebovať nejak označiť niektoré jeho konštanty alebo predikáty, aj keď nebudeme vedieť, aké konkrétne symboly to sú.

Na označenie symbolov použijeme ***meta premenné***:
premenné v (matematickej) slovenčine, pomocou ktorých budeme hovoriť o (po grécky *meta*) týchto symboloch.

Dohoda 1.12

Indivíduové konštanty budeme spravidla označovať meta premennými a, b, c, d s prípadnými dolnými indexmi.

Predikátové symboly budeme spravidla označovať meta premennými P, Q, R s prípadnými dolnými indexmi.

Atomické formuly jazyka

Čo sú atomické formuly?

Atomické formuly jazyka

Čo sú atomické formuly?

Definícia 1.13

Nech $\mathcal{L}$ je jazyk atomických formúl logiky prvého rádu.

Rovnostný atóm jazyka $\mathcal{L}$ je každá postupnosť symbolov $c_1 \doteq c_2$, kde c_1 a c_2 sú indivíduové konštanty z $\mathcal{C}_{\mathcal{L}}$.

Predikátový atóm jazyka $\mathcal{L}$ je každá postupnosť symbolov $P(c_1, \dots, c_n)$, kde P je predikátový symbol z $\mathcal{P}_{\mathcal{L}}$ s aritou n a $c_1, \dots, c_n$ sú indivíduové konštanty z $\mathcal{C}_{\mathcal{L}}$.

Atomickými formulami (skrátene *atómami*) jazyka $\mathcal{L}$ súhrnne nazývame všetky rovnostné a predikátové atómy jazyka $\mathcal{L}$.

Množinu všetkých atómov jazyka $\mathcal{L}$ označujeme $\mathcal{A}_{\mathcal{L}}$.

Slová jazyka atomických formúl logiky prvého rádu

Na Úvode do teoretickej informatiky by ste povedali,
že jazyk $\mathcal{L}$ atomických formúl logiky prvého rádu nad abecedou
$\Sigma_{\mathcal{L}} = \mathcal{C}_{\mathcal{L}} \cup \mathcal{P}_{\mathcal{L}} \cup \{\doteq, \textbf{(}, \textbf{)}, \textbf{,}\}$ je množina slov

$$\{\, c_1 \doteq c_2 \mid c_1 \in \mathcal{C}_{\mathcal{L}}, c_2 \in \mathcal{C}_{\mathcal{L}} \,\} \\ \cup \{\, P(c_1, \dots, c_n) \mid P \in \mathcal{P}_{\mathcal{L}}, \operatorname{ar}_{\mathcal{L}}(P) = n, c_1 \in \mathcal{C}_{\mathcal{L}}, \dots, c_n \in \mathcal{C}_{\mathcal{L}} \,\}.$$

V logike sa jazyk takto nedefinuje, pretože potrebujeme rozlišovať
rôzne druhy slov.

Príklady atómov jazyka

Príklad 1.14

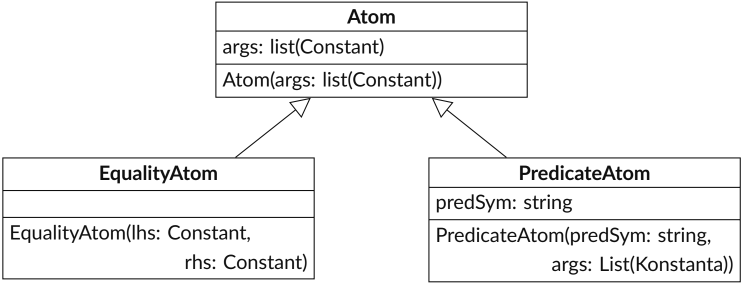
V jazyku $\mathcal{L}_{\text{dz}}$, kde $\mathcal{C}_{\mathcal{L}_{\text{dz}}} = \{\mathtt{Bobík}, \mathtt{Cilka}, \mathtt{Evka}, \mathtt{Jarka}, \mathtt{Milo}\}$, $\mathcal{P}_{\mathcal{L}_{\text{dz}}} = \{\mathtt{dal}, \mathtt{pes}\}$, $\mathrm{ar}_{\mathcal{L}_{\text{dz}}}(\mathtt{dal}) = 3$, $\mathrm{ar}_{\mathcal{L}_{\text{dz}}}(\mathtt{pes}) = 1$, sú *okrem iných* rovnostné atómy:

$$\mathtt{Bobík} \doteq \mathtt{Bobík} \qquad \mathtt{Cilka} \doteq \mathtt{Bobík}$$
$$\mathtt{Evka} \doteq \mathtt{Jarka} \qquad \mathtt{Bobík} \doteq \mathtt{Cilka}$$

a predikátové atómy:

$$\mathtt{pes}(\mathtt{Cilka}) \quad \mathtt{dal}(\mathtt{Cilka}, \mathtt{Milo}, \mathtt{Bobík}) \quad \mathtt{dal}(\mathtt{Jarka}, \mathtt{Evka}, \mathtt{Milo}).$$

Atómy ako triedy



Atomické formuly

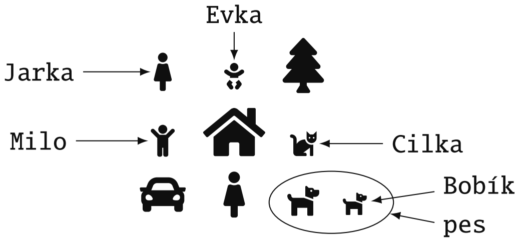
Sémantika atomických formúl

Vyhodnotenie atomickej formuly

Ako zistíme, či je atomická formula `pes(Bobík)` pravdivá v nejakej situácii (napríklad u babky Evky, Jarky a Mila na dedine)?

Pozrieme sa na túto situáciu a zistíme:

1. aký objekt b pomenúva konštanta `Bobík`;
2. akú vlastnosť p označuje predikát `pes`;
3. či objekt b má vlastnosť p.



Vyhodnotenie atomickej formuly

Ako môžeme tento postup matematicky alebo informaticky modelovať?

Potrebujeme:

- matematický/informatický model situácie
  (stavu vybranej časti sveta),
- postup na jeho použitie pri vyhodnocovaní pravdivosti formúl.

Matematický model stavu sveta

Ako môžeme matematicky popísať nejakú situáciu tak, aby sme pomocou tohto popisu mohli vyhodnocovať atomické formuly v nejakom jazyku logiky prvého rádu $\mathcal{L}$?

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,
- množina všetkých týchto objektov — **doména**;

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,
- množina všetkých týchto objektov — **doména**;
- jednoznačné priradenie významu všetkým indivíduovým konštantám a predikátom z jazyka $\mathcal{L}$

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,
  - množina všetkých týchto objektov — **doména**;
- jednoznačné priradenie významu všetkým indivíduovým konštantám a predikátom z jazyka $\mathcal{L}$
  - **interpretačná funkcia**;

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,
- ▶ množina všetkých týchto objektov — **doména**;
- jednoznačné priradenie významu všetkým indivíduovým konštantám a predikátom z jazyka $\mathcal{L}$
- ▶ **interpretačná funkcia**;
- pre každú indivíduovú konštantu c z jazyka $\mathcal{L}$, ktorý **objekt** z domény konštanta c pomenúva,

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,
- ▶ množina všetkých týchto objektov — **doména**;
- jednoznačné priradenie významu všetkým indivíduovým konštantám a predikátom z jazyka $\mathcal{L}$
- ▶ **interpretačná funkcia**;
- pre každú indivíduovú konštantu c z jazyka $\mathcal{L}$, ktorý **objekt** z domény konštanta c pomenúva,
- pre každý unárny predikát P z jazyka $\mathcal{L}$, ktoré objekty z domény majú vlastnosť označenú predikátom P,

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,
- ▶ množina všetkých týchto objektov — **doména**;
- jednoznačné priradenie významu všetkým indivíduovým konštantám a predikátom z jazyka $\mathcal{L}$
- ▶ **interpretačná funkcia**;
- pre každú indivíduovú konštantu c z jazyka $\mathcal{L}$, ktorý **objekt** z domény konštanta c pomenúva,
- pre každý unárny predikát P z jazyka $\mathcal{L}$, ktoré objekty z domény majú vlastnosť označenú predikátom P,
- ▶ tvoria **podmnožinu** domény;

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,
- ▶ množina všetkých týchto objektov — **doména**;
- jednoznačné priradenie významu všetkým indivíduovým konštantám a predikátom z jazyka $\mathcal{L}$
- ▶ **interpretačná funkcia**;
- pre každú indivíduovú konštantu c z jazyka $\mathcal{L}$, ktorý **objekt** z domény konštanta c pomenúva,
- pre každý unárny predikát P z jazyka $\mathcal{L}$, ktoré objekty z domény majú vlastnosť označenú predikátom P,
- ▶ tvoria **podmnožinu** domény;
- pre každý n-árny predikát R z jazyka $\mathcal{L}$, $n > 1$, ktoré n-tice objektov z domény sú vo vzťahu ozn. pred. R,

Matematický model stavu sveta

Potrebujeme vedieť:

- ktoré objekty sú v popisovanej situácii prítomné,
- ▶ množina všetkých týchto objektov — **doména**;
- jednoznačné priradenie významu všetkým indivíduovým konštantám a predikátom z jazyka $\mathcal{L}$
- ▶ **interpretačná funkcia**;
- pre každú indivíduovú konštantu c z jazyka $\mathcal{L}$, ktorý **objekt** z domény konštanta c pomenúva,
- pre každý unárny predikát P z jazyka $\mathcal{L}$, ktoré objekty z domény majú vlastnosť označenú predikátom P,
- ▶ tvoria **podmnožinu** domény;
- pre každý n-árny predikát R z jazyka $\mathcal{L}$, $n > 1$, ktoré n-tice objektov z domény sú vo vzťahu ozn. pred. R,
- ▶ tvoria **n-árnu reláciu** na doméne.

Štruktúra pre jazyk

Definícia 1.15

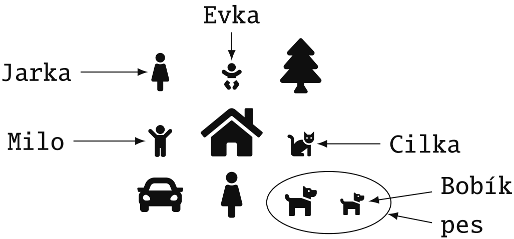
Nech $\mathcal{L}$ je jazyk atomických formúl logiky prvého rádu.
Štruktúrou pre jazyk $\mathcal{L}$ nazývame dvojicu $\mathcal{M} = (D, i)$, kde
D je ľubovoľná **neprázdna** množina nazývaná ***doména*** štruktúry $\mathcal{M}$;
i je zobrazenie, nazývané ***interpretačná funkcia*** štruktúry $\mathcal{M}$, ktoré

- každej indivíduovej konštante c jazyka $\mathcal{L}$
  priraďuje prvok $i(c) \in D$;
- každému predikátovému symbolu P jazyka $\mathcal{L}$ s aritou n
  priraďuje množinu $i(P) \subseteq D^n$.

Dohoda 1.16

Štruktúry označujeme veľkými *písanými* písmenami $\mathcal{M}, \mathcal{N}, \ldots$.

Príklad štruktúry



Príklad 1.17

Štruktúra ako informatický objekt

Štruktúru sme definovali pomocou *matematických* objektov.

Aký **informatický** objekt zodpovedá štruktúre?

Štruktúra ako informatický objekt

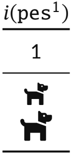
Štruktúru sme definovali pomocou *matematických* objektov.

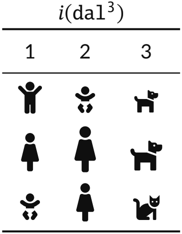
Aký **informatický** objekt zodpovedá štruktúre?

Databáza:

Predikátové symboly jazyka $\sim$ veľmi zjednodušená schéma DB
(arita $\sim$ počet stĺpcov)

Interpretácia predikátových symbolov $\sim$ konkrétne tabuľky s dátami





Štruktúry — upozornenia

Štruktúr pre daný jazyk je **nekonečne veľa**.

Doména štruktúry

- môže mať ľubovoľné prvky;
- nijak **nesúvisí** s intuitívnym významom interpretovaného jazyka;
- môže byť **nekonečná**.

Interpretácia symbolov konštánt:

- každej konštante je priradený objekt domény;
- nie každý objekt domény musí byť priradený nejakej konštante;
- rôznym konštantám môže byť priradený rovnaký objekt.

Interpretácie predikátových symbolov môžu byť **nekonečné**.

Príklad 1.18 (Štruktúra s nekonečnou doménou)

$\mathcal{M} = (\mathbb{N}, i) \quad i(\mathtt{pes}) = \{ 2n \mid n \in \mathbb{N} \} \quad i(\mathtt{dal}) = \{ (n, m, n + m) \mid n, m \in \mathbb{N} \}$

$i(\mathtt{Bobík}) = 0 \quad i(\mathtt{Cilka}) = 1 \quad i(\mathtt{Evka}) = 3 \quad i(\mathtt{Jarka}) = 5 \quad i(\mathtt{Milo}) = 0$

Pravdivosť atomickej formuly v štruktúre

Ako zistíme, či je atomická formula pravdivá v štruktúre?

Definícia 1.19

Nech $\mathcal{M} = (D, i)$ je štruktúra pre jazyk $\mathcal{L}$ atomických formúl jazyka logiky prvého rádu.

Rovnostný atóm $c_1 \doteq c_2$ jazyka $\mathcal{L}$ je ***pravdivý v štruktúre*** $\mathcal{M}$ vtedy a len vtedy, keď $i(c_1) = i(c_2)$.

Predikátový atóm $P(c_1, \dots, c_n)$ jazyka $\mathcal{L}$ je ***pravdivý v štruktúre*** $\mathcal{M}$ vtedy a len vtedy, keď $(i(c_1), \dots, i(c_n)) \in i(P)$.

Vzťah *atóm A je pravdivý v štruktúre $\mathcal{M}$* skrátene zapisujeme $\mathcal{M} \models A$. Hovoríme aj, že $\mathcal{M}$ je ***modelom*** A.

Vzťah *atóm A nie je pravdivý v štruktúre $\mathcal{M}$* zapisujeme $\mathcal{M} \not\models A$. Hovoríme aj, že A ***je nepravdivý*** v $\mathcal{M}$ a $\mathcal{M}$ ***nie je modelom*** A.

Príklad 1.20 (Určenie pravdivosti atómov v štruktúre)

$\mathcal{M} = (D, i)$, $D = \{$ 👧, 👶, 🌲, 🧒, 🏠, 🐈, 🚗, 👩, 🐕, 🐶 $\}$

$i(\mathtt{Bobík}) =$ 🐶 $i(\mathtt{Cilka}) =$ 🐈

$i(\mathtt{Evka}) =$ 👶 $i(\mathtt{Jarka}) =$ 👧 $i(\mathtt{Milo}) =$ 🧒

$i(\mathtt{pes}) = \{$ 🐕, 🐶 $\}$

$i(\mathtt{dal}) = \{($ 🧒, 👶, 🐶 $), ($ 👧, 👩, 🐕 $), ($ 👶, 👧, 🐈 $)\}$

Atóm $\mathtt{pes}(\mathtt{Bobík})$ **je pravdivý** v štruktúre $\mathcal{M}$, t.j., $\mathcal{M} \models \mathtt{pes}(\mathtt{Bobík})$,
lebo objekt $i(\mathtt{Bobík}) =$ 🐶 je prvkom množiny $\{$ 🐕, 🐶 $\} = i(\mathtt{pes})$.

Atóm $\mathtt{dal}(\mathtt{Evka}, \mathtt{Jarka}, \mathtt{Cilka})$ **je pravdivý** v $\mathcal{M}$,
t.j., $\mathcal{M} \models \mathtt{dal}(\mathtt{Evka}, \mathtt{Jarka}, \mathtt{Cilka})$,
lebo $(i(\mathtt{Evka}), i(\mathtt{Jarka}), i(\mathtt{Cilka})) = ($ 👶, 👧, 🐈 $) \in i(\mathtt{dal})$.

Atóm $\mathtt{Cilka} \doteq \mathtt{Bobík}$ **nie je pravdivý** v $\mathcal{M}$, t.j.,
$\mathcal{M} \not\models \mathtt{Cilka} \doteq \mathtt{Bobík}$,
lebo $i(\mathtt{Cilka}) =$ 🐈 $\neq$ 🐶 $= i(\mathtt{Bobík})$.

Atomické formuly

Zhrnutie

Zhrnutie

- Logika prvého rádu je rodina formálnych jazykov.
- Každý jazyk logiky prvého rádu je daný neprázdnou množinou indivíduových konštánt a množinou predikátových symbolov.
- Atomické formuly sú základnými výrazmi prvorádového jazyka.
    - Postupnosti symbolov $P(c_1, \ldots, c_n)$ (predikátové) a $c_1 \doteq c_2$ (rovnostné).
    - Zodpovedajú pozitívnym jednoduchým výrokom o vlastnostiach, stavoch, vzťahoch, rovnosti jednotlivých pomenovaných objektov.
- Význam jazyku dáva štruktúra — matematický opis stavu sveta
    - Skladá sa z neprázdnej domény a z interpretačnej funkcie.
    - Konštanty interpretuje ako prvky domény.
    - Predikáty interpretuje ako podmnožiny domény/relácie na doméne.
- Pravdivosť atómu určíme interpretovaním argumentov a zistením, či je výsledná n-tica objektov prvkom interpretácie predikátu, resp. pri rovnostnom atóme, či sa objekty rovnajú.