Paradigme de Programare

Ș.l. dr. ing. Andrei Olaru

Departamentul Calculatoare
slides: Andrei Olaru & Mihnea Muraru

2013 – 2014, semestrul 2

Cursul 11

Logica cu predicate de ordinul I

Cuprins

Introducere

Logica cu predicate de ordinul I

First Order Predicate Logic (FOL sau FOPL) – Context

- Extensie a logicii propoziționale, cu explicitarea:
  - obiectelor din universul problemei;
  - relațiilor dintre acestea.
- Logica propozițională:
  - p: “Andrei este prieten cu Bogdan.”
  - q: “Bogdan este prieten cu Andrei.”
  - $p \Leftrightarrow q$

  $\rightarrow$ Opacitate în raport cu obiectele și relațiile referite.
- FOL:
  - Generalizare: $prieten(x, y)$: “x este prieten cu y.”
  - $\forall x.\forall y.(prieten(x, y) \Leftrightarrow prieten(y, x))$

  $\rightarrow$ Aplicare pe cazuri particulare.

  $\rightarrow$ Transparență în raport cu obiectele și relațiile referite.

Sintaxă

Sintaxă

Simboluri utilizate

- Constante: obiecte particulare din universul discursului: *c*, *d*, *andrei*, *bogdan*, ...
- Variabile: obiecte generice: *x*, *y*, ...
- Simboluri funcționale: *succesor*, $+$, *abs* ...
- Simboluri relaționale (predicate): relații *n*-are peste obiectele din universul discursului: $prieten = \{(andrei, bogdan), (bogdan, andrei), \ldots\}$, $impar = \{1, 3, \ldots\}$, ...
- Conectori logici: $\neg$, $\wedge$, ...
- Cuantificatori: $\forall$, $\exists$

· Termeni (obiecte):

- Constante;
- Variabile;
- Aplicații de funcții: $f(t_1, \ldots, t_n)$, unde f este un simbol funcțional n-ar și $t_1, \ldots, t_n$ sunt termeni.

  Exemple:

  - $succesor(4)$: succesorul lui 4, și anume 5.
  - $+(2, x)$: aplicația funcției de adunare asupra numerelor 2 și x, și, totodată, suma lor.

· Atomi (relații): atomul $p(t_1, \ldots, t_n)$, unde p este un predicat n-ar și $t_1, \ldots, t_n$ sunt termeni.

Exemple:

- $impar(3)$
- $varsta(ion, 20)$
- $=(+(2,3), 5)$

· Propoziții (fapte) – dacă x variabilă, A atom, și α și β propoziții, atunci o propoziție are forma:

- Fals, Adevărat: $\perp$, $\top$
- Atomi: A
- Negații: $\neg\alpha$
- Conectori: $\alpha \wedge \beta$, $\alpha \Rightarrow \beta$, ...
- Cuantificări: $\forall x.\alpha$, $\exists x.\alpha$

Exemplul 37.1.

“Dan este prieten cu sora Ioanei”

$$\underbrace{\exists X.\underbrace{prieten(\underbrace{X}_{\text{termen}}, \underbrace{sora(\overbrace{ioana}^{\text{termen}})}_{\text{termen}})}_{\text{atom/propoziție}} \wedge \underbrace{destept(X)}_{\text{atom/propoziție}}}_{\text{propoziție}}$$

Semantică

Definiția 38.1 (Interpretare).

O interpretare constă din:

- Un domeniu nevid, D
- Pentru fiecare constantă c, un element $c^I \in D$
- Pentru fiecare simbol funcțional, n-ar f, o funcție $f^I : D^n \to D$
- Pentru fiecare predicat n-ar p, o funcție $p^I : D^n \to \{false, true\}$.

- Atom:
  $(p(t_1, \ldots, t_n))^I = p^I(t_1^I, \ldots, t_n^I)$
- Negație, conectori, implicații: v. logica propozițională
- Cuantificare universală:
  $$(\forall x.\alpha)^I = \begin{cases} \mathit{false} & \text{dacă } \exists d \in D \,.\, \alpha^I_{[d/x]} = \mathit{false} \\ \mathit{true} & \text{altfel} \end{cases}$$
- Cuantificare existențială:
  $$(\exists x.\alpha)^I = \begin{cases} \mathit{true} & \text{dacă } \exists d \in D \,.\, \alpha^I_{[d/x]} = \mathit{true} \\ \mathit{false} & \text{altfel} \end{cases}$$

Exemplul 38.2.

1. "Vrabia mălai visează."

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. "Unele vrăbii visează mălai."

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. "Unele vrăbii visează mălai."
   $\exists x.(vrabie(x) \wedge viseaza(x, malai))$

Exemplul 38.2.

1. “Vrabia mălai visează.” $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. “Unele vrăbii visează mălai.”
   $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
3. “Nu toate vrăbiile visează mălai.”

Exemplul 38.2.

1. “Vrabia mălai visează.” $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. “Unele vrăbii visează mălai.”
   $\exists x.(vrabie(x) \land viseaza(x, malai))$
3. “Nu toate vrăbiile visează mălai.”
   $\exists x.(vrabie(x) \land \neg viseaza(x, malai))$

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. "Unele vrăbii visează mălai."
   $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
3. "Nu toate vrăbiile visează mălai."
   $\exists x.(vrabie(x) \wedge \neg viseaza(x, malai))$
4. "Nicio vrabie nu visează mălai."

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. "Unele vrăbii visează mălai."
   $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
3. "Nu toate vrăbiile visează mălai."
   $\exists x.(vrabie(x) \wedge \neg viseaza(x, malai))$
4. "Nicio vrabie nu visează mălai."
   $\forall x.(vrabie(x) \Rightarrow \neg viseaza(x, malai))$

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. "Unele vrăbii visează mălai."
   $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
3. "Nu toate vrăbiile visează mălai."
   $\exists x.(vrabie(x) \wedge \neg viseaza(x, malai))$
4. "Nicio vrabie nu visează mălai."
   $\forall x.(vrabie(x) \Rightarrow \neg viseaza(x, malai))$
5. "Numai vrăbiile visează mălai."

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. "Unele vrăbii visează mălai."
   $\exists x.(vrabie(x) \land viseaza(x, malai))$
3. "Nu toate vrăbiile visează mălai."
   $\exists x.(vrabie(x) \land \neg viseaza(x, malai))$
4. "Nicio vrabie nu visează mălai."
   $\forall x.(vrabie(x) \Rightarrow \neg viseaza(x, malai))$
5. "Numai vrăbiile visează mălai."
   $\forall x.(viseaza(x, malai) \Rightarrow vrabie(x))$

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. "Unele vrăbii visează mălai."
   $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
3. "Nu toate vrăbiile visează mălai."
   $\exists x.(vrabie(x) \wedge \neg viseaza(x, malai))$
4. "Nicio vrabie nu visează mălai."
   $\forall x.(vrabie(x) \Rightarrow \neg viseaza(x, malai))$
5. "Numai vrăbiile visează mălai."
   $\forall x.(viseaza(x, malai) \Rightarrow vrabie(x))$
6. "Toate și numai vrăbiile visează mălai."

Exemplul 38.2.

1. "Vrabia mălai visează." $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
2. "Unele vrăbii visează mălai."
   $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
3. "Nu toate vrăbiile visează mălai."
   $\exists x.(vrabie(x) \wedge \neg viseaza(x, malai))$
4. "Nicio vrabie nu visează mălai."
   $\forall x.(vrabie(x) \Rightarrow \neg viseaza(x, malai))$
5. "Numai vrăbiile visează mălai."
   $\forall x.(viseaza(x, malai) \Rightarrow vrabie(x))$
6. "Toate și numai vrăbiile visează mălai."
   $\forall x.(viseaza(x, malai) \Leftrightarrow vrabie(x))$

Cuantificatori
Greșeli frecvente

- $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ corect: “Toate vrăbiile visează mălai.”
- $\forall x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ greșit: “Toți sunt vrăbii care visează mălai.”
- $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ corect: “Unele vrăbii visează mălai.”
- $\exists x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ greșit: adevărată și dacă există cineva care nu este vrabie.

Cuantificatori
Greșeli frecvente

- $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ corect: “Toate vrăbiile visează mălai.”

- $\forall x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ greșit: “Toți sunt vrăbii care visează mălai.”

- $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ corect: “Unele vrăbii visează mălai.”

- $\exists x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ greșit: adevărată și dacă există cineva care nu este vrabie.

- $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ corect: “Toate vrăbiile visează mălai.”
- $\forall x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ greșit: “Toți sunt vrăbii care visează mălai.”
- $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ corect: “Unele vrăbii visează mălai.”
- $\exists x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ greșit: adevărată și dacă există cineva care nu este vrabie.

Cuantificatori
Greșeli frecvente

- $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ corect: “Toate vrăbiile visează mălai.”
- $\forall x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ greșit: “Toți sunt vrăbii care visează mălai.”
- $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ corect: “Unele vrăbii visează mălai.”
- $\exists x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ greșit: adevărată și dacă există cineva care nu este vrabie.

Cuantificatori
Greșeli frecvente

- $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ corect: “Toate vrăbiile visează mălai.”
- $\forall x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ greșit: “Toți sunt vrăbii care visează mălai.”
- $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ corect: “Unele vrăbii visează mălai.”
- $\exists x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ greșit: adevărată și dacă există cineva care nu este vrabie.

Cuantificatori
Proprietăți

- Necomutativitate:
  - $\forall x.\exists y.viseaza(x,y) \rightarrow$ “Toți visează la ceva anume.”
  - $\exists x.\forall y.viseaza(x,y) \rightarrow$ “Există cineva care visează la orice.”
- Dualitate:
  - $\neg(\forall x.\alpha) \equiv \exists x.\neg\alpha$
  - $\neg(\exists x.\alpha) \equiv \forall x.\neg\alpha$

Aspecte legate de propoziții
Analoage logicii propoziționale

- Satisfiabilitate.
- Validitate.
- Derivabilitate.
- Inferență.

Forme normale

Forme normale
Definiții (1)

Definiția 39.1 (Literal).

Atom sau negația lui. Exemplu: $prieten(x, y)$, $\neg prieten(x, y)$.

Definiția 39.2 (Expresie clauzală).

Literal sau disjuncție de literali.
Exemplu: $prieten(x, y) \vee \neg doctor(x)$.

Definiția 39.3 (Clauză).

Mulțime de literali dintr-o expresie clauzală. Exemplu:
$\{prieten(x, y), \neg doctor(x)\}$.

Definiția 39.4 (Forma clauzală / Forma normală conjunctivă – FNC).

Reprezentarea unei propoziții sub forma unei mulțimi de clauze, implicit legate prin conjuncții.

Definiția 39.5 (Forma normală implicativă – FNI).

Reprezentarea unei propoziții sub forma unei mulțimi de clauze, implicit legate prin conjuncții, în care fiecare clauză are forma grupată
$\{\neg A_1, \ldots, \neg A_m, B_1, \ldots, B_n\}$,
corespunzătoare implicației
$(A_1 \wedge \ldots \wedge A_m) \Rightarrow (B_1 \vee \ldots \vee B_n)$, unde A_i și B_j sunt atomi.

Forme normale

Definiții (3) – Clauze Horn

Definiția 39.6 (Clauză Horn).

Clauză în care un singur literal este în formă pozitivă:
$\{\neg A_1, \ldots, \neg A_n, A\}$,
corespunzătoare implicației
$A_1 \wedge \ldots \wedge A_n \Rightarrow A$.

Exemplul 39.7 (Clauze Horn).

Transformarea propoziției
$vrabie(x) \vee ciocarlie(x) \Rightarrow pasare(x)$ în forme normale, utilizând clauze Horn:

- FNC: $\{\neg vrabie(x), pasare(x)\}, \{\neg ciocarlie(x), pasare(x)\}$
- FNI: $vrabie(x) \Rightarrow pasare(x), ciocarlie(x) \Rightarrow pasare(x)$

Conversia propozițiilor în FNC (1)
Eliminare implicații, împingere negații, redenumiri

1. Eliminarea implicațiilor ($\not\Rightarrow$)
2. Împingerea negațiilor până în fața literalilor ($\overset{\rightarrow}{\neg}$)
3. Redenumirea variabilelor cuantificate pentru obținerea unicității de nume (R):

$$\forall x.p(x) \wedge \forall x.q(x) \vee \exists x.r(x) \rightarrow \forall x.p(x) \wedge \forall y.q(y) \vee \exists z.r(z)$$

4. Deplasarea cuantificatorilor la începutul expresiei, conservându-le ordinea (forma normală *prenex*) (P):

$$\forall x.p(x) \wedge \forall y.q(y) \vee \exists z.r(z) \rightarrow \forall x.\forall y.\exists z.(p(x) \wedge q(y) \vee r(z))$$

Conversia propozițiilor în FNC (2)
Skolemizare

5. Eliminarea cuantificatorilor existențiali (skolemizare) (S):
   - Dacă nu este precedat de cuantificatori universali: înlocuirea aparițiilor variabilei cuantificate printr-o constantă:
   $$\exists x.p(x) \rightarrow p(c_x)$$
   - Dacă este precedat de cuantificatori universali: înlocuirea aparițiilor variabilei cuantificate prin aplicația unei funcții unice asupra variabilelor anterior cuantificate universal:
   $$\forall x.\forall y.\exists z.(p(x) \wedge q(y) \vee r(z))$$
   $$\rightarrow \forall x.\forall y.(p(x) \wedge q(y) \vee r(f_z(x, y)))$$

Conversia propozițiilor în FNC (3)

Cuantificatori universali, Distribuire $\vee$, Clauze

6. Eliminarea cuantificatorilor universali, considerați, acum, impliciți ($\cancel{\forall}$):

$$\forall x.\forall y.(p(x) \wedge q(y) \vee r(f_z(x,y))) \rightarrow p(x) \wedge q(y) \vee r(f_z(x,y))$$

7. Distribuirea lui $\vee$ față de $\wedge$ ($\vee/\wedge$):

$$\alpha \vee (\beta \wedge \gamma) \rightarrow (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$$

8. Transformarea expresiilor în clauze (C).

Conversia propozițiilor în FNC – Exemplu

Exemplul 39.8.

“Cine rezolvă toate laboratoarele este apreciat de cineva.”

Conversia propozițiilor în FNC – Exemplu

Exemplul 39.8.

"Cine rezolvă toate laboratoarele este apreciat de cineva."

$\forall x.(\forall y.(lab(y) \Rightarrow rezolva(x, y)) \Rightarrow \exists y.apreciaza(y, x))$

Conversia propozițiilor în FNC – Exemplu

Exemplul 39.8.

"Cine rezolvă toate laboratoarele este apreciat de cineva."

$\forall x.(\forall y.(lab(y) \Rightarrow rezolva(x,y)) \Rightarrow \exists y.apreciaza(y,x))$

$\not\Rightarrow$	$\forall x.(\neg\forall y.(\neg lab(y) \lor rezolva(x,y)) \lor \exists y.apreciaza(y,x))$
$\overset{\rightarrow}{\neg}$	$\forall x.(\exists y.\neg(\neg lab(y) \lor rezolva(x,y)) \lor \exists y.apreciaza(y,x))$
$\overset{\rightarrow}{\neg}$	$\forall x.(\exists y.(lab(y) \land \neg rezolva(x,y)) \lor \exists y.apreciaza(y,x))$
R	$\forall x.(\exists y.(lab(y) \land \neg rezolva(x,y)) \lor \exists z.apreciaza(z,x))$
P	$\forall x.\exists y.\exists z.((lab(y) \land \neg rezolva(x,y)) \lor apreciaza(z,x))$
S	$\forall x.((lab(f_y(x)) \land \neg rezolva(x,f_y(x))) \lor apreciaza(f_z(x),x))$
$\cancel{\forall}$	$(lab(f_y(x)) \land \neg rezolva(x,f_y(x))) \lor apreciaza(f_z(x),x)$
$\lor/\land$	$(lab(f_y(x)) \lor apr(f_z(x),x)) \land (\neg rez(x,f_y(x)) \lor apr(f_z(x),x))$
C	$\{lab(f_y(x)), apr(f_z(x),x)\}, \{\neg rez(x,f_y(x)), apr(f_z(x),x)\}$

Unificare și rezoluție

Unificare

- Utilizată pentru rezoluție
- vezi și sinteza de tip – Def. 23.5, 23.6, 23.8
- reguli:
  - o propoziție unifică cu o propoziție de aceeași formă
  - două predicate unifică dacă au același nume și parametri care unifică
  - o variabilă unifică cu un termen care nu conține variabila

- Problemă NP-completă;
- Posibile legări ciclice;
- Exemplu:
  $prieten(x, mama(x))$ și $prieten(mama(y), y)$
  MGU: $S = \{x \leftarrow mama(y), y \leftarrow mama(x)\}$
  $\Rightarrow x \leftarrow mama(mama(x)) \rightarrow$ imposibil!
- Soluție: verificarea apariției unei variabile în valoarea la care a fost legată (*occurrence check*);

- Rezoluția pentru clauze Horn:

$$\frac{\begin{array}{l} A_1 \wedge \ldots \wedge A_m \Rightarrow A \\ B_1 \wedge \ldots \wedge A' \wedge \ldots \wedge B_n \Rightarrow B \\ unificare(A, A') = S \end{array}}{subst(S, A_1 \wedge \ldots \wedge A_m \wedge B_1 \wedge \ldots \wedge B_n \Rightarrow B)}$$

- $unificare(\alpha, \beta) \rightarrow$ substituția sub care unifică propozițiile α și β;
- $subst(S, \alpha) \rightarrow$ propoziția rezultată în urma aplicării substituției S asupra propoziției α.

Exemplul 40.1.

Horses and hounds

- Horses are faster than dogs.
- There is a greyhound that is faster than any rabbit.
- Harry is a horse and Ralph is a rabbit.
- Is Harry faster than Ralph?

Sfârșitul cursului 11(a)
Ce am învățat

- sintaxa și semantica în LPOI
- Forme normale, Unificare, Rezoluție în LPOI