Paradigme de Programare

Ș.I. dr. ing. Andrei Olaru

Departamentul Calculatoare

2014 – 2015, semestrul 2

- Harry is a horse.
- Ralph is a rabbit.
- if all horses are faster than any dog and there is a greyhound faster than any rabbit,
- Is Harry faster than Ralph?

Cursul 11: Logica cu predicate de ordinul I

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** $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftarrow$
- **+ | 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.

Ex | 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.

Ex | 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: $\bot$, $\top$
- Atomi: A
- Negații: $\neg\alpha$
- Conectori: $\alpha \wedge \beta$, $\alpha \Rightarrow \beta$, ...
- Cuantificări: $\forall x.\alpha$, $\exists x.\alpha$

Ex Exemplu

"Sora Ioanei are un prieten deștept"

Ex Exemplu

"Sora Ioanei are un prieten deștept"

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

Semantică

+ Interpretarea constă din:

- Un domeniu nevid, D, de concepte (obiecte)
- 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:
  $$(\exists x.\alpha)^I = \begin{cases} \mathit{true} & \text{dacă } \exists d \in D \,.\, \alpha^I_{[d/x]} = \mathit{true} \\ \mathit{false} & \text{altfel} \end{cases}$$

Ex Exemple cu cuantificatori

1. "Vrabia mălai visează."

Ex Exemple cu cuantificatori

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

Ex Exemple cu cuantificatori

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

Ex | Exemple cu cuantificatori

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

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

Ex Exemple cu cuantificatori

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."

Exemple cu cuantificatori

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."
   $\exists x.(vrabie(x) \wedge \neg viseaza(x, malai))$

Exemple cu cuantificatori

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."

Ex | Exemple cu cuantificatori

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."
   $\forall x.(vrabie(x) \Rightarrow \neg viseaza(x, malai))$

Exemple cu cuantificatori

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."

Ex Exemple cu cuantificatori

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."
   $\forall x.(viseaza(x, malai) \Rightarrow vrabie(x))$

- $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$

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

- $\forall x.(vrabie(x) \wedge viseaza(x, malai))$

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 și toți visează mălai.”
- $\exists x.(vrabie(x) \wedge viseaza(x, malai))$

- $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ corect: "Toate vrăbiile visează mălai."
- $\forall x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ greșit: "Toți sunt vrăbii și toți visează mălai."
- $\exists x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ corect: "Unele vrăbii visează mălai."
- $\exists x.(vrabie(x) \Rightarrow viseaza(x, malai))$

- $\forall x.(vrabie(x) \Rightarrow viseaza(x, malai))$
  $\rightarrow$ corect: "Toate vrăbiile visează mălai."
- $\forall x.(vrabie(x) \wedge viseaza(x, malai))$
  $\rightarrow$ greșit: "Toți sunt vrăbii și toți 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: probabil nu are semnificația pe care o intenționăm. Este adevărată și dacă luăm un x care nu este vrabie (fals implică orice).

- 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:
  - $\neg(\forall x.\alpha) \equiv \exists x.\neg\alpha$
  - $\neg(\exists x.\alpha) \equiv \forall x.\neg\alpha$

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

Forme normale

+ Literal – Atom sau negația unui atom.

Exemplu $prieten(x,y)$, $\neg prieten(x,y)$.

+ Clauză – Mulțime de literali dintr-o expresie clauzală.

Exemplu $\{prieten(x,y), \neg doctor(x)\}$.

+ Forma normală conjunctivă – FNC – Reprezentare ca mulțime de clauze, cu semnificație conjunctivă.

+ Forma normală implicativă – FNI – Reprezentare ca mulțime de clauze cu clauzele în forma grupată $\{\neg A_1, \ldots, \neg A_m, B_1, \ldots, B_n\}$, $\Leftrightarrow$ $(A_1 \wedge \ldots \wedge A_m) \Rightarrow (B_1 \vee \ldots \vee B_n)$

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

Ex | Exemplu Transformarea propoziției
$vrabie(x) \vee ciocarlie(x) \Rightarrow pasare(x)$ în formă normale, utilizând clauze Horn:
FNC: $\{\neg vrabie(x), pasare(x)\}, \{\neg ciocarlie(x), pasare(x)\}$

Conversia propozițiilor în FNC (1)

Eliminare implicații, împingere negații, redenumiri

1. Eliminarea implicațiilor ($\not\Rightarrow$)
2. Împingerea negațiilor până în fața literalilor ($\overrightarrow{\neg}$)
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):

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

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 ($\not\forall$):

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

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

Ex | Exemplu "Cine rezolvă toate laboratoarele este apreciat de cineva."

Conversia propozițiilor în FNC – Exemplu

Ex Exemplu "Cine rezolvă toate laboratoarele este apreciat de cineva."

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

Ex Exemplu "Cine rezolvă toate laboratoarele este apreciat de cineva."

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

$\not\Rightarrow \quad \forall x.(\neg\forall y.(\neg lab(y) \vee rezolva(x,y)) \vee \exists y.apreciaza(y,x))$

$\neg\rightarrow \quad \forall x.(\exists y.\neg(\neg lab(y) \vee rezolva(x,y)) \vee \exists y.apreciaza(y,x))$

$\neg\rightarrow \quad \forall x.(\exists y.(lab(y) \wedge \neg rezolva(x,y)) \vee \exists y.apreciaza(y,x))$

R $\quad \forall x.(\exists y.(lab(y) \wedge \neg rezolva(x,y)) \vee \exists z.apreciaza(z,x))$

P $\quad \forall x.\exists y.\exists z.((lab(y) \wedge \neg rezolva(x,y)) \vee apreciaza(z,x))$

S $\quad \forall x.((lab(f_y(x)) \wedge \neg rezolva(x,f_y(x))) \vee apreciaza(f_z(x),x))$

$\not\forall \quad (lab(f_y(x)) \wedge \neg rezolva(x,f_y(x))) \vee apreciaza(f_z(x),x)$

$\vee/\wedge \quad (lab(f_y(x)) \vee apr(f_z(x),x)) \wedge (\neg rez(x,f_y(x)) \vee apr(f_z(x),x))$

C $\quad \{lab(f_y(x)), apr(f_z(x),x)\}, \{\neg rez(x,f_y(x)), apr(f_z(x),x)\}$

Unificare și rezoluție

- Utilizată pentru rezoluție
- vezi și sinteza de tip în Haskell

cum știm dacă folosind ipoteza $om(Marcel)$ și propoziția $\forall om(x) \Rightarrow are\_inima(x)$ putem demonstra că $are\_inima(Marcel)$ → unificând $om(Marcel)$ și $\forall om(x)$.

- reguli:
  - o propoziție unifică cu o propoziție de aceeași formă
  - două predicate unifică dacă au același nume și parametri care unifică (*om* cu *om*, *x* cu *Marcel*)
  - o constantă unifică cu o constantă cu același nume
  - o variabilă unifică cu un termen care nu conține variabila (*x* cu *Marcel*)

- 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.

$P \vee \overline{P}$

Exemplu

Horses and hounds

1. Horses are faster than dogs.
2. There is a greyhound that is faster than any rabbit.
3. Harry is a horse and Ralph is a rabbit.
4. Is Harry faster than Ralph?

1. $\forall x.\forall y.horse(x) \wedge dog(y) \Rightarrow faster(x,y)$
   $\rightarrow \neg horse(x) \vee \neg dog(y) \vee faster(x,y)$
2. $\exists x.greyhound(x) \wedge (\forall y.rabbit(y) \Rightarrow faster(x,y))$
   $\rightarrow greyhound(Greg) \quad ; \quad \neg rabbit(y) \vee faster(Greg,y)$
3. $horse(Harry) \quad ; \quad rabbit(Ralph)$
4. $\neg faster(Harry,Ralph)$ (concluzia negată)
5. $\neg greyhound(x) \vee dog(x)$ (common knowledge)
6. $\neg faster(x,y) \vee \neg faster(y,z) \vee faster(x,z)$ (tranzitivitate)
7. 1 + 3a $\rightarrow \neg dog(y) \vee faster(Harry,y)$ (cu $\{Harry/x\}$)
8. 2a + 5 $\rightarrow dog(Greg)$ (cu $\{Greg/x\}$)
9. 7 + 8 $\rightarrow faster(Harry,Greg)$ (cu $\{Greg/y\}$)
10. 2b + 3b $\rightarrow faster(Greg,Ralph)$ (cu $\{Ralph/y\}$)
11. 6 + 9 + 10 $\rightarrow faster(Harry,Ralph)$ $\{Harry/x, Greg/y, Ralph/z\}$
12. 11 + 4 $\rightarrow \square$ q.e.d.

Sfârșitul cursului 11(a)

Ce am învățat

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