FMI, Info, Anul II
Programare logică

Exerciţii

(P1) Folosind deducţia naturală, demonstraţi că următoarul secvent este valid:

(i) $\vdash \neg(p \wedge \neg p)$

(ii) $p \wedge q \rightarrow \neg u,\ p \rightarrow u,\ p,\ q \ \vdash\ \neg r$

(iii) $p, p \rightarrow q \vdash p \wedge q$

(P2) Găsiţi un cel mai general unificator pentru termenii

(i) $g(x, f(x), f(a), v)$ şi $g(h(y, v), f(h(z, u)), z, f(z))$,

(ii) $h(f(h(x, y)), a)$, $h(f(z), u)$ şi $h(f(x), f(y))$

(iii) $g(x, f(x), f(a), v)$ şi $g(h(y, v), f(h(z, u)), z, f(z))$

unde x, y, z şi u sunt variabile, a este simbol de constantă, f este simbol de operaţie de aritate 1, h este simbol de operaţie de aritate 2, iar g este simbol de operaţie de aritate 4.

(P3) Să se aducă la forma normală prenex şi apoi la forma Skolem următoarea formulă:

$$\forall x(\exists y A(x, y) \wedge \forall y \neg B(x, y) \rightarrow \neg(\exists y A(x, y) \wedge C(c)))$$

unde A şi B sunt predicate de aritate 2, C este un predicat de aritate 1, iar c este o constantă.

(P4) Să se aducă la forma normală prenex şi apoi la forma Skolem următoarea formulă:

$$\neg(\forall x \exists y A(u, x, y) \rightarrow \exists x(\neg \forall y B(y, v) \rightarrow C(x)))$$

unde A este un predicat de aritate 3, B este un predicat de aritae 2, iar C este un predicat de aritate 1.

(P5) Fie C o clauză în calculul propoziţional clasic şi p o variabilă propoziţională astfel încât $\{p, \neg p\} \notin C$ (adică nu apar în C). Demonstraţi că, oricare ar fi $a : Var \rightarrow \{0, 1\}$ o evaluare, a^+ este model pentru $C \cup \{p\}$ sau a^+ este model pentru $C \cup \{\neg p\}$.

(P6) Aduceţi următoarea formulă de ordinul I la forma clauzală:

$$\neg(\forall x \exists y A(u, x, y) \to \exists x(\neg \forall y B(y, v) \to C(x)))$$

unde A este un predicat de aritate 3, B este un predicat de aritae 2, iar C este un predicat de aritate 1.

(P7) Găsiţi o SLD-respingere pentru programul Prolog de mai jos şi ţinta `?- p(X),m(Y,X),p(Y)`. Indicaţi la fiecare pas regula şi substituţia folosite pentru a aplica rezoluţia.

(1) `f(d,b) .`
(2) `f(d,d) .`
(3) `m(a,d) .`
(4) `m(b,c) .`
(5) `p(a) .`
(6) `p(d) .`
(7) `p(Y) :- f(Y,X), p(X) .`

(P8) Găsiţi o SLD-respingere pentru programul Prolog de mai jos şi ţinta `?- q(c,a)`. Indicaţi la fiecare pas regula şi substituţia folosite pentru a aplica rezoluţia.

(i) `q(a,b) .`

(ii) `q(c,b) .`

(iii) `q(X,Z) :- q(X,Y), q(Y,Z) .`

(iv) `q(X,Y) :- q(Y,X) .`

(P9) Fie T următorul program Prolog:

```
lista([]).
lista([a|L]) :- lista(L).
lista([b|L]) :- lista(L).
```

(i) Determinaţi cel mai mic model Herbrand al lui T.

(ii) Pentru acest program particular, explicaţi modul în care cel mai mic model Herbrand se poate calcula aplicând teorema de punct fix Knaster-Tarski.

(P10) Se dă următorul program Prolog:

```
e(a). o(b). o(c).
e(p(X)) :- o(X). o(p(X)):- e(X).
e(p(p(X))) :- e(X). o(p(p(X))) :- o(X).
```

Pentru programul Prolog de mai sus determinaţi universul Herbrand, baza Herbrand şi cel mai mic model Herbrand.