Programare Logică – listă subiecte de examen

Claudia MUREŞAN, c.muresan@yahoo.com, cmuresan@fmi.unibuc.ro
Universitatea din Bucureşti, Facultatea de Matematică şi Informatică

2019–2020, Semestrul I

Exerciţiul 1. Considerăm un limbaj de ordinul I conţinând un simbol de operaţie binară f şi două simboluri distincte de operaţii unare g şi h. Fie V, X şi Y variabile distincte.

Să se deseneze arborii de expresii asociaţi următorilor doi termeni, apoi, prin aplicarea algoritmului de unificare, să se determine dacă aceşti termeni au unificator şi, în caz afirmativ, să se determine un cel mai general unificator pentru aceştia:

$$f(f(h(X), h(h(V))), g(h(f(V, V)))) \text{ şi } f(f(h(g(V)), h(h(Y))), g(h(f(X, Y)))).$$

Exerciţiul 2. Având următoarea bază de cunoştinţe în Prolog, scrisă respectând sintaxa Prolog:
$barbat(alex)$. $barbat(andrei)$.
$femeie(ana)$. $femeie(carmen)$. $femeie(elena)$. $femeie(maria)$.
$ochi(alex, negri)$.
$ochi(ana, caprui)$.
$ochi(andrei, albastri)$.
$ochi(carmen, violeti)$.
$ochi(elena, verzi)$.
$ochi(maria, caprui)$.
$place(B, F)$:- $barbat(B)$, $femeie(F)$, $ochi(F, violeti)$.
$place(B, F)$:- $barbat(B)$, $femeie(F)$, $ochi(F, verzi)$.
$place(F, B)$:- $femeie(F)$, $barbat(B)$, $ochi(B, albastri)$.
$logoditi(B, F)$:- $place(B, F)$, $place(F, B)$.
$rivali(P, Q)$:- $femeie(P)$, $femeie(Q)$, $place(B, P)$, $place(B, Q)$.
$rivali(P, Q)$:- $barbat(P)$, $barbat(Q)$, $place(F, P)$, $place(F, Q)$.
să se scrie arborele de derivare prin rezoluţie SLD pentru următoarea interogare:
?- $rivali(Cine, CuCine)$.

După obţinerea tuturor soluţiilor interogării, dacă apar în arborele de derivare noduri pentru care expandarea (recursia) continuă la infinit, să se indice acele noduri, fără a continua cu expandarea lor.

Toate nodurile care au doar nu număr finit de descendenţi vor fi expandate complet, chiar dacă produc aceleaşi soluţii de mai multe ori.

Exerciţiul 3. Să se scrie în Prolog un predicat binar $listintermnr(Lista, ListaElemsiListe)$, definit ca mai jos, precum şi toate predicatele auxiliare necesare pentru definirea acestuia:

listintermnr să fie satisfăcut ddacă ambele sale argumente sunt liste, iar al doilea argument se obţine din primul prin înlocuirea secvenţelor nevide de elemente ale sale de pe poziţii consecutive care nu sunt constante numerice cu listele formate din elementele din acele secvenţe;

şi, într–o interogare în Prolog, *listintermnr* să funcţioneze sub forma: dacă primeşte o listă arbitrară *Lista* în primul argument, să construiască în al doilea argument lista L obţinută din *Lista* lsând constantele numerice din *Lista* pe loc şi înlocuind şirurile (nevide) de elemente $E1, \ldots, En$ din *Lista* cuprinse între constantele numerice cu câte un singur element, dat de lista formată din elementele $E1, \ldots, En$; de exemplu:

la interogările următoare:	Prologul să răspundă:
?- $listintermnr([], L)$.	$L = []$;
?- $listintermnr([1, -2.5, 0.0, 0], L)$.	$L = [1, -2.5, 0.0, 0]$;
?- $listintermnr([0, a, b, X, 1, -1, X, a], L)$.	$L = [0, [a, b, X], 1, -1, [X, a]]$;

iar, la interogarea:
?- $listintermnr([[f(a),[1,a,V,2]],[\,],0,[a],[1,-1],[b,X],[1,a,b,-1],1,-1,f(X),a,2],L)$.
Prologul să răspundă:
$L = [[[f(a),[1,a,V,2]],[\,]],0,[[a],[1,-1],[b,X],[1,a,b,-1]],1,-1,[f(X),a],2]$.

Exerciţiul 4. Să se scrie în Prolog un predicat binar $argtermcomp(Termen, TermenModificat)$ definit ca mai jos, precum şi toate predicatele auxiliare necesare pentru definirea acestuia:

argtermcomp să fie satisfăcut ddacă ambele argumente ale sale sunt termeni Prolog, iar al doilea argument al său coincide cu primul, dacă primul argument este o constantă sau o variabilă, iar, dacă primul argument este un termen compus, atunci doilea argument se obţine din primul prin înlocuirea, în fiecare subtermen, a secvenţelor de argumente consecutive ale operatorului dominant al acelui subtermen care nu sunt termeni compuşi cu un singur argument, egal cu numărul argumentelor din secvenţa înlocuită;

şi, într–o interogare în Prolog, *argtermcomp* să funcţioneze sub forma: dacă primeşte în primul argument un termen Prolog arbitrar T, să construiască în al doilea argument termenul obţinut din T ca mai sus; de exemplu:

la interogările următoare:	Prologul să răspundă:
?- $argtermcomp(X, Termen)$.	$Termen = X$;
?- $argtermcomp(1.5, Termen)$.	$Termen = 1.5$;
?- $argtermcomp(c, Termen)$.	$Termen = c$;
?- $argtermcomp(f(X, f(a, g(b)), 1), Termen)$.	$Termen = f(1, f(1, g(1)), 1)$;

iar, la interogarea:
?- $argtermcomp(f(X,X,Y,1,f(a),f(a),u,V,f(a,b,g(f(a),c,V,h(a,X,b),2.5))),Termen)$.
Prologul să răspundă:
$Termen = f(4, f(1), f(1), 2, f(2, g(f(1), 2, h(3), 1)))$.