

Programare Logică – TEMĂ DE LABORATOR PENTRU AMBELE SERII

Claudia MUREȘAN

UNIVERSITATEA DIN BUCUREȘTI, FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

c.muresan@yahoo.com, cmuresan@fmi.unibuc.ro

2019–2020, Semestrul II

Fiecare semigrupă sau grupă din seria de la Zi, precum și seria de la ID, să îmi trimită pe AMBELE adrese de email de mai sus câte UN SINGUR fișier .pl conținând rezolvarea fiecărui exercițiu:

- pentru Exercițiul 1, până la data de **17 mai** INCLUSIV (adică până la miezul nopții din **17 mai** spre **18 mai**);
- pentru Exercițiul 2, până la data de **31 mai** inclusiv (adică până la miezul nopții din **31 mai** spre **1 iunie**).

Exercițiul 1. Încărcați în Prolog fișierele **arbbin.pl** (din lecția de laborator cu arbori binari) și **liniar_liste_termeni_arbori_expresii.pl** (din lecția de laborator cu liste liniarizate și arbori de expresii), apoi dați interogările următoare:

?- arbbinterm([1, 2, 3], ArbBinExpr), repr(ArbBinExpr).

?- arbbinterm([1, 2, 3|[4, 5]], ArbBinExpr), repr(ArbBinExpr).

Puteți vedea, în aceste interogări, structura listelor [1, 2, 3], respectiv [1, 2, 3|[4, 5]], ca expresii scrise cu operatorul binar [|] și constanta [], reprezentate ca arbori binari crescând din stânga spre dreapta.

Observați că predicatul **arbbinterm** din **liniar_liste_termeni_arbori_expresii.pl** e satisfăcut numai pentru termeni formați cu operatori de aritate cel mult 2. Dați interogările:

?- arbbinterm(1, ArbBinExpr), repr(ArbBinExpr).

?- arbbinterm(f(1), ArbBinExpr), repr(ArbBinExpr).

?- arbbinterm(g(1, 2), ArbBinExpr), repr(ArbBinExpr).

?- arbbinterm(g(f(f(1)), g(2, f(3))), ArbBinExpr), repr(ArbBinExpr).

?- arbbinterm([1, f(1), g(1, 2), g(f(f(1))), g(2, f(3))], ArbBinExpr), repr(ArbBinExpr).

?- arbbinterm(h(1, 2, 3), ArbBinExpr), repr(ArbBinExpr).

Pentru interogările următoare, observați că numele de variabile sunt înlocuite cu nume temporare în afișarea realizată cu predicatul **repr** din fișierul **arbbin.pl**:

?- arbbinterm(g(1, X), ArbBinExpr), repr(ArbBinExpr).

?- arbbinterm(g(f(f(X)), g(Y, f(3))), ArbBinExpr), repr(ArbBinExpr).

Tema voastră constă în modificarea predicatului **arbbinterm** din fișierul **liniar_liste_termeni_arbori_expresii.pl** în următoarele moduri:

- ① să accepte în primul argument termeni formați cu operații de orice aritate, dar, dacă un termen are o operație dominantă h de aritate strict mai mare decât 2, atunci arborele binar asociat celui termen să aibă drept subarboare drept al rădăcinii o frunză etichetată cu lista argumentelor de pe pozițiile 2, 3, ... ale lui h , neexpandate, i.e. ca atare, nu transformate în arborii asociați lor (eticheta rădăcinii să rămână h , iar subarboarele stâng al rădăcinii să rămână arborele binar asociat primului argument al lui h);
- ② să expandeze doar listele, adică, dacă un termen nu are operația dominantă [|], atunci, indiferent dacă are sau nu subtermeni care sunt liste, subarboarele asociat celui termen să fie o frunză cu eticheta dată de acel întreg termen.

Exercițiul 2. Observați, în fișierul **xsio.pl**, mutările efectuate cu predicatul **pune**, și cum memorăm, de la mutare la mutare, configurația curentă a tablei de joc, folosind predicatele predefinite **retract** și **asserta** pentru modificarea bazei de cunoștințe și predicatul unar **config**, care trebuie declarat ca fiind dinamic, adică, în anumite momente, să poată fi nedefinit (i.e. fără clauze în baza de cunoștințe care să specifice valori ale unicului său argument pentru care acest predicat e satisfăcut sau nu).

Să se modifice predicatele acest fișier astfel încât jocul din fereastra interpretorului de Prolog, jucător contra jucător, să fie:

- ① **jocul de șah;**
- ② **jocul de reversi.**

Pentru fiecare dintre aceste jocuri, în configurația memorată în baza de cunoștințe să se rețină și jucătorul care trebuie să facă următoarea mutare:

- ① pentru **jocul de șah**, sub forma: `config(AlbulsauNegrul, ListadeLiniiTablei)` sau `config(AlbulsauNegrul, ListaCoordonatelorPieselor)` sau `config(AlbulsauNegrul, ListaCoordonatelorPieselor, ListadeLiniiTablei)`;
- ② pentru **jocul de reversi**, sub forma `config(VerdelesauRosul, ListadeLiniiTablei)`, eventual și cu alte date utile pentru implementare.

În cazul **jocului de șah**, piesele să fie reprezentate pe tabla, de exemplu, prin constantele: *ra, qa, ta, na, ca, pa, rn, qn, tn, nn, cn, pn* (sau prin constantele șir de caractere 'R', 'Q', 'T', 'N', 'C', 'P', 'r', 'q', 't', 'n', 'c', 'p') pentru piesele: regele, regina, turele, nebunii, caii, respectiv pionii albi, respectiv regele, regina, turele, nebunii, caii, respectiv pionii negri.

Mutările să fie date sub forma:

- ① `muta(DeLaLinia, DeLaColoana, LaLinia, LaColoana)` în cazul **jocului de șah**;
- ② `pune(Culoare, PeLinia, PeColoana)` în cazul **jocului de reversi**.

La fiecare mutare să se verifice corectitudinea mutării (iar, dacă mutarea nu este corectă, la fel ca în **xsio.pl**, să nu se modifice configurația curentă și să se aștepte o mutare corectă):

- ① în cazul **jocului de șah**, să se verifice faptul că piesa mutată (aflată la coordonatele (DeLaLinia, DeLaColoana)), este a jucătorului care trebuie să facă mutarea, că mutarea e legală pentru acea piesă (de exemplu, dacă piesa este un nebun, atunci $|DeLaLinia - LaLinia| = |DeLaColoana - LaColoana|$) și că, dacă locația (LaLinia, LaColoana) nu este vacantă, atunci piesa de la această locație este a adversarului;
- ② în cazul **jocului de reversi**, că locația (PeLinia, PeColoana) este vacantă și că discul pus pe tablă are în sus fața de culoarea cu care joacă jucătorul care trebuie să facă mutarea.

În plus, la **jocul de reversi**, la fiecare mutare, să se modifice configurația tablei și prin întoarcerea discurilor de culoarea cu care joacă adversarul prinse între discul nou plasat și alte discuri de culoarea cu care joacă jucătorul care face mutarea, pe linie, coloană sau diagonală. Dacă numărul discurilor întoarse este 0, atunci mutarea să fie declarată nelegală și să se aștepte o mutare legală fără a se schimba configurația tablei.

La finalul jocului, o interogare cu un predicat zeroar **stop** să producă afișarea scorului pe ecran, sub forma `NumarDiscuriVerzi - NumarDiscuriRosii`.