Logică Matematică şi Computaţională – Subiecte de Examen

Claudia MUREŞAN

cmuresan@fmi.unibuc.ro, claudia.muresan@g.unibuc.ro

Universitatea din Bucureşti, Facultatea de Matematică şi Informatică

27 iunie 2025

Timp de lucru: 3 ore.

Materiale permise: orice material tipărit sau scris de mână.

Este interzisă folosirea dispozitivelor electronice.

Este interzisă părăsirea sălii de examen timp de o oră şi jumătate din momentul primirii subiectelor.

Punctaj (maxim **11,5 puncte**; nota: $\min\{10, \text{punctaj}\}$; observaţi că **nota 10** poate fi obţinută, în cazul unui punctaj maxim pentru TEMELE COLECTIVE, cu două *cerinţe reduse* de programare în Prolog, pentru **jumătate de punctaj**, dacă rezolvările sunt aproape perfecte):

- **1 punct** din oficiu;
- **3 puncte** pentru TEMELE COLECTIVE;
- fiecare dintre cele două cerinţe *numerotate* ale fiecărui exerciţiu: **1,25 puncte**.

Pentru cerinţele de programare în Prolog se poate folosi orice predicat predefinit, precum şi orice predicat scris la LABORATOR sau într–o TEMĂ COLECTIVĂ, utilizând directiva pentru includerea bazelor de cunoştinţe $labNrlmcVer.pl$ şi $temeleNr.pl$ în cea curentă, cu condiţia respectării **denumirilor predicatelor** din FIŞIERELE .PL de la LABORATOR şi din ENUNŢURILE TEMELOR COLECTIVE. Toate celelalte predicate auxiliare necesare pentru a defini predicatele cerute trebuie scrise pe lucrarea de examen. Atenţie la cerinţa ca predicatele auxiliare să poată fi aplicate pentru orice argumente de tipurile specificate: menţiunea ca acestea să fie **arbitrare**!

Fiecare coală (nu pagină) din lucrare va fi semnată cu numele şi prenumele în clar. Pe prima pagină vor fi scrise şi numărul grupei şi data examenului.

Toate subiectele vor fi tratate pe lucrarea de examen. Dacă nu aveţi nicio idee de abordare a unui exerciţiu, atunci veţi scrie noţiuni şi/sau proprietăţi teoretice din curs şi/sau predicate din laborator legate de acel exerciţiu pe lucrare. Dar **numai** în cazul în care nu ştiţi să abordaţi exerciţiul. Orice încercare de abordare a unui exerciţiu valorează mai mult, ca punctaj, decât o astfel de tratare minimală, scriind teorie sau predicate făcute la laborator, iar acestea nu se punctează în plus în cazul în care abordaţi acel exerciţiu.

Exerciţiul 1. Să se determine toate morfismele de latici mărginite $f : \mathcal{N}_5 \to \mathcal{L}_3$ de la pentagon la lanţul cu exact 3 elemente şi să se arate că niciunul nu e surjectiv:

(1) matematic;

(2) prin următoarele predicate în Prolog:

- un predicat unar $morfN5laL3(-ListaMorf)$, care determină în argumentul său $ListaMorf$ lista morfismelor de latici mărginite de la $\mathcal{N}_5$ la $\mathcal{L}_3$, folosind un predicat auxiliar care determină lista morfismelor de latici mărginite între două latici finite arbitrare;
- un predicat zeroar $niciunasurj$ care întoarce $true$ ddacă toate morfismele de latici mărginite de la $\mathcal{N}_5$ la $\mathcal{L}_3$ sunt nesurjective, care apelează predicatul $morfN5laL3(-ListaMorf)$, apoi aplică listei $ListaMorf$ un predicat care testează dacă toate funcţiile între două mulţimi finite arbitrare dintr–o listă arbitrară de funcţii între aceste mulţimi sunt nesurjective.

Pentru **jumătate din punctajul** de la **această a doua cerinţă**, puteţi scrie doar predicatul unar $morfN5laL3$ definit ca mai sus.

Exerciţiul 2. Fie E mulţimea enunţurilor logicii propoziţionale clasice. Să se demonstreze că, pentru orice $\alpha, \beta, \gamma \in E$, are loc următoarea deducţie în logica propoziţională clasică:

$$\{\neg\,\alpha \to (\beta \leftrightarrow \neg\,\gamma),\ (\alpha \wedge \beta) \leftrightarrow (\beta \wedge \gamma)\} \ \vdash\ (\beta \wedge \gamma) \to \alpha$$

(1) matematic;

(2) prin următoarele predicate în Prolog:

• un predicat ternar $ipoteza1(+Alfa, +Beta, +Gama)$, care întoarce valoarea de adevăr a enunţului compus $\neg\alpha \rightarrow (\beta \leftrightarrow \neg\gamma)$ în funcţie de valorile de adevăr $Alfa$, $Beta$, respectiv $Gama$ ale enunţurilor α, β, respectiv γ într–o interpretare arbitrară;
• un predicat binar $ipoteza2(+Alfa, +Beta, +Gama)$, care întoarce valoarea de adevăr a enunţului compus $(\alpha \wedge \beta) \leftrightarrow (\beta \wedge \gamma)$ în funcţie de valorile de adevăr $Alfa$, $Beta$, respectiv $Gama$ ale enunţurilor α, β, respectiv γ într–o interpretare arbitrară;
• un predicat binar $concluzia(+Alfa, +Beta, +Gama)$, care întoarce valoarea de adevăr a enunţului compus $(\beta \wedge \gamma) \rightarrow \alpha$ în funcţie de valorile de adevăr $Alfa$, $Beta$, respectiv $Gama$ ale enunţurilor α, β, respectiv γ într–o interpretare arbitrară;
• un predicat zeroar $deductia$, care întoarce $true$ ddacă din cele trei ipoteze de mai sus se deduce concluzia de mai sus pentru orice triplet de valori booleene ale enunţurilor α, β, γ într–o interpretare.

Desigur, la fel ca în lecţiile de laborator, prin **valoare de adevăr** a unui enunţ $\varepsilon \in E$ într–o interpretare $h : V \rightarrow \mathcal{L}_2$, unde V este mulţimea variabilelor propoziţionale, ne referim la valoarea $f(\widetilde{h}(\varepsilon))$ a funcţiei $f \circ \widetilde{h} : E \rightarrow \{false, true\}$ în enunţul ε, unde $\widetilde{h} : E \rightarrow \mathcal{L}_2$ este prelungirea lui h la mulţimea E a enunţurilor care transformă conectorii logici în operaţii booleene, iar $f : \mathcal{L}_2 \rightarrow \{false, true\}$ este izomorfismul boolean: $f(0) = false$, $f(1) = true$.

Pentru **jumătate din punctajul** de la **această a doua cerinţă**, puteţi scrie unul dintre predicatele $ipoteza1$, $ipoteza2$ şi $concluzia$.

Exerciţiul 3. Considerăm signatura de ordinul I: $\tau = (1; 2; \emptyset)$, simbolul de operaţie unară f şi simbolul de relaţie binară R, o mulţime $A = \{a, b, c, d\}$ având $|A| = 4$ şi o structură de ordinul I de signatură τ: $\mathcal{A} = (A, f^{\mathcal{A}}, R^{\mathcal{A}})$, cu mulţimea suport A, iar $f^{\mathcal{A}} : A \rightarrow A$ şi $R^{\mathcal{A}} \subseteq A^2$, astfel încât:

- $R^{\mathcal{A}}$ este relaţia de ordine strictă a algebrei Boole $\mathcal{B} = (A, \vee, \wedge, \leq, \bar{\cdot}, a, d)$ cu mulţimea suport A, minimul a şi maximul d;
- $f^{\mathcal{A}}$ este automorfism boolean al algebrei Boole $\mathcal{B}$ de mai sus şi nu este funcţia identică a lui A.

Considerăm două variabile distincte $x, y \in Var$ şi enunţul:

$$\varepsilon = \exists\, x\, \forall\, y\, (f(x) = f(f(y)) \rightarrow \neg\, R(x, y)).$$

Să se determine funcţia $f^{\mathcal{A}}$ şi relaţia binară $R^{\mathcal{A}}$, apoi să se determine dacă $\mathcal{A} \vDash \varepsilon$:

① matematic;

② prin următoarele predicate în Prolog, pentru care mulţimea $A = \{a, b, c, d\}$ va fi introdusă ca listă de constante, iar restul argumentelor vor fi calculate în aceste predicate:

• un predicat binar $algBooleA(-MultElemA, -OrdA)$ care instanţiază variabila $MultElemA$ cu lista de constante care dă mulţimea A şi determină în argumentul $OrdA$ relaţia de ordine a algebrei Boole cu mulţimea suport A, minimul a şi maximul d, folosind un predicat auxiliar pentru determinarea relaţiilor de ordine $\leq$ pe o mulţime finită arbitrară M (dată ca listă de constante Prolog) cu proprietatea că $(M, \leq)$ sunt poseturile subiacente unor algebre Boole cu minimul şi maximul date de două elemente arbitrare furnizate predicatului;
• un predicat unar $detR(-RelR)$, care apelează predicatul $algBooleA(-MultElemA, -OrdA)$, apoi determină în argumentul său $RelR$ relaţia de ordine strictă asociată relaţiei de ordine $OrdA$, folosind un predicat auxiliar care determină relaţia de ordine strictă asociată relaţiei de ordine a unui poset finit arbitrar;
• un predicat unar $detf(-Fctf)$, care apelează predicatul $algBooleA(-MultElemA, -OrdA)$, apoi determină în argumentul său $Fctf$ funcţia de la $MultElemA$ la $MultElemA$ care este izomorfism boolean de la algebra Boole cu posetul subiacent $(MultElemA, OrdA)$ la ea însăşi şi nu este funcţia identică, folosind un predicat auxiliar care determină izomorfismele booleene între două algebre Boole finite arbitrare date prin poseturile subiacente acestora şi unul care verifică dacă o funcţie de la o mulţime finită arbitrară M la M este diferită de id_M;
• un predicat zeroar $verifAsatepsilon$, care întoarce $true$ ddacă $\mathcal{A} \vDash \varepsilon$, efectuând o demonstraţie semantică, prin testarea perechilor de valori din mulţimea A pentru variabilele x, y într–o interpretare arbitrară (**indicaţie:** atenţie la reprezentarea unei funcţii ca relaţie binară funcţională totală, deci ca listă de perechi, care poate fi furnizată unui predicat unar într–o singură clauză, versus reprezentarea ei prin atâtea clauze pentru un predicat binar câte elemente are domeniul acelei funcţii; a doua reprezentare e suficientă pentru cerinţa redusă de mai jos, dar predicatele pentru generare şi colectare de funcţii scrise la laborator folosesc prima dintre aceste reprezentări).

Pentru **jumătate din punctajul** de la **această a doua cerinţă**, puteţi scrie doar predicatul zeroar $verifAsatepsilon$, introducând direct operaţia unară $f^{\mathcal{A}}$ şi relaţia binară $R^{\mathcal{A}}$ în baza de cunoştinţe.