

Programare logică și funcțională

INFORMATICĂ română, sem. 3

Cadre didactice îndrumătoare

Prof. dr. CZIBULA Gabriela ([gabriela.czibula\[at\] ubbcluj.ro](mailto:gabriela.czibula@ubbcluj.ro))

Lect. dr. ONEȚ-MARIAN Zsuzsanna ([zsuzsanna.onet\[at\] ubbcluj.ro](mailto:zsuzsanna.onet@ubbcluj.ro))

Lect. dr. MIHOLCA Diana-Lucia ([diana.miholca\[at\].ubbcluj.ro](mailto:diana.miholca[at].ubbcluj.ro))

ACS. ALBU Alexandra-Ioana ([alexandra.albu\[at\].ubbcluj.ro](mailto:alexandra.albu[at].ubbcluj.ro))

I. Obiective

1. Să deprindă studentul cu noi paradigme de programare (programarea logică și programarea funcțională).
2. Să introducă câte un limbaj de programare pentru fiecare din aceste paradigme (Prolog și Common Lisp).
3. Să inducă ideea utilizării acestor paradigme în funcție de necesitățile aplicațiilor.
4. Să asigure baza necesară urmăririi unor cursuri avansate.

II. Conținut

1. Introducere

Programare și limbaje de programare. Programare imperativă vs. programare declarativă.
Recursivitate

2. Programare Logică. Limbajul PROLOG

- 2.1. Elemente fundamentale ale limbajului Prolog. Fapte și reguli Prolog. Întrebări. Strategia de control în Prolog. Variabile și propoziții compuse. Variabile anonime. Reguli de definire a potrivirilor. Model de flux. Exemple.
- 2.2. Programul Prolog. Domenii predefinite. Întrebări. Predicate cu aritate multiplă. Simbolul IF (Prolog) și instrucțiunea IF (alte limbaje). Expresii aritmetice și comparații.
- 2.3. Controlarea backtracking-ului în Prolog. Predicatele fail și ! (cut). Utilizarea lui !. Tipuri de tăieturi. Predicatul "not". Liste Prolog. Recursivitate. Exemple de tratare a backtracking-ului. Găsirea tuturor soluțiilor în același timp. Exemple de predicate Prolog. Predicate nedeterminate.
- 2.4. Liste eterogene. Arbori. Backtracking cu ciclări. Exemple de proceduri recursive. Cadrul stivei. Optimizarea prin recursivitate de coadă. Utilizarea tăieturii pentru păstrarea recursivității de coadă.
- 2.5. Backtracking în Prolog.

3. Programare Funcțională. Limbajul LISP

- 3.1. Introducere. Importanța programării funcționale ca nouă metodologie de programare. Istoric și prezentare a limbajului LISP.
- 3.2. Elemente de bază Lisp. Structuri dinamice de date. Reguli sintactice și semantice. Clasificarea funcțiilor Lisp. Funcții primitive în Lisp. Predicate de baza în Lisp. Predicate pentru liste; pentru numere. Funcții logice și aritmetice. Definirea funcțiilor utilizator. Ramificarea prelucrărilor.

Metoda variabilei colectoare. Exemple.

- 3.3. Gestiunea simbolurilor. Alte funcții de acces la liste. OBLIST și ALIST. Funcții cu caracter destructiv. Comparații. Exemple.
- 3.4. Mecanisme definiționale evolute. Forma EVAL. Forme funcționale, funcțiile FUNCALL și APPLY. Expresii LAMBDA. Generatori, argumente funcționale. Funcții MAP. Forme iterative. Exemple.
- 3.5. Alte elemente ale limbajului Lisp. Structuri de date. Macrodefiniții. Argumente opționale. Exemple.

III. Bibliografie

1. Czibula, G., Pop, H.F., *Elemente avansate de programare în Lisp și Prolog. Aplicații în Inteligența Artificială.*, Ed. Albastră, Cluj-Napoca, 2012

Bibliografie opțională

2. FIELD A., Functional Programming, Addison Wesley, New York, 1988.
3. GIUMALE G., et. al., LISP, 2 Volume, Editura Tehnica, Bucuresti, 1987.
4. HOGER C.J., Introduction to Logic Programming, Academic Press, New York, 1984.
5. KLEENE S.E., Object Oriented Programming in Common Lisp, Addison Wesley, New York, 1m
6. PÂRV B., Alexandru Vancea, Fundamentele limbajelor de programare, Litografia Universitatii Babes-Bolyai Cluj-Napoca, 1992.
7. REEDE C., Elements of Functional Programming, Addison Wesley, New York, 1989.
8. WINSTON P.H., Lisp, Addison Wesley, New York, 2nd edition, 1984.
9. WINSTON P.H., Artificial Intelligence, Addison Wesley, New York, 2nd edition, 1984.
10. <http://www.ifcomputer.com/PrologCourse>, Lecture on Prolog
11. <http://www.lpa.co.uk>, Logic Programming

IV. Lucrarea scrisă

Colocviul la PLF va consta în lucrare scrisă PROLOG+LISP și va avea loc în **S13/S14**, în cadrul orelor de curs.

Detalii legate de desfășurare examenului scris vor fi comunicate ulterior.

Proba practică PROLOG va avea loc în **S7/S8** la laborator, iar proba practică LISP va avea loc în **S13/S14** la laborator. Proba practică va consta în rezolvarea unei probleme LISP/PROLOG și va dura **45 minute** (a se vedea **Secțiunea XII**).

V. Desfășurarea activităților didactice

Activitățile didactice se vor desfășura conform orarului oficial afișat pe pagina facultății.

VI. Situația notelor/prezențelor

Situația prezențelor și notelor la activitățile didactice (seminar, laborator, probe practice) o găsiți la adresa

https://docs.google.com/spreadsheets/d/e/2PACX-1vSCq91YxxR7_2rIrKMuHcwtp0eqiHsHB_CDqOWST598WSLwTHE7pY3eRkK55mRy9OmzMGvk-HBNNXn/pubhtml?gid=1675021479&single=true

Acest document va fi folosit și pentru afișarea punctajelor la colocviu/rezultatelor finale la PLF. Codul folosit pentru identificare este codul unic de identificare din Academic Info.

VII. Planificarea cursurilor

Curs 1	Introducere. Recursivitate
Curs 2-6	Prolog
Curs 7-12	Lisp
Curs 13/14	Examen

VIII. Activitatea de seminar

- (1) Prezența la activitatea de seminar este **OBLIGATORIE** în proporție de **75%** (minim 5 seminarii din 7). Prezența la seminar poate fi recuperată cu o altă grupă (pe parcursul celor 2 săptămâni alocate seminarului), cu acordul cadrului didactic.
 - Pentru activitatea studenților din cadrul orelor de seminar se va acorda un “bonus” la nota finală (maxim 0.5 puncte, acordate proporțional cu numărul de activități).
- (2) În caz de boală, motivarea absenței se va face de către cadrul didactic îndrumător de seminar pe baza adeverinței medicale. Se acceptă adeverințe medicale în **maxim o săptămână** după absență.
- (3) Studenții care nu au minim 5 prezențe la activitatea de seminar, **NU** se pot prezenta la lucrarea scrisă (nici în S13/S14, nici în sesiunea de restanțe) și **NU** pot promova examenul.

IX. Activitatea de laborator

IX.1 Programarea Laboratoarelor

Lab	Descrierea temei	Săptămâna de primire-predare a temei
R1	Programare recursivă (laborator opțional)	--
P1	Liste în Prolog (I)	1/2 - 3/4

P2	Liste în Prolog (II)	3/4 - 5/6
P3	Utilizarea facilității de backtracking in Prolog	5/6 - 7/8
T1	PROBA PRACTICĂ PROLOG	7/8 - o oră de laborator
L1	Programare recursivă în Lisp (I)	7/8 - 9/10
L2	Programare recursivă în Lisp (II)	9/10 - 11/12
L3	Utilizarea funcțiilor MAP în Lisp	11/12 - 13/14
T2	PROBA PRACTICĂ LISP	13/14 – o oră de laborator

IX.2 Lucrările de laborator și notarea acestora

(1) Părțile componente ale unei lucrări de laborator sunt enumerate mai jos. Acestea pot fi incluse direct în codul sursă (sub forma de comentarii) sau se pot include într-un document separat (redactat la calculator sau scris de mână, la alegere).

1. Descrieri formale:

- PROLOG – modelele recursive de rezolvare, modelul de flux al tuturor predicatelor folosite, semnificația argumentelor predicatelor.
- LISP – modelele matematice (formulele recursive de descriere a funcțiilor)

2. Codul sursă Prolog/Lisp al funcțiilor/predicatelor

3. Exemple de testare care să parcurgă cât mai cazuri de testare pentru funcțiile/predicatele de bază.

(2) Lucrarea de laborator se va prezenta cadrului didactic îndrumător de laborator, în momentul predării lucrării de laborator. În momentul predării lucrării de laborator, se va putea primi o cerință suplimentară, care va trebui rezolvată pe loc. Lucrarea de laborator se evaluează cu o notă de la 0 la 10 astfel:

2 puncte	modele recursive (componenta 1)
1.5 puncte:	codul sursă = componenta 2
0.5 puncte:	exemple = componenta 3
1 punct:	verificarea execuției programului realizat
6 puncte:	explicații (modele recursive, cod sursă) (eventual, 4 puncte) cerință suplimentară de modificare în cod, rezolvată pe loc

(3) O lucrare de laborator **COPIATA** se notează cu **0**.

(4) În cazul în care se întârzie un laborator **nota finală se înmulțește cu 0.6**; dacă se întârzie mai mult, nota finală este **0 (zero)**.

(5) În caz de boală, motivarea absenței se va face de către cadrul didactic îndrumător de laborator pe baza adeverinței medicale. Se acceptă adeverințe medicale în **maxim o săptămână** după absență.

IX.3 Software PROLOG pentru orele de laborator

Se va folosi SWI Prolog. (Web <http://www.swi-prolog.org/>, disponibil pe pagina cursului).

- Editoare: editorul SWI, Notepad++, Sublime, etc.

Puteți folosi Eclipse ca mediu de dezvoltare, folosind un plugin Prolog (de ex. cel descris în fișierul [Class Materials/Lab/Config/PrologPlugin_Eclipse/readme.txt](#)).

IX.4 Software LISP pentru orele de laborator

Se va folosi GNU CLisp (Web: <http://clisp.cons.org/>, disponibil pe pagina cursului).

- Editoare: Notepad++, Sublime, etc.

Puteți folosi Eclipse ca mediu de dezvoltare, folosind un plugin Lisp (de ex. cel descris în fișierul [Class Materials/Lab/Config/LispPlugin_Eclipse/readme.txt](#)).

X. Acordarea notei finale

NL (10%)	Nota laborator
NT1 (15%)	Nota test practic Prolog
NT2 (15%)	Nota test practic Lisp
E (60%)	Nota lucrare scrisă (Prolog+Lisp)

Pentru promovare sunt necesare următoarele criterii minime:

1. nota minim **5** la lucrarea scrisă.
2. nota finală minim **5**.

XI. Reguli de laborator

- (1). **Prezența la activitatea de laborator** este **OBLIGATORIE** în proporție de **90%** (minim 6 laboratoare din 7). **Studentii care nu au minim 6 prezențe** la activitatea de laborator, **NU** se pot prezenta la lucrarea scrisă (nici în S13/S14, nici în sesiunea de restanțe) și **NU** pot promova examenul.
- (2) Este obligatorie prezența cu semigrupa de laborator indicată în orar. Nu se acceptă recuperare cu altă semigrupa de laborator.
- (3) În caz de boală, motivarea absenței se va face de către cadrul didactic îndrumător de laborator pe baza adeverinței medicale. Se acceptă adeverințe medicale **în maxim o săptămână** după absență.
- (4) În cadrul unei ore de laborator se pot preda **cel mult 2 teme de laborator**. De asemenea, laboratoarele de Prolog se pot preda DOAR până în săptămâna 7/8.
- (5) Nota de laborator NL se va calcula ca fiind media aritmetică a notelor pe cele 6 lucrări de laborator. În cazul nepredării unei lucrări de laborator, nota corespunzătoare va fi 0.

XII. Notarea testelor practice (NT1 si NT2)

Probele practice vor consta în rezolvarea unei probleme PROLOG/LISP și vor dura **45** minute. Problemele la proba practică PROLOG vor fi în genul problemelor din laboratoarele P1 sau P2-a, iar problemele la proba practică LISP vor fi în genul problemelor din laboratoarele L1 sau L2 (doar arbore în forma 2).

ABSENȚA de la proba practică sau **COPIATUL** determină notarea cu **0 (ZERO)** a probei practice.

XIII. Participarea în sesiunea de restanțe

În cazul participării în sesiunea de restanțe, calculul notei se va face conform punctului **X**.

- studenții care nu au prezență la minim 5 seminarii și 6 laboratoare **NU** se pot prezenta în sesiunea de restanțe;
- probele practice **NU** pot fi susținute în sesiunea de restanțe;
- temele de laborator **NU** pot fi predate în sesiunea de restanțe;
- lucrarea scrisă poate fi susținută în sesiunea de restanțe, în caz de nepromovare în sesiunea normală sau pentru mărire de notă.