

INFORMAȚII PUBLICE PRIVITOARE LA CONCURSURI

Denumire câmp	Descriere
Facultatea	Matematică și Informatică
Departamentul	Informatică
Poziția în statul de funcții	100
Funcția	Asistent universitar
Disciplinele din încercătura postului	Algoritmica grafelor (în lb. română) Limbaje formale si tehnici de compilare (în lb. română) Programare logica si functionala (în lb. engleză) Arhitectura calculatoarelor (în lb. engleză)
Domeniul științific	Informatică
Descrierea postului scos la concurs	Asistent, 100, Departamentul de Informatică. Postul de asistent universitar presupune desfășurarea de activități didactice, de cercetare științifică și de îndrumare a studenților, precum și efectuarea de servicii pentru comunitatea academică. Candidații la ocuparea postului vacant de asistent universitar trebuie să aibă palmaresul științific în concordanță cu standardele domeniului Informatică și cu disciplinele postului. De asemenea, candidații trebuie să facă dovada stăpânirii limbii engleze prin documente depuse la dosar (nivel C1 sau documente atestând studii sau stagii de cercetare cumulate de cel puțin 9 luni în străinătate, în instituții de învățământ sau cercetare unde comunicarea s-a făcut în limba engleză).
Atribuții	Activitatea didactică: seminar, laborator, proiecte, consultații, lucrări de control, examene, elaborarea de materiale didactice pentru disciplinele din post. Activitatea de cercetare științifică: <ul style="list-style-type: none"> • participarea la cel puțin un seminar de cercetare în cadrul facultății; • participarea la competiții pentru obținerea de granturi de cercetare științifică; • publicarea, în fiecare perioadă de 3 ani, a cel puțin 3 articole/studii indexate BDI (Mathematical Reviews/ MathSciNet, ZMath (Emis), Computing Reviews, IEEE Xplore, DOAJ, SCOPUS, DBLP); • îndeplinirea cerințelor din anexa la fisa postului, cel puțin la nivelul calificativului -Satisfăcător-. Activitatea de îndrumare a studenților: îndrumare de lucrări de diplomă, tutore la o formație de studenți, îndrumarea acestora pentru participarea la activitatea cercurilor științifice și la concursuri studentești. Servicii pentru comunitatea academică: participare la acțiunile desfășurate de departament, facultate și universitate (promovarea admiterii, colaborarea cu mediul economic, etc).
Data și ora susținerii prelegerii/ probei orale	13.07.2026 ora 10.00

Locul susținerii prelegerii/ probei orale (adresa Facultății/ Institutului și sala)	Departamentul de Informatică, str. Teodor Mihali nr. 58-60, sala L320
Descrierea probei orale	<p>Proba orală constă în prezentarea unui proiect de seminar/ laborator/ lucrări practice.</p> <p>Comisia stabilește, pe baza tematicii și bibliografiei de concurs, tema prezentării probei orale și o comunică candidaților cu 48 de ore înaintea susținerii probei prin e-mail și prin afișarea pe pagina web a facultății, cu menționarea datei și orei afișării, sub semnătura președintelui comisiei de concurs.</p> <p>Durata minimă a probei orale susținute de către candidat este de 30 de minute; proba conține în mod obligatoriu și o sesiune de întrebări din partea comisiei și/ sau a publicului.</p> <p>Dacă sunt mai mulți candidați, comisia va decide ordinea în care aceștia vor susține proba orală.</p>
Data și ora evaluării dosarului candidatului	N/A
Locul evaluării dosarului candidatului (adresa Facultății/ Institutului și sala)	N/A
Descrierea evaluării dosarului candidatului	N/A
Data și ora susținerii probei scrise	13.07.2026 ora 9.00
Locul susținerii probei scrise (adresa Facultății/ Institutului și sala)	Departamentul de Informatică, str. Teodor Mihali nr. 58-60, sala L320
Descrierea probei scrise	Evaluarea cunoștințelor fundamentale în domeniul Informatică prin examen scris din tematica anunțată.
Tematica și bibliografia probelor de concurs	<p>Proba 1 – Probă scrisă</p> <p>Tematică:</p> <p>A. Fundamentele programării Subalgoritmi: specificare, testare. Clase de algoritmi: căutare, sortare, interclasare. Metode de proiectare a algoritmilor: top-down, rafinare succesivă. Subprograme, apel și modalități de transmitere a parametrilor (prin valoare și referință). Tehnici de programare: Backtracking, Divide et impera, Greedy. Programare modulară: modul, interfață, implementare; concretizare în C/C++, Java, Python.</p> <p>B. Programare orientată pe obiecte Clase, obiecte. Moștenire, polimorfism. Programare bazată pe interfețe.</p>

C. Structuri de date

Tipuri abstracte de date (TAD).

TAD Mulțime, Colectie, Listă, Stivă, Coadă, Dicționar.

Specificarea TAD.

Implementări pentru TAD folosind: vectori, liste înlănțuite, arbori binari.

Bibliografie:

1. M. Frențiu, B. Pârv, Elaborarea programelor. Metode și tehnici moderne, ProMedia, Cluj-Napoca, 1994
2. M. Frențiu, H.F. Pop, G. Șerban, Programming fundamentals, Cluj University Press, 2006
3. T. Cormen, C. Leiserson, R. Rivest: Introducere în algoritmi. Cluj-Napoca: Editura Computer Libris Agora, 2000
4. B. Eckel, Thinking in C++, vol I și II, <http://www.mindview.net>
5. B. Eckel, Thinking in Java, <http://www.mindview.net>
6. M.A. Ellis, B. Stroustrup, The annotated C++ reference manual, Addison-Wesley, 1994
7. The Python language reference.
8. <http://docs.python.org/py3k/reference/index.html>
9. R.S. Pressman, Software engineering. A practitioner's approach, 6th ed., McGraw-Hill, 2005

Proba 2 – Probă orală: susținerea unui proiect de seminar/ laborator/ lucrări practice

Tematică:

A. Algoritmica grafelor (în lb. română)

1. Noțiuni de bază (graf, multigraf, graf/multigraf orientat, drum, drum elementar, lanț simplu), reprezentări ale grafelor, grafe tare conexe, conexe (alg. pentru determinarea componentelor conexe).
2. Studiu aprofundat al reprezentării grafelor. Drumuri în grafe: lungimea unui drum (matricea distanțelor, centru, raza, diametru), valoarea unui drum, optimizări în mulțimea drumurilor. algoritmul lui Moore-Dijkstra.
3. Algoritmul lui Bellman-Kalaba, algoritmul lui Ford, algoritmi matriceali (Floyd-Hu, Dantzing, Floyd-Hu-Warshall), drum ciclic, drumuri Euleriene, drumuri Hamiltoniene.
4. Conectivitate și probleme de lanț minim. Parcurgeri de graf în lațime și adâncime.
5. Numere fundamentale în teoria grafelor: număr de stabilitate internă, algoritm pentru determinarea mulțimilor interior stabile, număr de stabilitate externă, algoritm pentru determinarea mulțimilor exterior stabile, număr cromatic, număr ciclomatic.
6. Arbori și păduri: noțiuni generale, algoritmi lui Kruskal și Prim.
7. Cuplaje în grafe: definiții, algoritm pentru determinarea cuplajului maxim, algoritm pentru determinarea cuplajului de pondere maximă.
8. Probleme extremale (teoremele lui Ramsey și Turan).

9. Probleme grele: ciclul Hamiltonian, problema comis voiajorului. Probleme de numărare și enumerare.
10. Probleme grele: clique, vertex cover, colorare.
11. Ciclul elementar Eulerian. Grafe planare: relația lui Euler, grafe K_5 și $K_{3,3}$, relații între numărul de muchii și noduri.
12. Rețele de transport.
13. Fluxuri în rețele de transport: definiții de bază, algoritmul lui Ford-Fulkerson, extensii ale algoritmului lui Ford-Fulkerson, fluxuri de cost minim.
14. Probleme de cuplaj

Bibliografie:

1. Berge C., Graphes et hypergraphes, Dunod, Paris 1970.
2. B. ANDRÁSFALVI: Introductory graph theory, Akademiai Kiado - North Holland, 1987.
3. Berge C., Teoria grafurilor și aplicațiile ei, Ed. Tehnica, 1972
4. T. Toader, Grafe. Teorie, algoritmi și aplicații, Ed. Alabastru, Cluj-N (ed. I, II, III), 2002 și 2009
5. KÁSA ZOLTÁN, Combinatorică cu aplicații, Presa Universitară Clujeană, 2003.
6. Cormen, Leiserson, Rivest, Introducere în algoritmi, Editura Computer Libris Agora, 2000.
7. Rosu A., Teoria grafurilor, algoritmi, aplicații. Ed. Militară, 1974.
8. Ciurea E., Ciupala L., Algoritmi - algoritmi fluxurilor în rețele, Ed. Matrix Rom, 2006.
9. KÁSA Z., TARTIA C., TAMBULEA L.: Culegere de probleme de teoria grafurilor, Lito. Univ. Cluj-Napoca 1979.
10. CATARANCIUC S., IACOB M.E., TOADERE T., Probleme de teoria grafurilor, Lito. Univ. Cluj-Napoca, 1994.
11. TOMESCU I., Probleme de combinatorică și teoria grafurilor. Ed. Did. și Pedagog. București 1981.

B. Limbaje formale și tehnici de compilare ((în lb. română)

1. Limbaje formale, compilator: prezentare generală. Analiza lexicală.
2. Automate finite
3. Limbaje regulate: teoreme echivalență, proprietăți de închidere, lema de pompare
4. Expresii regulate. Generatoare de analizoare lexicale
5. Limbaje independente de context. Proprietăți de închidere. Lema de pompare.
6. Generatoare de analizoare sintactice. Automate push-down
7. Analiza sintactică descendentă.
8. Analiza sintactică ascendentă.
9. Gramatici de atribute. Cod intermediar.
10. Automate și traductoare. Mașini Turing.
11. Gramatici și automate.

Bibliografie:

1. A.V. AHO, D.J. ULLMAN - Principles of computer design, Addison-Wesley, 1978.
2. A.V. AHO, D.J. ULLMAN - The theory of parsing, translation and compiling, Prentice-Hall, Engl. Cliffs., N.J., 1972, 1973.
3. D. GRIES - Compiler construction for digital computers,, John Wiley, New York, 1971.
4. MOTOGNA, S. – Metode de proiectare a compilatoarelor, Ed. Albastra, 2006
5. SIPSER, M., Introduction to the theory of computation, PWS Pub. Co., 1997.
6. L.D. SERBANATI - Limbaje de programare si compilatoare, Ed. Academiei RSR, 1987.
7. GRUNE, DICK - BAL, H. - JACOBS, C. - LANGENDOEN, K.: Modern Compiler Design, John Wiley, 2000

C. Programare logica și funcțională (în lb. engleză)

1. Recursivitate în Pseudocod
2. Liste în Prolog
3. Arbori în Prolog. Gestiunea listelor (eterogene) în Prolog.
4. Backtracking în Prolog
5. Prelucrarea listelor în LISP
5. Programare recursivă în Lisp
6. Funcții MAP în LISP.

Bibliografie:

1. CZIBULA G., POP H.F., Elemente avansate de programare in Lisp si Prolog. Aplicatii in Inteligenta Artificiala, Editura Albastra, Cluj-Napoca, 2012
2. POP H.F., SERBAN G., Programare in Inteligenta Artificiala - Lisp si Prolog, Editura Albastra, ClujNapoca, 2003
3. <http://www.ifcomputer.com/PrologCourse>, Lecture on Prolog
4. <http://www.lpa.co.uk>, Logic Programming
5. FIELD A., Functional Programming, Addison Wesley, New York, 1988.
6. WINSTON P.H., Lisp, Addison Wesley, New York, 2nd edition, 1984.

D. Arhitectura calculatoarelor (în lb. engleză)

1. Reprezentarea datelor: date elementare, reprezentari binare si ordini de plasare, organizarea si memorarea datelor
2. Codificarea caracterelor, reprezentarea cu semn si fara semn, cod complementar, conversii, conceptul de depasire
3. Arhitectura sistemelor de calcul: organizarea unui SC, unitatea centrala, ceasul sistem, calculator pe n biti, memoria, dispozitivele periferice
4. Performantele unui SC, arhitectura microprocesorului 80x86 – structura,

- registri, calculul de adresa, moduri de adresare, adrese far si near
5. Unitatea executiva (EU) a microprocesorului 80x86: rolul si functiile registrilor si al flagurilor. Clasificare (Registrii si Flaguri) si studii de caz.
 6. Unitatea BIU a microprocesorului 80x86: registrii de adresa, registrii de segment, reprezentarea instructiunilor. Formula de specificare offset pe 32 biti si formula de specificare offset pe 16 biti.
 7. Elementele limbajului de asamblare: formatul unei linii sursa, expresii, tipuri de accesare a operanzilor, operatori. Conversii nondistructive (si operatorii specifici)
 8. Directive pentru definirea segmentelor, pt.definirea datelor, directivele EQU și INCLUDE, macrouri
 9. Instructiuni ale limbajului de asamblare: instructiuni de transfer, conversii, operatii aritmetice cu semn si fara semn, operatii de deplasare si rotire de biti, operatii logice pe biti
 10. Instructiuni de salt conditionat si neconditionat, instructiuni de ciclare, instructiuni pe siruri. Conceptul de depășire și modul în care arhitectura 80x86 se comportă
 11. Reprezentarea instructiunilor mașină. Formatul intern al unei instructiuni. Prefixe de instructiuni.
 12. Programarea multimodul ASM-ASM: directivele de import-export GLOBAL si EXTERN. Legarea de module scrise in limbaj de asamblare si modul de comunicare între acestea
 13. Implementarea apelului de subprograme. Convenții de apel: CDECL și STDCALL, cod de apel, cod de intrare, cod de iesire la nivelul limbajelor de nivel inalt vs. Limbaj de asamblare
 14. Legarea de module NASM cu module scrise în limbaje de nivel înalt (studiu de caz – programarea C).

Bibliografie:

1. Al. Vancea, F. Boian, D. Bufnea, A. Andreica, A. Darabant, A. Navroschi – Arhitectura calculatoarelor. Limbajul de asamblare 80x86., Editura Risoprint, Cluj-Napoca, 2014.
2. Al. Vancea, F. Boian, D. Bufnea, A. Gog, A. Darabant, A. Sabau – Arhitectura calculatoarelor. Limbajul de asamblare 80x86., Editura Risoprint, Cluj-Napoca, 2005.
3. A. Gog, A. Sabau, D. Bufnea, A. Sterca, A. Darabant, Al. Vancea – Programarea în limbaj de asamblare 80x86. Exemple si aplicatii., Editura Risoprint, Cluj-Napoca, 2005
4. Randal Hyde – The Art of Assembly Programming, No Starch Press, 2003. (<http://homepage.mac.com/randyhyde/webster.cs.ucr.edu/www.artofasm.com/DOS/index.html>)
5. Boian F.M. Vancea A. Arhitectura calculatoarelor, suport de curs. Facultatea de Matematica si Informatica, Centrul de Formare Continua si Invatamânt la Distanta., Ed. Centrului de Formare Continua si Invatamânt la Distanta, Cluj, 2002
6. Irvine, K.R., 2015. Assembly language for x86 processors.
7. Kusswurm, D., 2014. Modern X86 Assembly Language Programming.

	<p>Springer.</p> <p>8. Carter, P.A., 2004. PC Assembly Language. Github: (http://pacman128.github.io/static/pcasm-book.pdf)</p> <p>9. Cavanagh, J., 2013. X86 Assembly Language and C Fundamentals. CRC Press.</p> <p>10. Guide, P., 2011. Intel® 64 and ia-32 architectures software developer's manual. Volume 3B: System programming Guide, Part, 2, p.11. (http://www.facweb.iitkgp.ac.in/~goutam/compiler/readingMaterial/intelXeon/253665.pdf)</p> <p>10. Guide, P., 2011. Intel® 64 and ia-32 architectures software developer's manual. Volume 3B: System programming Guide, Part, 2, p.11. (http://www.facweb.iitkgp.ac.in/~goutam/compiler/readingMaterial/intelXeon/253665.pdf)</p>
<p>Descrierea procedurii de concurs</p>	<p>Comisia de concurs evaluează candidații ținând sub următoarele aspecte:</p> <ul style="list-style-type: none"> • Conținutul dosarului individual; • Proba orală (proba 1) • Proba scrisă (proba 2). <p>În evaluarea activității științifice se va ține cont de calitatea publicațiilor și contribuțiile candidaților în raport cu exigențele prevăzute în norma didactică.</p> <p>În cazul tuturor posturilor, durata minimă a probei orale/ prelegerii susținute de către candidat este de 30 de minute; proba conține în mod obligatoriu și o sesiune de întrebări din partea comisiei.</p> <p>În cazul unui post cu încărcătură integrală într-o singură limbă străină, probele de concurs vor fi susținute în limba respectivă în fața comisiei de concurs; în cazul unui post cu încărcătură în mai multe limbi, probele de concurs vor fi susținute în limbile respective, conform unei proceduri stabilite prin anunțul de concurs.</p> <p>Nota finală a fiecărui candidat se calculează ca medie aritmetică a notelor obținute la criteriile de mai sus. Fiecare membru al comisiei (inclusiv președintele) întocmește un referat individual de apreciere care propune o notă finală pentru fiecare candidat.</p> <p>Candidații eligibili pentru ocuparea postului scos la concurs trebuie să obțină:</p> <ul style="list-style-type: none"> • cel puțin nota 7 (șapte) la fiecare probă; • media generală cel puțin 8,50 (opt și 50%). <p>Președintele comisiei de concurs întocmește un raport asupra concursului în care prezintă notele finale atribuite candidaților de către membrii comisiei și indică media generală obținută de fiecare candidat, calculată ca medie aritmetică a notelor finale din referatele individuale. Media generală astfel obținută reprezintă rezultatul concursului pentru fiecare candidat. Pe baza mediei generale, comisia de concurs decide ierarhia candidaților și nominalizează candidatul eligibil care a înregistrat cel mai bun rezultat în concurs. Președintele comisiei de concurs supune raportul de sinteză asupra concursului votului deschis al membrilor comisiei. În urma exercitării votului, președintele constată rezultatul votului, îl comunică membrilor comisiei și îl menționează în încheierea raportului asupra concursului, cu precizarea numărului de voturi "pentru", respectiv "contra". În cazul în care</p>

	votul "pentru" nu este acordat de majoritatea membrilor comisiei, postul scos la concurs nu este ocupat de niciun candidat. Ierarhia candidaților stabilită prin media generală nu poate fi modificată prin votul comisiei. Raportul asupra concursului este semnat de fiecare dintre membrii comisiei de concurs și de către președintele comisiei.
Perioada de comunicare a rezultatelor	<i>În aceeași zi cu încheierea lucrărilor comisiei.</i>
Perioada de depunere a contestațiilor	<i>În termen de 3 zile lucrătoare de la comunicarea rezultatului.</i>
Salariul minim de încadrare a postului la momentul angajării	7417 lei

**Director Departament,
Conf. dr. Adrian STERCA**

