

INFORMAȚII PUBLICE PRIVITOARE LA CONCURSURI

Denumire câmp	Descriere
Facultatea	Matematică și Informatică
Departamentul	Informatică
Poziția în statul de funcții	85
Funcția	Asistent universitar
Disciplinele din încărcătura postului/ariile de cercetare, așa cum figurează în statul de funcții	Ingineria sistemelor soft Limbaje formale și tehnici de compilare (în lb. engleză)
Domeniul științific	Informatică
Descrierea postului scos la concurs	<p>Asistent, 85, 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.</p> <p>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, atât 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), cât și prin susținerea probei orale în limba engleză în fața comisiei de concurs.</p>
Atribuții	<p>Activitatea didactică: seminar, laborator, proiecte, consultații, lucrări de control, examene, elaborarea de materiale didactice pentru disciplinele din post.</p> <p>Activitatea de cercetare științifică:</p> <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 4 ani, a cel puțin 5 articole/studii indexate BDI (Mathematical Reviews/ MathSciNet, ZMath (Emis), Computing Reviews, IEEE Xplore, DOAJ, SCOPUS, DBLP) din care cel puțin 2 indexate ISI sau SCOPUS sau în volumele unor conferințe internaționale relevante (ACM, IEEE, AMS, EMS). <p>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 studențești.</p> <p>Servicii pentru comunitatea academică: participare la acțiunile desfășurate de departament, facultate și universitate (promovarea admiterii, colaborarea cu mediul economic, etc).</p>

Data și ora susținerii probei orale	18.06.2020, ora 11:00
Locul susținerii probei orale	Departamentul de Informatică, str. Teodor Mihali nr. 58-60, sala C512
Probele de concurs, data, ora și locul de susținere a acestora	<p>Pentru postul de ASISTENT UNIVERSITAR, concursul constă în:</p> <ol style="list-style-type: none"> 1. evaluarea dosarului individual; 2. susținerea unei probe orale; 3. susținerea unei probe scrise. <p>Proba orală constă în prezentarea unui proiect de seminar/ laborator/ lucrări practice. 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. 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>Proba 1 – Probă scrisă: 18.06.2020, ora 8:00, Departamentul de Informatică, str. Teodor Mihali nr. 58-60, sala C512.</p> <p>Proba 2 – Probă orală: susținerea unui proiect de seminar/ laborator/ lucrări practice – 18.06.2020, ora 11:00, Departamentul de Informatică, str. Teodor Mihali nr. 58-60, sala C512. Dacă sunt mai mulți candidați, comisia va decide ordinea în care aceștia vor susține proba orală.</p> <p>Dosarul individual, proba orală și proba scrisă contează în proporții egale la nota finală acordată în referatul individual de apreciere întocmit de fiecare membru al comisiei de concurs;</p>
Tematica și bibliografia probelor de concurs	<p>Proba 1 – Probă scrisă</p> <p>Tematică:</p> <ol style="list-style-type: none"> A. Limbaje formale și tehnici de compilare 1. Compilator. Introducere 2. Analiza lexicală. Limbaje. Introducere 3. Gramatica. Clasificarea Chomsky. Automate finite 4. Limbaje regulate. Generatoare de analizoare lexicale 5. Proprietăți de închidere ale limbajelor regulate. 6. Gramatici independente de context 7. Generatoare de analizoare sintactice. Automate Push Down 8. Gramatici de atribut 9. Analiza sintactică 10. Cod intermediar . Modele de calcul 11. Aspecte recapitulative teoretice și practice. Aplicativitatea mecanismelor formale studiate în proiectarea compilatoarelor. <p>Bibliografie</p> <ol style="list-style-type: none"> 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 Pulb. Co., 1997
6. CSÖRNYEI ZOLTÁN, Bevezetés a fordítóprogramok elméletébe, I, II., ELTE, Budapest, 1996
7. L.D. SERBANATI - Limbaje de programare si compilatoare, Ed. Academiei RSR, 1987.
8. CSÖRNYEI ZOLTÁN, Fordítási algoritmusok, Erdélyi Tankönyvtanács, Kolozsvár, 2000.
9. DEMETROVICS JÁNOS-DENEV, J.-PAVLOV, R., A számítástudomány matematikai alapjai, Nemzeti Tankönyvkiadó, Budapest, 1999
10. GRUNE, DICK - BAL, H. - JACOBS, C. - LANGENDOEN, K.: Modern Compiler Design, John Wiley, 2000

B. Fundamentele programării, Programare orientată pe obiecte, Structuri de date

1. 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 in C/C++, Java, Python.

2. Programare orientată pe obiecte

Clase, obiecte.

Moștenire, polimorfism.

Programare bazată pe interfețe.

3. Structuri de date

Tipuri abstracte de date (TAD).

TAD Mulțime, Colecție, 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.
<http://docs.python.org/py3k/reference/index.html>

	<p>8. R.S. Pressman, Software engineering. A practitioner's approach, 6th ed., McGraw-Hill, 2005</p> <p>Proba 2 –Probă orală: susținerea unui proiect de seminar/ laborator/ lucrări practice</p> <p>Tematică:</p> <p>A. Ingineria sistemelor soft</p> <ol style="list-style-type: none"> 1. Introducere în ingineria sistemelor soft 2. Ciclul de viață al softului. Modele de procese soft 3. Tehnici de gestionare a complexității softului. Modelarea în ingineria sistemelor soft 4. Modelarea sistemelor soft folosind UML 5. Colectarea cerințelor 6. Analiza cerințelor 7. Proiectarea sistemelor soft: <ul style="list-style-type: none"> - concepte, principii, activități. Proiectare arhitecturii; - Proiectarea detaliată - șabloane de proiectare; - Proiectarea detaliată - specificarea interfețelor. 8. Implementarea sistemelor soft 9. Verificarea și validarea sistemelor soft 10. Gestiunea proiectelor soft 11. Metodologii de dezvoltare a sistemelor soft <p>Bibliografie</p> <ol style="list-style-type: none"> 1. BOOCH, G., RUMBAUGH, J., JACOBSON, I., The Unified Modeling Language User Guide - V.2.0, Addison Wesley, 2005. 2. BRUEGGE, B., DUTOIT, A., Object-Oriented Software Engineering Using UML, Patterns and Java - 3rd Edition, Prentice Hall, 2009. 3. FOWLER, M. et al., Refactoring - Improving the Design of Existing Code, Addison Wesley, 1999. 4. FOWLER, M., SCOTT, K., UML Distilled: A Brief Guide to the Standard Object Modeling Language -3rd ed., Addison-Wesley, 2003. 5. GAMMA, E., HELM, R., JOHNSON, R., VLISSIDES, J., Design Patterns, Addison-Wesley, 1996. 6. MARTIN, R.C., Agile Software Development: Principles, Patterns, and Practices, Prentice Hall, 2002. 7. PÂRV, B., Analiza și proiectarea sistemelor, Univ. Babeș-Bolyai, CFCID, Facultatea de Matematică și Informatică, Cluj-Napoca, 2004. 8. PRESSMAN, R.S., Software Engineering - A Practitioners Approach - 8th ed., McGraw-Hill, 2014. 9. RUMBAUGH, J., JACOBSON, I., BOOCH, G., The Unified Modeling Language Reference Manual (2nd edition), Addison Wesley, 2010. 10. SCHACH, S.R., Object-Oriented and Classical Software Engineering - 8th ed., McGraw-Hill, 2010. 11. SOMMERVILLE, I., Software Engineering - 10th edition, Addison-Wesley, 2015.
<p>Descrierea procedurii de concurs</p>	<p>Comisia de concurs evaluează candidații ținând cont de următoarele criterii:</p> <ul style="list-style-type: none"> • Conținutul dosarului individual; • Proba orală (proba 1) • Proba scrisă (proba 2). <p>Nota finală a fiecărui candidat se calculează ca medie aritmetică a</p>

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.

Candidații eligibili pentru ocuparea postului scos la concurs trebuie să obțină:

- cel puțin nota 6 (șase) la fiecare criteriu;
- nota finală cel puțin 7 (șapte) dată de fiecare referent;
- media generală cel puțin 8,50 (opt și 50%).

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 întrunit cel mai bun rezultat în concurs. Președintele comisiei de concurs supune raportul asupra concursului votului secret al membrilor comisiei. În urma exercitării votului secret, președintele constată rezultatul votului, îl comunică membrilor comisiei și îl menționează în încheierea raportului asupra concursului, cu precizarea numărului de voturi "pentru", respectiv "contra", votul fiind menținut secret. În cazul în care votul "pentru" nu este acordat de majoritatea membrilor comisiei, postul scos la concurs nu este ocupat de niciun candidat. Raportul asupra concursului este semnat de fiecare dintre membrii comisiei de concurs și de către președintele comisiei.

Director department,

Lect. dr. Adrian STERCA