

## ANEXA Nr. 4

## INFORMAȚII PUBLICE PRIVITOARE LA CONCURSURI

	Română
Universitatea	Universitatea Babeș-Bolyai din Cluj-Napoca
Facultatea / Structura organizațională de conducere	Facultatea de Matematică și Informatică
Departamentul / Unitatea organizațională	Institutul de cercetare în Inteligența Artificială, Realitate Virtuală și Robotică
Poziția în statul de funcții	7
Funcția	Asistent de cercetare științifică
Disciplinele din încercătura postului/ ariile de cercetare, așa cum figurează în statul de funcții	Cercetare fundamentală/ exploratorie/ dezvoltare-inovare legată de informatică
Domeniul științific	Informatică
Descrierea postului scos la concurs	<p>Postul de asistent de cercetare presupune desfășurarea de activități de cercetare științifică și de îndrumare a studenților, precum și efectuarea de servicii pentru comunitatea academică. Postul presupune implicare în activitatea de cercetare a unor proiecte aflate în derulare în domeniul Informatică.</p> <p>Candidații la ocuparea postului vacant de asistent de cercetare trebuie să aibă palmaresul științific în concordanță cu standardele domeniului Informatică și cu activitățile postului.</p>
Atribuții	<p>Activitatea de cercetare științifică:</p> <ul style="list-style-type: none"> <li>• participarea la cel puțin un seminar de cercetare în cadrul facultății;</li> <li>• participarea la competiții pentru obținerea de granturi de cercetare științifică;</li> <li>• 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);</li> <li>• îndeplinirea cerințelor din anexa la fișa postului, cel puțin la nivelul calificativului -Satisfăcător-.</li> <li>• contribuții la derularea și dezvoltarea proiectelor de cercetare și de transfer tehnologic.</li> </ul> <p>Activitatea de îndrumare a studenților: 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>
Perioada de înscriere la concurs	14.05.2024-13.06.2024
Data și ora susținerii probei orale	08.07.2024, 11:00
Locul susținerii probei orale (adresa Facultății/ Institutului și sala)	Institutul de cercetare în Inteligența Artificială, Realitate Virtuală și Robotică, Facultatea de Matematică și Informatică, Departamentul de Informatică, str.

	Kogalniceanu, nr. 1, sala MOS S15.
Descrierea probei orale	Proba orală constă în prezentarea unui proiect de seminar/ laborator/ lucrări practice, în funcție de natura și specialitatea postului, cu tema stabilită în condițiile art.17, alin (1), lit b) din prezenta metodologie de către comisia de concurs (Comisia stabilește, pe baza tematicii și bibliografiei de concurs, tema prezentării probei orale pentru posturile de asistent de cercetare științifică, asistent universitar, cercetător științific, lector universitar, șef de lucrări, cercetător științific gradul III 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.
Data și ora susținerii probei scrise	08.07.2024, 8:00
Locul susținerii probei scrise (adresa Facultății/ Institutului și sala)	Institutul de cercetare în Inteligența Artificială, Realitate Virtuală și Robotică, Facultatea de Matematică și Informatică, Departamentul de Informatică, str. Kogalniceanu, nr. 1, sala MOS S15.
Descrierea probei scrise	Evaluarea cunostintelor fundamentale in domeniul Informatica prin examen scris din tematica anuntata.
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 in C/C++, Java, Python.</p> <p>B. Programare orientată pe obiecte  Clase, obiecte.  Moștenire, polimorfism.  Programare bazată pe interfețe.</p> <p>C. 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.</p> <p>Bibliografie:  1. M. Frențiu, B. Pârv, Elaborarea programelor. Metode și tehnici moderne, ProMedia, Cluj-Napoca,</p>

- 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.htm>
  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. Algoritmi și programare
  1. Introducere în procese de dezvoltare software
  2. Programare procedurală
  3. Programare modulară
  4. Tipuri definite de utilizator
  5. Principii de proiectare și programare
  6. Programare orientată pe obiecte
  7. Proiectarea programelor
  8. Testarea și inspectarea programelor
  9. Recursivitate
  10. Complexitatea algoritmilor
  11. Algoritmi de căutare și sortare
  12. Metode de rezolvare a problemelor (I) – Backtracking, Greedy
  13. Metode de rezolvare a problemelor (II) - Divide & Conquer, Programare dinamică

Bibliografie:

1. M.L. Hetland, Beginning Python: From Novice to Professional, Apress, 2005.
2. M. Frențiu, H.F. Pop, Fundamentals of Programming, Cluj University Press, 2006.
3. K. Beck, Test Driven Development: By Example. Addison-Wesley Longman, 2002.
4. [http://en.wikipedia.org/wiki/Test-driven\\_development](http://en.wikipedia.org/wiki/Test-driven_development)
5. M. Fowler, Refactoring. Improving the Design of Existing Code, Addison-Wesley, 1999. <http://refactoring.com/catalog/index.html>
6. The Python Programming Language - <https://www.python.org/>
7. The Python Standard Library - <https://docs.python.org/3/library/index.html>

8. The Python Tutorial -  
<https://docs.python.org/3/tutorial/>

B. Fundamentele programarii

1. Introducere in procese de dezvoltare software
2. Programare procedurala
3. Programare modulara
4. Tipuri definite de utilizator
5. Principii de proiectare si programare
6. Programare orientata pe obiecte
7. Proiectarea programelor
8. Testarea si inspectarea programelor
9. Recursivitate
10. Complexitatea algoritmilor
11. Algoritmi de cautare și sortare
12. Metode de rezolvare a problemelor (I) – Backtracking, Greedy
13. Metode de rezolvare a problemelor (II) - Divide & Conquer, Programare dinamica

Bibliografie:

1. Kent Beck - Test Driven Development: By Example. Addison-Wesley Longman, 2002.
2. Kleinberg and Tardos – Algorithm Design. Pearson Educational, 2014
3. (<http://www.cs.princeton.edu/~wayne/kleinberg-tardos/>)
4. Martin Fowler - Refactoring. Improving the Design of Existing Code. Addison-Wesley, 1999.
5. (<http://refactoring.com/catalog/index.html>)
6. Frentiu, M., H.F. Pop, Serban G. - Programming Fundamentals, Cluj University Press, 2006
7. The Python language reference.  
(<https://docs.python.org/3/reference/index.html>)
8. The Python standard library.  
(<https://docs.python.org/3/library/index.html>)
9. The Python tutorial.  
(<https://docs.python.org/3/tutorial/index.html>)

C. Programare orientata obiect

1. Elemente de bază ale limbajului C
2. Programare modulară în C/C++
3. Programare orientată obiect în C++.
4. Tipuri de date derivate și tipuri definite de utilizator, alocare dinamică în C++. Elemente de programare generica
5. Moștenire
6. Polimorfism
7. Ierarhii de clase
8. Interfețe grafice utilizator (GUI)
9. Elemente de programare bazată pe evenimente (Evenimente: Semnale si sloturi Qt; Proiectare GUI; Callback/Observer)
10. Elemente de programare bazată pe evenimente (Componente grafice cu modele; Șablonul MVC; Modele predefinite)
11. Șabloane de proiectare

	<p>Bibliografie:</p> <ol style="list-style-type: none"> <li>1. B. Stroustrup. The C++ Programming Language, Addison Wesley, 1998.</li> <li>2. Bruce Eckel. Thinking in C++, Prentice Hall, 1995.</li> <li>3. A. Alexandrescu. Programarea moderna in C++: Programare generica si modele de proiectare aplicade, Editura Teora, 2002.</li> <li>4. S. Meyers. Effective C++: 55 Specific Ways to Improve Your Programs and Designs (3rd Edition), Addison-Wesley, 2005.</li> <li>5. S. Meyers. More effective C++: 35 New Ways to Improve Your Programs and Designs, Addison-Wesley, 1995.</li> <li>6. B. Stroustrup. A Tour of C++, Addison Wesley, 2013.</li> <li>7. C++ reference (<a href="http://en.cppreference.com/w/">http://en.cppreference.com/w/</a>).</li> <li>8. Qt Documentation (<a href="http://doc.qt.io/qt-5/">http://doc.qt.io/qt-5/</a>).</li> <li>9. E. Gamma, R. Helm, R. Johnson, J. Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Longman Publishing, 1995.</li> </ol> <p>D. Structuri de date si algoritmi</p> <ol style="list-style-type: none"> <li>1. Introducere. Structuri de date. Structuri statice, semistatice si dinamice.</li> <li>2. Tipuri de date: domeniu, operatii si reprezentarea datelor</li> <li>3. Tabloul</li> <li>4. Abstract Data Types - TAD Colectie, TAD Dictionar, TAD Stiva si Coadă, TAD Coadă cu prioritati</li> <li>5. Linked Lists - TAD Lista, Lista înlantuita</li> <li>6. Ansamblul (heap)</li> <li>7. Hash Table - Tabela de dispersie</li> <li>8. TAD Arbore - Arborele binar</li> <li>9. Arborele binar de cautare</li> <li>10. Arbori binari de cautare echilibrati</li> <li>11. Aplicații și structuri de date în diferite limbaje de programare (Python, C++, Java, C#)</li> </ol> <p>Bibliografie:</p> <ol style="list-style-type: none"> <li>1. T. Cormen, C. Leiserson, R. Rivest, C. Stein: Introduction to algorithms, Third Edition, The MIT Press, 2009</li> <li>2. S. Skiena: The algorithms design manual, Second Edition, Springer, 2008</li> <li>3. N. Karumanchi: Data structures and algorithms made easy, CareerMonk Publications, 2016</li> <li>4. M. A. Weiss: Data structures and algorithm analysis in Java, Third Edition, Pearson, 2012</li> <li>5. R. Sedgwick: Algorithms, Addison-Wesley Publishing, 1984</li> </ol>
Descrierea procedurii de concurs	<p>Comisia de concurs evaluează candidații ținând cont de următoarele criterii:</p> <ul style="list-style-type: none"> <li>• Conținutul dosarului individual;</li> </ul>

	<ul style="list-style-type: none"> <li>• Proba orală (proba 1)</li> <li>• Proba scrisă (proba 2).</li> </ul> <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. Candidații eligibili pentru ocuparea postului scos la concurs trebuie să obțină:</p> <ul style="list-style-type: none"> <li>• cel puțin nota 6 (șase) la fiecare criteriu;</li> <li>• nota finală cel puțin 7 (șapte) dată de fiecare referent;</li> <li>• media generală cel puțin 8,50 (opt și 50%).</li> </ul> <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 înrunit 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 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.</p>
Perioada de comunicare a rezultatelor	8.07.2024-9.07.2024
Perioada de depunere a contestațiilor	10.07.2024-12.07.2024
Salariul minim de încadrare a postului la momentul angajării	6325
Lista completă a documentelor pe care candidații trebuie să le includă în dosarul de concurs	<a href="https://www.ubbcluj.ro/ro/infoubb/posturi_vacante/posturi_didactice_perioada_nedeterminata">https://www.ubbcluj.ro/ro/infoubb/posturi_vacante/posturi_didactice_perioada_nedeterminata</a>
Adresa la care trebuie trimis dosarul de concurs	Registratura Universității “Babeș-Bolyai”, (camera P20), str. M. Kogălniceanu nr. 1, Cluj-Napoca