

## ANEXA Nr. 4

## INFORMAȚII PUBLICE PRIVITOARE LA CONCURSURI

Denumire câmp	Descriere
Facultatea	Matematică și Informatică
Departamentul	Institutul de cercetare în Inteligența Artificială, Realitate Virtuală și Robotică
Poziția în statul de funcții	4
Funcția	Asistent de cercetare (pe perioadă determinată 3 ani)
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>Asistent de cercetare (normă 4 ore/zi)</p> <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 fisa 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 studentești.</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 prelegerii/probei orale	27 ianuarie 2023, ora 11:00
Locul susținerii prelegerii/ probei orale (adresa Facultății/ Institutului și sala)	Institutul de cercetare în Inteligența Artificială, Realitate Virtuală și Robotică, Facultatea de Matematică și Informatică,

<p><b>Probele de concurs, data, ora și locul de susținere a acestora (adresa Facultății/ Institutului și sala)</b></p>	<p>str. T. Mihali, nr. 58-60, sala L343</p> <p>Pentru postul de asistent de cercetare, concursul constă în:</p> <ol style="list-style-type: none"> <li>1. evaluarea dosarului individual;</li> <li>2. susținerea unei probe orale;</li> <li>3. susținerea unei probe scrise.</li> </ol> <p>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 (<i>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</i>). 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 (limba română): 27 ianuarie 2023, ora 8:00, Institutul de cercetare în Inteligența Artificială, Realitate Virtuală și Robotică, Facultatea de Matematică și Informatică, str. T. Mihali, nr. 58-60, sala L343</p> <p>Proba 2 – Probă orală (limba română): susținerea unui proiect de seminar/ laborator/ lucrări practice – 27 ianuarie 2023, ora 11:00, Institutul de cercetare în Inteligența Artificială, Realitate Virtuală și Robotică, Facultatea de Matematică și Informatică, str. T. Mihali, nr. 58-60, sala L343. Dacă sunt mai mulți candidați, comisia va decide ordinea în care aceștia vor susține proba orală.</p> <p><b>Dosarul individual, proba orală și proba scrisă contează în proporții egale la nota finală</b> acordată în referatul individual de apreciere întocmit de fiecare membru al comisiei de concurs.</p>
<p><b>Tematica și bibliografia probelor de concurs</b></p>	<p><b>Proba 1 – Probă scrisă</b></p> <p>Tematică:</p> <p><b>A. Fundamentele programării</b></p> <p>Subalgoritmi: specificare, testare.</p> <p>Clase de algoritmi: căutare, sortare, interclasare.</p> <p>Metode de proiectare a algoritmilor: top-down, rafinare succesivă.</p> <p>Subprograme, apel și modalități de transmitere a parametrilor (prin valoare și referință).</p>

Tehnici de programare: Backtracking, Divide et impera, Greedy.

Programare modulară: modul, interfață, implementare; concretizare în C/C++, Java, Python.

### **B. Programare orientată pe obiecte**

Clase, obiecte.

Moștenire, polimorfism.

Programare bazată pe interfețe.

### **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.

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. 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 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. M.L. Hetland, Beginning Python: From Novice to Professional, Apress, 2005.
2. M. Frentiu, 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 și 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

### **Bibliografie:**

1. B. Stroustrup. The C++ Programming Language, Addison Wesley, 1998.
2. Bruce Eckel. Thinking in C++, Prentice Hall, 1995.
3. A. Alexandrescu. Programarea moderna in C++: Programare generica si modele de proiectare aplicate, Editura Teora, 2002.
4. S. Meyers. Effective C++: 55 Specific Ways to Improve Your Programs and Designs (3rd Edition), Addison-Wesley, 2005.
5. S. Meyers. More effective C++: 35 New Ways to Improve Your Programs and Designs, Addison-Wesley, 1995.
6. B. Stroustrup. A Tour of C++, Addison Wesley, 2013.
7. C++ reference (<http://en.cppreference.com/w/>).
8. Qt Documentation (<http://doc.qt.io/qt-5/>).
9. E. Gamma, R. Helm, R. Johnson, J. Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Longman Publishing, 1995.

### **D. Structuri de date si algoritmi**

1. Introducere. Structuri de date. Structuri statice, semistatice și dinamice.
2. Tipuri de date: domeniu, operatii și reprezentarea datelor
3. Tabloul
4. Abstract Data Types - TAD Colectie, TAD Dictionar, TAD

	<p>Stiva si Coadă, TAD Coadă cu prioritati</p> <ol style="list-style-type: none"> <li>5. Linked Lists - TAD Lista, Lista ȩlantuita</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. Aplicaȩii și structuri de date ȩn 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. Sedgewick: Algorithms, Addison-Wesley Publishing, 1984</li> </ol>
<p><b>Descrierea procedurii de concurs</b></p>	<p>Comisia de concurs evaluează candidaȩii ȩinȩnd cont de următoarele criterii:</p> <ul style="list-style-type: none"> <li>• Conȩinutul dosarului individual;</li> <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 obȩnute la criteriile de mai sus.</p> <p>Fiecare membru al comisiei (inclusiv președintele) ȩntocmește un referat individual de apreciere care propune o notă finală pentru fiecare candidat.</p> <p>Candidaȩii eligibili pentru ocuparea postului scos la concurs trebuie să obȩină:</p> <ul style="list-style-type: none"> <li>• cel puȩin nota 6 (șase) la fiecare criteriu;</li> <li>• nota finală cel puȩin 7 (șapte) dată de fiecare referent;</li> <li>• media generală cel puȩin 8,50 (opt și 50%).</li> </ul> <p>Președintele comisiei de concurs ȩntocmește un raport asupra concursului ȩn care prezintă notele finale atribuite candidaȩilor de către membrii comisiei și indică media generală obȩnută de fiecare candidat, calculată ca medie aritmetică a notelor finale din referatele individuale. Media generală astfel obȩnută reprezintă rezultatul concursului pentru fiecare candidat. Pe baza mediei generale, comisia de concurs decide ierarhia candidaȩilor și nominalizează candidatul eligibil care a ȩntrunit cel mai bun rezultat ȩn concurs. Președintele comisiei de concurs supune raportul asupra concursului votului secret al membrilor comisiei. ȩn urma exercitării votului secret, președintele constată rezultatul votului, ȩl comunică membrilor comisiei și ȩl menȩionează ȩn ȩncheierea raportului asupra concursului, cu precizarea numărului de voturi "pentru", respectiv "contra", votul fiind menȩinut secret. ȩn</p>

	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 institut**  
**Prof. dr. Horia F. Pop**

