

ANEXA Nr. 4

INFORMAȚII PUBLICE PRIVITOARE LA CONCURSURI

	Română	Engleză
Universitatea	Universitatea Babeș-Bolyai din Cluj-Napoca	Babes-Bolyai University, Cluj-Napoca
Facultatea / Structura organizațională de conducere	Facultatea de Matematică și Informatică	Faculty of Mathematics and Computer Science
Departamentul / Unitatea organizațională	Departamentul de Informatică	Department of Computer Science
Poziția în statul de funcții	61	61
Funcția	Lector universitar dr.	Lecturer
Disciplinele din încărcătura postului/ ariile de cercetare, așa cum figurează în statul de funcții	Algoritmica grafelor (în lb. engleză); Fundamentele programării (în lb. engleză); Programare orientată obiecte (în lb. engleză); Elaborarea lucrării de licență (în lb. engleză)	Graph algorithms; Fundamentals of Programming; Object Oriented Programming; Preparation of the Bachelor Thesis
Domeniul științific	Informatică	Computer Science
Descrierea postului scos la concurs	<p>Lector, 61, Departamentul Informatică. Postul de lector universitar presupune desfășurarea de activități didactice, de cercetare științifică și de îndrumare a studenților, precum și efectuarea de servicii pentru comunitatea academică.</p> <p>Cerințe pentru candidați: Candidații la ocuparea postului vacant de lector 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).</p>	<p>Lecturer, 61, Department of Computer Science. The position of lecturer involves teaching activities, scientific research and guidance of the students, as well as activities for the academic community.</p> <p>Requirements for candidates: The candidates must have the scientific record in conformity with the standards of computer science domain and with the required disciplines. Also, candidates must prove knowledge of the English language through documents (level C1 or documents attesting the study or research internships for at least 9 months abroad in institutions where communication was made in English).</p>
Atribuții	Activitatea didactică: curs, seminar, laborator, proiecte, consultații, lucrări de control, examene, elaborarea de	Teaching activity: course, seminar, laboratory, projects, tutoring, exams, preparation of teaching

	<p>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 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 fișa postului, cel puțin la nivelul calificativului -Satisfăcător-. <p>Activitatea de îndrumare a studenților: îndrumare de lucrări de diplomă și disertație, 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>	<p>materials.</p> <p>The scientific research activity:</p> <ul style="list-style-type: none"> • at least one participation to a research seminar within the faculty; • participation in competitions in order to obtain grants for scientific research; • the publication, every 3 years, of at least 3 papers/studies BDI indexed (Mathematical Reviews/ MathSciNet, ZMath (Emis), Computing Reviews, IEEE Xplore, DOAJ, SCOPUS, DBLP); • fulfillment of the requirements in the annex to the job description, at least at qualification level -Satisfactory-. <p>The activity of mentoring students: supervision of diploma and dissertation theses, tutoring activities, guidance for students participation in scientific seminars and student contests.</p> <p>Activities for the academic community: participation in the activities performed by the department, faculty and university (promotion of admission, the collaboration with the economic environment, etc).</p>
Perioada de înscriere la concurs	30.04.2024-13.06.2024	30.04.2024-13.06.2024
Data și ora susținerii probei orale	09.07.2024, 9:00	09.07.2024, 9:00 - (EEST GMT+3 - dd.mm.yyyy, hh:mm)
Locul susținerii probei orale (adresa Facultății/ Institutului și sala)	Departamentul de Informatică, str. Teodor Mihali nr. 58-60, sala L302	Department of Computer Science, Teodor Mihali Str, Room L302
Descrierea probei orale	Proba orală constă în prezentarea unui curs, în funcție de natura postului. Comisia de concurs stabilește titlul și îl anunță candidatului/candidaților cu 48 de ore înaintea prelegerii, fiind urmată obligatoriu de o sesiune de întrebări din partea comisiei și/sau a publicului.	The oral test consists of the presentation of a lecture, depending on the nature of the position. The competition committee determines the title and announces it to the candidate/candidates 48 hours before the lecture, followed by a question session from the committee and/or the audience.

	<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. Dacă sunt mai mulți candidați, comisia va decide ordinea în care aceștia vor susține proba orală.</p>	<p>The public lecture must have at least 30 minutes, being always followed by a session of questions of the committee members and/or the public. If there are more candidates, the committee will decide the order in which they will present the lecture.</p>
<p>Tematica și bibliografia probelor de concurs</p>	<p>Proba 1 –Probă orală: susținerea unui curs</p> <p>Tematică:</p> <p>A. Algoritmica grafelor (în lb. engleză)</p> <ol style="list-style-type: none"> 1. Noțiuni de bază (graf, multigraf, graf/multigraf orientat, drum, drum elementar, lanț simplu), reprezentări ale grafelor, grafe tare conexe, conexe (algoritmi 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). 	<p>Examination 1 – Public lecture – supporting a course</p> <p>Subjects:</p> <p>A. Graph algorithms</p> <ol style="list-style-type: none"> 1. Basic graph theory definitions (graph, multigraph, directed graph/multigraph, walk, trail, path), basics on graph representations, related hard graphs, related (algorithms for determining related components). 2. In-deep study of possible graph representations. Paths in graphs: length of a path (distance matrix, center, radius, diameter), value of a path, optimizations in the crowd of paths, Moore-Dijkstra algorithm. 3. Bellman-Kalaba algorithm, Ford's algorithm, matrix algorithms (Floyd-Hu, Dantzing, Floyd-Hu-Warshall), cyclic path, Eulerian paths, Hamiltonian paths. 4. Connectivity and shortest path problems. Depth-first and breadth-first traversal of a graph. 5. Fundamental numbers in graph theory: inner stability number, algorithm for determining inner stable manifolds, outer stability number, algorithm for determining outer stable manifolds, chromatic number, cyclomatic number. 6. Trees and forests: general notions, Kruskal's and Prim's algorithms. 7. Couplings in graphs: definitions, algorithm for determining maximum coupling, algorithm for determining maximum weight coupling. 8. Extreme problems (Ramsey and Turan theorems). 9. Hard problems: Hamiltonian cycle, the travelling

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 K5 și K3,3, relații între numărul de muchii și noduri.
12. Rețele de transport.
13. Fluxuri în rețele de transport: definiții de baza, 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. TOADERE, Grafe. Teorie, algoritmi și aplicații, Ed. Albastru, 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

- salesman problem. Counting and enumeration problems.
10. Hard problems: clique, vertex cover, coloring
11. Elementary Eulerian cycle. Planar graphs: Euler's relation, K5 and K3,3 graphs, relations between number of edges and nodes.
12. Transport networks.
13. Flows in transport networks: basic definiții, Ford-Fulkerson algorithm, Ford-Fulkerson algorithm extensions, minimum cost flows.
14. Matching problems.

Bibliography:

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. TOADERE, Grafe. Teorie, algoritmi și aplicații, Ed. Albastru, 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.

	<p>grafurilor. Ed. Did. si Pedag. Bucuresti 1981.</p> <p>B. Fundamentele programării (în lb. engleză)</p> <ol style="list-style-type: none"> 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 <p>Bibliografie:</p> <ol style="list-style-type: none"> 1. Kent Beck - Test Driven Development: By Example. Addison-Wesley Longman, 2002. 2. Kleinberg and Tardos – Algorithm Design. Pearson Educational, 2014 (http://www.cs.princeton.edu/~wayne/kleinberg-tardos/) 3. Martin Fowler - Refactoring. Improving the Design of Existing Code. Addison-Wesley, 1999. (http://refactoring.com/catalog/index.html) 4. Frentiu, M., H.F. Pop, Serban G. - Programming Fundamentals, Cluj University Press, 2006 5. The Python language reference. (https://docs.python.org/3/reference/index.html) 6. The Python standard library. (https://docs.python.org/3/library/index.html) 7. The Python tutorial. 	<p>B. Fundamentals of Programming</p> <ol style="list-style-type: none"> 1. Introduction to software development processes 2. Procedural programming 3. Modular programming 4. User defined types 5. Design guidelines 6. Object based programming 7. Program design 8. Program testing and inspection 9. Recursion 10. Algorithm complexity 11. Searching. Sorting. 12. Problem solving methods (I) – Backtracking, Greedy 13. Problem solving methods (II) – Divide & Conquer, Dynamic Programming methods <p>Bibliography:</p> <ol style="list-style-type: none"> 1. Kent Beck - Test Driven Development: By Example. Addison-Wesley Longman, 2002. 2. Kleinberg and Tardos – Algorithm Design. Pearson Educational, 2014 (http://www.cs.princeton.edu/~wayne/kleinberg-tardos/) 3. Martin Fowler - Refactoring. Improving the Design of Existing Code. Addison-Wesley, 1999. (http://refactoring.com/catalog/index.html) 4. Frentiu, M., H.F. Pop, Serban G. - Programming Fundamentals, Cluj University Press, 2006 5. The Python language reference. (https://docs.python.org/3/reference/index.html) 6. The Python standard library. (https://docs.python.org/3/library/index.html) 7. The Python tutorial. (https://docs.python.org/3/tutorial/index.html)
--	--	---

(<https://docs.python.org/3/tutorial/index.html>)

C. Programare orientată obiecte (în lb. engleză)

1. Elemente de bază ale limbajului C
2. Programare modulară în C/C++
3. Tipuri de date derivate și tipuri definite de utilizator, alocare dinamică în C++
4. Programare orientată obiect în C++
5. Elemente de programare generica
6. Moștenire
7. Polimorfism
8. Ierarhii de clase
9. Interfețe grafice utilizator
10. Elemente de programare bazată pe evenimente
11. Șabloane de proiectare
12. Aplicație POS (Point Of Sale)

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.

C. Object Oriented Programming

1. Basic elements in C
2. Modular programming in C/C++
3. Derived data types and user-defined types, dynamic allocation in C++
4. Object-oriented programming in C++
5. Generic programming elements
6. Inheritance
7. Polymorphism
8. Class hierarchies
9. Graphical User Interfaces
10. Event driven programming elements
11. Design patterns
12. Point Of Sale (POS) application

Bibliography:

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

	<p>VLISSIDES. Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Longman Publishing, 1995.</p>	<p>Object-Oriented Software, Addison-Wesley Longman Publishing, 1995.</p>
<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 (pondere 75%); • Susținerea unui curs (proba 1) (pondere 25%). <p>Nota finală a fiecărui candidat se calculează ca medie ponderată 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 6 (șase) la fiecare criteriu; • nota finală cel puțin 7 (șapte) dată de fiecare referent; • media generală cel puțin 8,50 (opt și 50%). <p>Președintele comisiei de concurs întocmește un raport de sinteză 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 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,</p>	<p>The committee evaluates the candidates taking into account the following criteria:</p> <ul style="list-style-type: none"> • the content of the candidate file (75%); • public lecture (examination 1) (25%). <p>The final grade of every candidate will be computed as the weighted average of grades obtained for the above criteria. Each member of the committee (including the president) writes an individual report and proposes a final grade for each candidate.</p> <p>Eligible candidates must obtain:</p> <ul style="list-style-type: none"> • at least grade 6 (six) for each criterion; • the final grade at least 7 (seven) from each committee member; • the overall average at least 8.50 (eight and 50%). <p>The committee president will write a synthesis report containing the final grades assigned to the candidates by the committee members and indicates the overall average obtained by each candidate, calculated as the arithmetic mean of the individual final grades. The overall average thus obtained represents the result of the competition for each candidate. Based on the general average, the committee decides the hierarchy of the candidates and nominates the eligible candidate who has qualified the best in the competition. The committee members will vote the final report of the committee president; the voting will be open. As a result of the voting, the president acknowledges the result of the vote, communicates it to the committee members and mentions it in the conclusion of the competition report, together with an indication of the number of votes "for"</p>

	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. 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.	and "against". In the case when the vote "for" is not granted by the majority of the committee members, the position is not taken by any candidate. The hierarchy of the candidates as it results from the overall average grades obtained by each candidate can not be altered by the committee's vote. The competition report is signed by each committee member and by the committee president.
Perioada de comunicare a rezultatelor	8.07.2024-9.07.2024	8.07.2024-9.07.2024
Perioada de depunere a contestațiilor	10.07.2024-12.07.2024	10.07.2024-12.07.2024
Salariul minim de încadrare a postului la momentul angajării	7247	7247
Lista completă a documentelor pe care candidații trebuie să le includă în dosarul de concurs	https://www.ubbcluj.ro/ro/infoubb/posturi_vacante/posturi_didactice_perioada_nedeterminata	https://www.ubbcluj.ro/ro/infoubb/posturi_vacante/posturi_didactice_perioada_nedeterminata
Adresa la care trebuie trimis dosarul de concurs	Registratura Universității “Babeș-Bolyai”, (camera P20), str. M. Kogălniceanu nr. 1, Cluj-Napoca	Main building of Babes-Bolyai University, (Room P20) M. Kogălniceanu Street, 1, Cluj-Napoca