

UNIVERSITATEA TEHNICA "GHEORGHE ASACHI" DIN IAȘI

Facultatea de Automatica și Calculatoare

Departamentul de Calculatoare

Concurs pentru ocuparea postului de asistent universitar, poz. 58

Disciplinele postului: Programare orientată pe obiecte,
Programare II,
Sisteme distribuite

Tematica de concurs

**pentru ocuparea postului de asistent poz. 58, din Statul de
funcții al Departamentului de Calculatoare
pe anul universitar 2018-2019**

Pentru proba scrisă, proba orală și proba practică

A. Programare orientată pe obiecte și Programare II - pentru proba orală și proba practică

1. Introducere C++:
 - a. tipuri de data în C++
 - b. operații intrare ieșire (cin, cout) C++
 - c. funcții membre ale structurilor
 - d. operatorul de rezoluție
2. Tipul referință în C++
3. Alocarea și dealocarea dinamică de memorie (pentru variabile simple și pentru tablouri)
4. Funcții cu parametri implicați
5. Supraîncărcarea funcțiilor
6. Constante
 - a. Variabile constante
 - b. Funcții constante
 - c. Pointeri constanți
 - d. Funcții membre constante
7. Namespace-uri
8. Principiile de baza ale POO:
 - a. Abstractizarea datelor
 - b. Incapsularea
 - c. Polimorfismul
 - d. Moștenirea
9. Clase
 - a. specificatorii de acces (teorie, exemple, specificatori implicați)
 - b. Constructori (constructorul de copiere (apelare implicită), liste de inițializare, constructori expliți, constructor implicit, constructor fără lista de argumente, constructor de inițializare, constructor de copiere)
 - c. Destructor
 - d. Pointerul this
10. Supraîncărcarea operatorilor
 - a. prin funcții membre

- b. prin funcții globale
 - c. exemplu operatorul +
 - d. operatorii implicați = și &
 - e. Operatorul ++ forma post și prefixată
- 11. Membri statici
 - a. Variabile membre statice
 - b. Funcții membre statice
- 12. Funcții inline
- 13. Compoziția
- 14. Agregarea
- 15. Template-uri
 - a. Conversii int-float
 - b. Templateuri de funcții
- 16. Standard C++ library
 - a. STD vs STL
 - b. STD (Fișiere header STD, Iteratori, <vector>, <string>, std::sort – descriere, exemplu de utilizare)
- 17. Funcții prietene
- 18. Clase prietene
- 19. Moștenirea
 - a. Ierarhii de tipuri
 - b. Specificatori de acces
 - c. Suprascrierea metodelor din clasa de bază
 - d. Ordinea de apelare a constructorilor și a destructorilor
 - e. Constructori apelați implicit
 - f. Rescrierea funcțiilor din clasa de bază
 - g. Moștenirea multiplă
- 20. Polimorfism
 - a. Pointeri la clasa de bază
 - b. Operatorul de referențiere &
 - c. Tip declarat / tip real
 - d. Apelarea unei funcții suprascrise
 - e. Funcții virtuale
 - i. Tabela virtuală
 - f. Apelare constructori și destructori in cazul polimorfismului
 - i. Constructori virtuali
 - ii. Destructorii virtuali
 - g. Funcții virtuale pure
 - h. Type casting
- 21. Design patterns
 - a. Factory Method
 - b. Adapter
 - c. Chain of Responsibility
- 22. Excepții in C++

Bibliografie:

1. Bjarne Stroustrup - The C++ Programming Language, 4th edition, 2013
2. Bjarne Stroustrup - Programming: Principles and Practice Using C++ (2nd Edition), 2014

B. Sisteme distribuite - pentru proba scrisă și proba orală

1. Sisteme distribuite de fișiere (SDF)
 - a. Exemple de SDF și analiza lor
 - b. Atomicitate și ordinea evenimentelor
 - c. Consistența și tratarea la nivel hardware și software
 - d. Tehnici și algoritmi pentru replicarea fișierelor
 - e. Sistemul NFS, SNF, Decorum, AFS
2. Echilibrarea Încărcării
 - a. Problematica
 - b. Echilibrarea statică a încărcării
 - c. Metode de planificare dinamică
 - d. Echilibrarea prin predicția încărcărilor
3. Memoria virtuală distribuită DSM
 - a. Memoria distribuită simulată
 - b. Memorie comuna distribuită realizată software
 - c. Modele de consistență
4. Cluster
 - a. Justificare apariției conceptului de cluster și clasificări
 - b. Arhitectura unui cluster
 - c. Tehnici și suportul pentru disponibilitate
 - d. Exemple de Clustere
5. GRID
 - a. Calcul de mare încărcare
 - b. Descrierea Arhitecturii Grid
 - c. Clasificări ale Tehnologiilor Grid
 - d. Problema programării și tehnologii utilizate
6. SOA
 - a. Fundamentele SOA
 - b. Implementări SOA
 - c. Tipuri de Nori
7. Peer to Peer P2P
 - a. Autoorganizarea distribuită
 - b. Sisteme P2P nestructurate și structurate, exemple
 - c. Proiectarea unei aplicații P2P
 - d. Problema clasificării documentelor
8. Sisteme distribuite tolerante la erori
 - a. Servere, servicii și relația lor de dependență
 - b. Abordări hardware
 - c. Sincronizarea membrilor unui grup tolerant la erori
 - d. Puncte de salvare a execuției și refacerea
9. Calcul distribuit mobil
 - a. Arhitecturi specifice
 - b. Rețele sensor
 - c. Modele de mobilitate și protocoale de rutare
10. Inteligența Artificială Distribuită – DAI
 - a. Agenți mobili
 - b. Limbaje specific pentru comunicare
 - c. Limbaje pentru conținut

Bibliografie:

1. Muhl, G., Fiege L, Pietzuch P., „Distributed event based systems”, Springer-Verlag Berlin Heidelberg, Germania, 2006
2. Raynal, M. , “Distributed Algorithms and Protocols”, John Wiley & Sons Ltd, 1988.
3. Raynal, M., Helary, J.M., “Synchronisation and Control of Distributed Systems and Programs ”, John Wiley & Sons Ltd, 1988.
4. Tanembaum A. S., "Distributed Systems", Prentice Hall 1993.
5. Mihai Horia Zaharia, Cătălin Horghidan, Claudiu Mihăilă, Sisteme Distribuite: îndrumar de laborator, Politehnia, 2007, ISBN 978-973-621-175-1, 225 format academic
6. Dmitry Korzun, Andrei Gurtov, Structured Peer-to-Peer Systems - Fundamentals of Hierarchical Organization, Routing, Scaling, and Security, Springer Heidelberg, 2013
7. A.D. Kshemkalyani, M. Singhal, Distributed Computing: Principles, Algorithms, and Systems, ISBN: Cambridge University Press, March 2011
8. Coulouris G, Dollimore J, Kindberg T. Distributed Systems: Concepts and Design 5th Edition. New York, PA: Pearson 2011.
9. Schumacher M, Fernandez-Buglioni E, Hybertson D, et al, Security Patterns - Integrating Security and Systems Engineering. Chichester: John Wiley & Sons Ltd. 2006:59-83.
10. Betz, C. T. (2007). Architecture and Patterns for IT Service Management, Resource Planning, and Governance: Making Shoes for the Cobbler's Children: Making Shoes for the Cobbler's Children. San Francisco: Morgan Kaufmann Publishers Inc.

Decan,
Prof. Vasile-Ion Manta

Director de departament,
Prof. Petru Cașcaval