

Întrebările pentru examenul de licen 2009
„Specializare: Tehnici de programare. Limbajul HTML. Rețele Petri. Limbajul C++.
Limbaajul de programare logic Prolog. Limbajul de asamblare.
Inteligenta artificială . Grafuri”

1. Noțiuni generale despre recursivitate. Varianta recursiv și nerecursiv :
 - irului Fibonacci;
 - calcularea factorialului;
 - calcularea sumei primelor n termeni ai irului: $1, 2, 5, 7, \dots, 2n - 1$;
 - calcularea sumei primelor n termeni ai irului: $1, 5, 9, 13, \dots, 4n - 3$;
 - calcularea sumei primelor n termeni ai irului: $2, 7, 12, 17, \dots, 5n - 3$.
2. Aplicații ale recursivității. Varianta recursiv și nerecursiv :
 - Algoritmul Euclid;
 - Suma cifrelor unui număr natural;
 - Calcularea numărului de combinații;
3. Proceduri recursive:
 - Returnarea irului de numere $1, 2, 3, \dots, k$;
 - Inversarea cuvântului;
 - Transformarea unui număr natural din baza 10 în baza k ($k > 1, k < 10$).
4. Recursivitatea indirectă. Noțiuni preliminare. Să se scrie un program pentru calcularea valorilor următoarelor funcții:

$$4.1. f(x) = \begin{cases} 1 + g(x), & x \leq 3, \\ x^2, & x > 3. \end{cases}; \quad g(x) = \begin{cases} 5, & x < 0, \\ 2x + f(x+1), & x \geq 0. \end{cases};$$

$$4.2. f(x) = \begin{cases} g(x) - 2, & x > 1, \\ x + 2, & x \leq 1. \end{cases}; \quad g(x) = \begin{cases} f(x) + 1, & x \geq 0, \\ -x, & x < 0. \end{cases}.$$

5. Aplicarea recursivității la calcularea polinoamelor. Să se calculeze valorile polinomului:

$$5.1. \text{ Formula Ackerman: } \quad ack(m, n) = \begin{cases} n + 1, & m = 0, \\ ack(m - 1, 1), & n = 0, \\ ack(m - 1, ack(m, n - 1)). \end{cases}$$

$$5.2. \text{ Formula Cebîșev: } \quad T(x) = \begin{cases} 1, & n = 0, \\ 0, & n = 1, \\ 2 \cdot x \cdot T_{n-1}(x) - T_{n-2}(x), & n \geq 2. \end{cases}$$

$$5.3. \text{ Formula Hermite: } \quad H(x) = \begin{cases} 1, & n = 0, \\ 2x, & n = 1, \\ 2xH_{n-1}(x) - 2(n-1)H_{n-2}(x), & n > 1. \end{cases}$$

5.4. Funcia Manna-Pnueli:

$$F(x) = \begin{cases} x-1, & x \geq 12 \\ F(F(x+2)), & x < 12 \end{cases}$$

6. Metoda Backtraking. Necesitatea metodei
 - Generarea permut rilor
 - Problema secven ei de litere
 - Generarea combin rilor de n elemente luate cte k

Implementarea metodei Backtraking la solu ionarea problemelor. Varianta recursiv
7. Metoda Backtraking. Rolul stivei
 - Generarea arajamentelor de n elemente luate cte k
 - Generarea produsului cartezian din n mul imi
 - Generarea submul imilor

Implementarea metodei Backtraking la solu ionarea problemelor. Varianta recursiv
8. Backtraking nerecursiv
 - Generarea permut rilor
 - Problema secven ei de litere
 - Generarea combin rilor de n elemente luate cte k

Implementarea metodei Backtraking la solu ionarea problemei. Varianta nerecursiv
9. Backtraking nerecursiv
 - Generarea arajamentelor de n elemente luate cte k
 - Generarea produsului cartezian din n mul imi
 - Generarea submul imilor

Implementarea metodei Backtraking la solu ionarea problemelor. Varianta nerecursiv
10. Aplicarea metoda Backtraking la solu ionarea problemei:
 - comisvoiajorului
 - cump r turilor
 - mesei rotunde
 - generarea numerelor prime de n cifre, formate din 2, 0 i 9.
11. Metoda Divide et Impera.
 - Determinarea elementului maximal dintr-un ir de numere ntregi.
 - C utarea binar într-un ir.
12. Metoda Divide et Impera
 - Turnurile din Hanoi
 - Determinarea c.m.m.d.c
 - Foaia de tabl
13. Metode de sortare
 - Sortarea prin interschimbare direct
 - Metoda bulelor
 - Metoda selec iei
 - Quick Sort

14. Tehnica Greedy

- Determinarea celor mai mari 2 numere întregi introduse de la tastatur
- Determinarea celor mai mari 3 numere întregi introduse de la tastatur
- Determinarea celor mai mari 2 numere impare introduse de la tastatur

15. Tehnica Greedy

- Determinarea celor mai mari 3 numere pare introduse de la tastatur
- Determinarea celui mai mare element par a_2 și celui mai mare element impar a_1 ale unui vector de componente întregi, cu condiția $a_1 < a_2$

16. Programarea dinamică. Necesitatea metodei. Determinarea celui mai mare subșir crescător

17. Tehnici de lucru privind alcătuirea programelor care operează cu mulțimi. Exemple

18. Backtracking generalizat. Exemple

19. HTML. Cerințe generale privind crearea paginilor web. Etapele de elaborare a unei pagini web. Structura unui document html. Etichete de bază

20. HTML. Liste (numerotate și nenumerotate). Formatarea textului. Stiluri logice, fizice, fontul, &-consecutivități

21. HTML. Crearea și redactarea tabelelor

22. HTML. Legături (în cadrul aceluiași document și cu alte documente)

23. HTML. Inserarea imaginilor. Hărți-imagini

24. HTML. Cadre (interne, externe). Formulare

25. HTML. Stiluri. Clase. Pseudoclasă. Modalități de utilizare. Foi de stiluri

26. Rețele Petri locale tranziție. Regula de tranziție. Secvențe de tranziții. Situații fundamentale (secvența conflictului concurenței)

27. Rețele Petri. Tehnica invarianților și verificarea proprietăților. Invarianți locale și invarianți tranziție

28. Structuri de acoperire pentru rețele Petri. Structuri de accesibilitate. Probleme de decizie

29. Limbajul de asamblare. Reprezentarea numerelor în formatul mării

30. Limbajul de asamblare. Instrucțiuni aritmetice

31. Limbajul de asamblare. Deplasări și rotații

32. Limbajul de asamblare. Instrucțiuni de lucru cu șiruri

33. Limbajul de programare logică Prolog. Nucleul limbajului

34. Limbajul de programare logică Prolog. Controlarea backtrackingului. Utilizarea predicatului ! (cut)

35. Limbajul de programare logic Prolog. Liste și operații cu ele
36. Limbajul de programare logic Prolog. Baze de date dinamice
37. Limbajul de programare logic Prolog. Lucrul cu irurile
38. Predicate interne utilizate în Turbo-Prolog: write, read, makewindow etc.
39. Inteligența artificială . Sisteme de inteligență artificială în comparație cu cele tradiționale
40. Inteligența artificială . Testul Turing
41. Inteligența artificială . Sistemul “Eliza”
42. Inteligența artificială . “Matricea elementelor de cunoștințe” – produs efectiv pentru extragerea și prezentarea cunoștințelor
43. Limbajul C++. Funcțiile *cin* și *cout*. Formatele de citire-scriere. Bibliotecile *iostream*, *omanip*
44. Limbajul C++. Structura claselor. Modificatori de acces. Incapsularea datelor. Alocarea dinamică a memoriei. Operatorii *new* și *delete*. Constructori și destructori
45. Limbajul C++. Fișiere. Deschiderea/închiderea unui fișier. Funcții de citire/scriere. Clasa *fstream*. Funcții prietene.
46. Limbajul C++. Supraîncărcarea operatorilor (prin intermediul funcțiilor membre, prin intermediul funcțiilor prietene).
47. Limbajul C++. Relația de moștenire. Clase de bază . Clase derivate. Controlul accesului la membrii clasei de bază
48. Limbajul C++. Constructori și destructori în clasele derivate. Funcții virtuale. Polimorfismul
49. Limbajul C++. Pointeri către obiecte. Apelul metodelor virtuale. Deosebirea dintre metodele statice și cele virtuale
50. Grafuri neorientate. Reprezentarea și parcurgerea (algoritmii Breadth First și Depth First). Grafuri parțiale, subgrafuri. Grafuri conexe
51. Grafuri hamiltoniene. Grafuri euleriene
52. Arbori. Arbore parțial de cost minim. Algoritmi de construire
53. Arbori binari

**eful catedrei ITI
doctor conferențiar
Andrei Braicov**