

Seminar 2

Problemă rezolvată de un algoritm. Complexitatea algoritmilor.

Ștefan Ciobâcă, Dorel Lucanu
Universitatea “Alexandru Ioan Cuza”, Iași

Săptămâna 22 Februarie - 26 Februarie 2016

1. Scrieți trei algoritmi (Euclid prin scăderi repetate, Euclid prin împărțiri repetate, căutare explicită) pentru următoarea problemă:
Input: două numere naturale $a, b \in \mathbb{N}$ cu proprietatea că $a + b \neq 0$;
Output: cel mai mare divizor comun al numerelor a, b .
2. Dați exemple de câteva instanțe ale problemei de mai sus și de răspunsurile aferente.
3. Problema de mai sus este aceeași cu problema ce urmează?
Input: două numere naturale $a, b \in \mathbb{N}$;
Output: “da” și $\text{cmmdc}(a, b)$, dacă a și b au un cel mai mare divizor comun și “nu” altfel.
4. Calculați timpul de execuție în cazul cel mai nefavorabil în modelul uniform și în modelul logaritmnic pentru algoritmi de mai sus.
5. Enunțați problema celor 8 regine. Dați exemplu de instanța a problemei.
6. Enunțați problema sortării în ordine crescătoare a unui șir de numere naturale.
7. Scrieți în Alk algoritmi bubble-sort, insertion-sort și selection sort pentru problema de mai sus.
8. Arătați că algoritmi de mai sus rezolvă într-adevăr problema.
9. Calculați timpul de execuție în cazul cel mai nefavorabil în cele două modele de calcul pentru algoritmi de sortare de mai sus.
10. Enunțați problema satisfiabilității pentru formule din LP ca pereche **Input-Output**.

11. Enunțați problema satisfiabilității pentru formule din LP1 ca pereche **Input-Output**.
12. Enunțați problema opririi.
13. Gândiți-vă la trei algoritmi pe care i-ați învățat la liceu și enunțați ce probleme rezolva algoritmii.
14. Care dintre problemele discutate la acest seminar sunt probleme de decizie? Care sunt probleme rezolvabile?
15. Enunțați problemele adunării/înmulțirii a două numere naturale, numere reprezentate printr-un șir de cifre.
16. Scrieți algoritmi pentru problemele de mai sus.
17. Calculați timpul de calcul în cazul ce mai nefavorabil pentru cei doi algoritmi de mai sus (în funcție de dimensiunea datelor de intrare).
18. Problemă mai grea: scrieți algoritmul lui Karatsuba pentru înmulțirea a două numere naturale (de complexitate $O(n^{\log_2(3)})$).
19. Problemă și mai grea: scrieți un algoritm în timp $O(n \log(n))$ pentru înmulțirea a două numere naturale folosind transformarea Fourier rapidă.
20. Găsiți soluții cât mai eficiente din punct de vedere al timpului de calcul și al necesarului de memorie pentru următoare problemă: dându-se un număr natural n și $n - 1$ numere distincte între 1 și n să se determine numărul lipsă. Nu uitați să formalizați problema ca pereche **Input-Output**. Arătați că algoritmul (sau algoritmii) propuși rezolvă într-adevăr problema; calculați timpul de calcul și necesarul de memorie în cele două modele.
21. Același exercițiu ca mai sus, dar de data aceasta un număr apare de două ori și un număr lipsește (înainte să scrieți algoritmul formalizați problema).
22. Formalizați următoarea problemă și scrieți un algoritm pentru ea: să se scrie un natural n ca sumă de numere naturale consecutive în toate variantele posibile.
23. De ce nu “citim” datele de intrare de la tastatură (sau dintr-un fișier) și de ce nu “scriem” rezultatul pe ecran (sau într-un fișier) la acest curs?
24. Temă pentru punctul bonus: implementați în Alk algoritmul bubble-sort și rulați algoritmul pe un șir de 5 până la 10 numere la alegere. Predați un screenshot cu rezultatul execuției la următorul seminar.