

FIȘA 21. Metoda de programare *Greedy*

Pentru fiecare dintre următoarele enunțuri scrieți programul C++ corespunzător.

1. Se dă o mulțime $A = \{a_1, a_2, \dots, a_n\}$ cu elemente reale. Să se determine o submulțime a lui s astfel încât suma elementelor submulțimii să fie maximă.
2. Se dă o mulțime $A = \{a_1, a_2, \dots, a_n\}$ cu elemente întregi. Să se determine cele mai mari două elemente ale mulțimii.
3. La o festivitate de premiere, dirigintele clasei are n obiecte ($n \leq 1000$), de valori cunoscute, mai mici decât 100 lei. Știind că elevului care a obținut premiul I, îi va fi înmănat m obiecte, realizați un program care identifică valoarea maximă a premiului I și care obiecte au fost selectate.
4. La o casă de marcat sunt serviți n clienți ($n \leq 1000$). Cunoscându-se timpul necesar de servire pentru fiecare client, să se determine o ordine de servire a clienților, astfel încât timpul mediu de așteptare să fie minim.
5. Într-o sală de spectacole, într-o zi, trebuie planificate n spectacole. Pentru fiecare spectacol se cunoaște intervalul în care se desfășoară. Se cere să se planifice un număr maxim de spectacole astfel încât să nu se suprapună.
6. Pentru două mulțimi de numere întregi nenule, C cu n elemente și A cu m elemente, ($n \leq m$), se cere să se formeze o submulțime de n elemente din mulțimea A , astfel încât expresia: $E = c_1 \cdot x_1 + c_2 \cdot x_2 + \dots + c_n \cdot x_n$ să aibă valoarea maximă, ($x_i \in A$).
7. Să se plătească o sumă s cu un număr minim de bancnote cu valori date. Se consideră că din fiecare tip de bancnotă se poate folosi un număr nelimitat de bancnote, iar pentru ca problema să aibă soluție, vom considera că există și bancnote cu valoarea 1.
8. Fiind dat un număr natural n , afișați pe ecran primele n numere frumoase. Numerele frumoase sunt numerele care au ca factori primi doar pe 2, 3 și 5.
9. Se consideră un rucsac cu care se poate transporta o greutate maximă G_{max} și mai multe obiecte de greutate g_1, g_2, \dots, g_n , la transportul cărora se obțin câștigurile c_1, c_2, \dots, c_n . Se cere să se încarce rucsacul astfel încât să se obțină un câștig maxim. Problema discretă a rucsacului (obiectele nu pot fi tăiate).
10. Se consideră un rucsac cu care se poate transporta o greutate maximă G_{max} și mai multe obiecte de greutate g_1, g_2, \dots, g_n , la transportul cărora se obțin câștigurile c_1, c_2, \dots, c_n . Se cere să se încarce rucsacul astfel încât să se obțină un câștig maxim. Problema continuă a rucsacului (obiectele pot fi tăiate).