Data: 11.10.2022

Clasa: a 11 – a MI

Lecţia: Metoda Backtracking. Aplicații. Generare de submulțimi, partiții

Tipul lecţiei: Lecție mixtă

|  |  |
| --- | --- |
| **Activitatea din lecţie** | **Observaţii** |
| **Titlul lecției:** Metoda Backtracking. Aplicații. Generare de submulțimi, partiții  **Probleme rezolvate:**  **Cerință:**  Se citește un număr natural nenul n. Să se afişeze, în ordine lexicografică, toate submulțimile nevide ale mulţimii {1,2,..,n}.  **Date de intrare**:  Fişierul de intrare submultimi.in conţine pe prima linie numărul n. Date de ieşire Fişierul de ieşire submultimi.out va conţine pe fiecare linie câte o submulțime, elementele unei submulțimi fiind separate printr-un spațiu. Restricţii şi precizări 1 ≤ n ≤ 10  **Rezolvare:**  #include <fstream>  using namespace std;  ifstream cin("submultimi.in");  ofstream cout("submultimi.out");  int x[11], n;  void afisare (int k)  {  for (int i=1;i<=k;i++)  cout<<x[i]<<" ";  cout<<endl;  }  void back(int k)  {  for(int i=x[k-1]+1;i<=n;i++)  {  x[k]=i;  afisare(k);  back(k+1);  }  }  int main()  {  cin>>n;  back(1);  return 0;  } | **Obiective operaționale:**   * **O1:** Să știe să aplice metoda backtracking * **O2:** Să afișeze în ordine lexicografică toate submulțimile nevide ale mulțimii {1,2,…,n} * **O3:** Să identifice probleme care pot fi rezolvate cu metoda backtracking * **O4:** Să scrie un pogram corect, fără erori de compilare * **O5:** Să scrie un program corect, fără erori de logică |