Data: 12.10.2022

Clasa: a 11 – a MI

Lecţia: Metoda Backtracking. Aplicații. Generare de partiții

Tipul lecţiei: Lecție mixtă

|  |  |
| --- | --- |
| **Activitatea din lecţie** | **Observaţii** |
| **Titlul lecției:** Metoda Backtracking. Aplicații. Generare de partiții  **Probleme rezolvate:**   * S-a folosit subiectul pentru competențe digitale din data de (06.07.2022) și s-a rezolvat problema 3 de la Sub III.   **Cerință:**  Se dă o mulțime alcătuită din n elemente (1, 2, ..., n). Să se construiască și să se afișeze partițiile mulțimii.  **Rezolvare:**  #include <iostream>  using namespace std;  int x[20], n, nrsol = 0, max[20], maxim;  int maxx(int k)  {  int maxim = 0, i;  for(i = 1; i <= k; i++)  if(x[i] > maxim)  maxim = x[i];  return maxim;  }  void tipar()  {  int i, j;  maxim = maxx(n);  for(j = 1; j <= maxim; j++)  {  cout << “{ “;  for(i = 1; i <= n; i++)  {  if(x[i] == j)  cout << i << “ “;  }  cout << “} “;  }  cout << endl;  nrsol++;  }  void backtr(int k)  {  int i;  for(i = maxx(k-1) + 1; i >= 1; i--)  {  x[k] = i;  if(k == n)  tipar();  else  backtr(k+1);  }  }  int main()  {  cout << “Partitiile multimii {1, 2, …, n}” << endl;  cout << “Dati valoare lui n: “ < endl;  cin >> n;  backtr(1);  cout << Numar soluții “ < nrsol;  return 0;  } | | **Obiective operaționale:**   * **O1:** Să știe să aplice metoda backtracking * **O2:** Să se construiască și să se afișeze partițiile unei mulțimi. * **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 |