

## Tutoriat 8 – Divide et Impera. Backtracking - probleme -

1. Se citește o listă **l1** de numere sortată crescător. Citindu-se o listă nouă **l2**, afișați pentru fiecare număr din **l2** dacă se află în **l1**.
2. Folosind metoda Divide et Impera, aflați elementul cu valoare maximă dintr-o listă citită de la tastatură.
3. Scrieți subprogramul **mergeSort**, care, primind ca parametru o listă și o funcție cheie, sortează lista prin interclasare după cheia dată.
4. Se citește o listă de numere naturale, reprezentând valorile unor bancnote. Citindu-se o sumă **s**, generați toate modurile în care se poate achita suma dată, cu bancnotele puse la dispoziție. Se consideră că există un număr infinit de bancnote de fiecare tip.

### *Exemplu:*

Intrare	Iesire
1 2 5 s = 5	1 1 1 1 1 1 1 1 2 1 2 2 5

5. Generați toate aranjamentele de **n** luate câte **k** ale unei mulțimi de **n** elemente, citită de la tastatură. Atât elementele listei, cât și **n** și **k** vor fi numere naturale,  $0 \leq k \leq n$ .

### *Exemplu:*

Intrare	Iesire
7 1 25 k = 2	7 1 7 25 1 7 1 25 25 7 25 1