

Tema Nr. 7: Arbori Multicai

Transformari intre diferite reprezentari

Timp alocat: 2 ore

Implementare

Se cere implementarea **corecta** si **eficienta** a unor algoritmi de complexitate *liniara* pentru transformarea arborilor multicai intre urmatoarele reprezentari:

R1: *reprezentarea parinte:* pentru fiecare index, valoare din vector reprezinta indexul parintele, e.g: $\Pi = \{2, 7, 5, 2, 7, 7, -1, 5, 2\}$

R2: *reprezentare arbore multicai:* fiecare nod contine cheia si un vector de noduri copil

R3: *reprezentare binara:* fiecare nod contine cheia si doi pointeri: unul catre primul copil si al doilea catre fratele din dreapta (e.g. urmatorul frate).

Pentru *reprezentarea binara (R3)* trebuie sa implementati afisarea prietenoasa (**PP**).

Asadar, trebuie sa definiti transformarea **T1** din reprezentarea *parinte (R1)* in reprezentarea *arbore multicai (R2)* iar apoi transformarea **T2** in reprezentarea *binara (R3)*. Folositi afisarea prietenoasa pentru cele trei reprezentari (consultati pag 2 pentru un exemplu).

Definiti structurile de date. Puteti folosi structuri intermediare (e.g. memorie aditionala).

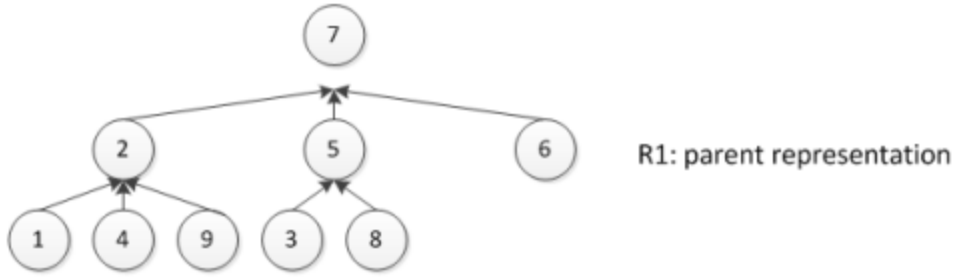
Evaluare

Corectitudinea algoritmilor va trebui demonstrata pe exemplul de la **R1** (Π). Folositi afisarea prietenoasa pentru cele trei reprezentari.

Explicati ce structuri de date ati folosit pentru reprezentarile **R2** si **R3**.

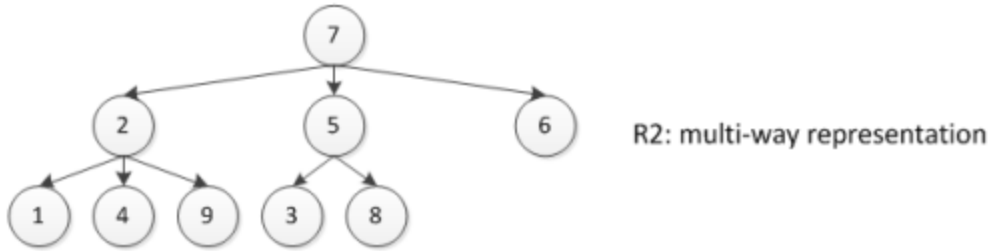
Analizati eficienta in timp si spatiu a celor doua transformari. Ati atins $O(n)$? Ati folosit memorie aditionala?

Input (R1): $\Pi = \{2, 7, 5, 2, 7, 7, -1, 5, 2\}$
 1 2 3 4 5 6 7 8 9



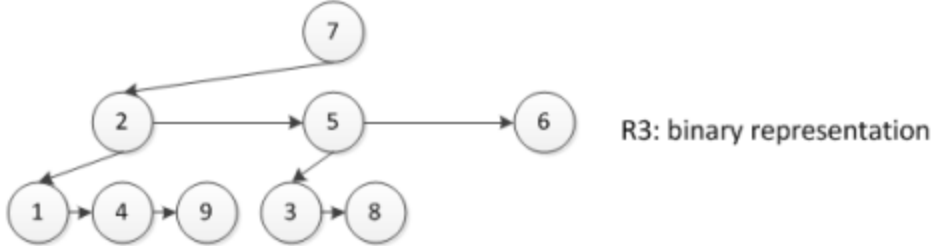
R1: parent representation

T1: parent -> multi-way



R2: multi-way representation

T2: multi-way -> binary



R3: binary representation

PP: pretty_print (binary)

