

Cvičenie 5 – K-cestný strom

Cieľ:

Cieľom cvičenia je implementovať k-cestný strom.

Postup:

1. Stiahnite si zo vzdelávania zdrojové súbory potrebné pre toto cvičenie.
 - Do balíčka gui v projekte DataStructures prekopírujte súbory Rutines.java a PanelTree.java (prepíšte súbory Routines.java a PanelFront.java).
 - Vytvorte balíček datastructures.Trees, do ktorého vložte súbory ITree.java, ITreeNode.java a ETree.java.
2. Prejdite si rozhranie ITree, ITreeNode a výnimku EPriorityFront.
3. Implementujte triedu KWayTreeNode (do balíčka datastructures.trees.KWayTree) implementujúcu rozhranie ITreeNode. Táto trieda bude predstavovať vrchol stromu.
4. Implementujte triedu KWayTree (do balíčka datastructures.trees.KWayTree) implementujúcu rozhranie ITree. Úlohou tejto triedy je správna manipulácia so stromom. Metódy, ktoré reprezentujú dotazy a prehliadky zatiaľ neimplementujte (urobte v nich len bezpečný return, ktorý umožní preloženie programu).
5. Do triedy FrameMain pridajte konečné atribúty typu Vášho stromu. V konštruktore ho inicializujte a následne ním inicializujte ovládací panel štruktúry. Panel si je potrebné najskôr vyrobiť (rovnakým postupom, ako pre ovládanie zásobníkov, frontov a prioritných frontov). Štruktúra sa ešte nebude správne zobrazovať v rozhraní – na to je potrebné dokončiť ďalší bod.
6. Implementujte metódu levelOrder. Na jej implementáciu využite Vami naprogramovaný zásobník.
7. Implementujte metódy preOrder a postOrder.

Čo môžete spraviť doma:

Každý rekurzívny algoritmus je možné zapísať aj nerekurzívne. Metódy preOrder a postOrder sú veľmi jednoduché rekurzívne metódy, kde si je možné jednoducho vyskúšať takýto prepis. Prepíšte metódy preOrder a postOrder tak, aby neboli rekurzívne. Pri prepise môžete využívať len Vami naprogramované štruktúry (ako napr. zásobník).

Trieda KWayTree má z rozhrania implementovať aj dotazy isComplete a isBalanced. Navrhnite a implementujte pre ne efektívne algoritmy.