

ALGORYTMY I STRUKTURY DANYCH

SŁOWNIKI ZEWNĘTRZNE I DRZEWIA POZYCYJNE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. **[**]** Wykaż, że każde drzewo BST o n węzłach można przekształcić w dowolne inne drzewo BST o n węzłach za pomocą $O(n)$ pojedynczych rotacji.
Wskazówka. Pokaż, jak za pomocą $O(n)$ pojedynczych rotacji przekształcić dowolne drzewo w listę wykorzystującą tylko prawe (albo tylko lewe) wskaźniki w węzłach.
2. **[*]** Wykaż, że najdłuższa prosta ścieżka z węzła x do czarnego liścia w drzewie RBT jest co najwyżej dwa razy dłuższa niż najkrótsza ścieżka z x do liścia.
3. **[**]** Jaka jest najmniejsza możliwa ilość czerwonych węzłów w drzewie RBT o n kluczach?
Wskazówka. Jeśli n jest postaci $2^k - 1$, to drzewie RBT o n kluczach może nie być żadnego węzła czerwonego (drzewo pełne z samymi czarnymi węzłami).
4. **[*]** Niech T będzie n -elementowym drzewem *SPLAY*, w którym zapamiętane są kolejne liczby naturalne $1, 2, \dots, n$. Przedstaw taki ciąg operacji *splay()*, w wyniku którego otrzymamy drzewo o wysokości $n - 1$.
5. **[**]** Zaimplementuj operację *splay* dla drzewa *SPLAY*. Dla uproszczenia zakładamy, że w każdym węźle drzewa oprócz wskaźników na lewego i prawego syna pamiętamy jest jeszcze wskaźnik do ojca.
6. **[*]** Zbuduj B -drzewo o minimalnym stopniu wierzchołka $t = 3$ o najmniejszej możliwej liczbie węzłów dla elementów o kluczach: $1, \dots, 35$. Dla powstałego drzewa wykonaj operację *insert(36)*.
7. **[*]** Zbuduj B -drzewo o minimalnym stopniu wierzchołka $t = 3$ o największej możliwej liczbie węzłów dla elementów o kluczach: $1, \dots, 17$. Dla powstałego drzewa wykonaj operację *delete(13)*.
8. **[**]** Do początkowo pustego 2-3-4-drzewa wstawiamy klucze: $1, 2, \dots, n$. Jaką wysokość ma końcowe drzewo? Odpowiedź uzasadnij.
9. **[**]** Zmodyfikuj strukturę B -drzewa tak, aby oprócz efektywnie wykonywanych operacji słownikowych można było też wykonywać dodatkowe instrukcje:
 - (a) *min()* — zwraca element minimalny w słowniku;
 - (b) *max()* — zwraca element maksymalny w słowniku;
 - (c) *med()* — zwraca element środkowy w słowniku (klucz, który jest medianą spośród wszystkich pamiętanych aktualnie kluczy).

Czas wykonania wymienionych instrukcji powinien być stały $O(1)$. Podaj algorytmy realizujące te operacje oraz ewentualne zmiany w procedurach słownikowych *search*, *insert* i *delete*.