
ALGORYTMY I STRUKTURY DANYCH

ZŁĄCZALNE KOLEJKI PRIORYTETOWE, ZBIORY ROZŁĄCZNE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

-
-
1. [*] Wykaż, że w drzewie dwumianowym stopnia k , które zawiera 2^k węzłów, długość ścieżki od korzenia do dowolnego węzła w tym drzewie jest nie większa niż k .
 2. **[**] Jak zaimplementować drzewo dwumianowe k -tego stopnia w tablicy 2^k elementowej? Twoje rozwiązanie powinno pozwalać na łatwe łączenie dwóch drzew dwumianowych. Napisz wzory pozwalające ustalić na podstawie indeksu w tablicy numer poziomu, na którym węzeł się znajduje, jego stopień oraz indeksy ojca, brata i kolejnych synów.
 3. [***] Dane jest drzewo dwumianowe stopnia k , które zawiera 2^k węzłów. W drzewie tym jest zachowany porządek kopcowy. Następnie zmieniamy w dowolnym wybranym węźle pamiętaną w nim wartość. Porządek kopcowy może więc zostać zaburzony. Jeśli zmienimy wartość na większą, to zwykle *sianie w górę* przywróci ten porządek w $O(\log n)$ krokach, a więc w akceptowalnym czasie. Jeśli natomiast zmienimy wartość na mniejszą, to przywrócenie porządku kopcowego poprzez proste *sianie w dół* będzie działać istotnie dłużej ($O(\log^2 n)$ kroków). Opracuj metodę, która po zmianie wartości na mniejszą we wskazanym elemencie przywróci porządek kopcowy w czasie $O(\log n)$.
 4. [***] *Drzewem lewicowym* nazywamy drzewo binarne, w którym dla każdego wierzchołka x prawa skrajna ścieżka od x do liścia jest najkrótszą ścieżką od x do wskaźnika pustego w poddrzewie zakorzenionym w x . Opracuj implementację złączalnej kolejki priorytetowej, korzystając z drzew lewicowych. Złożoność operacji kolejkowych *insert*, *extract-max* i *union* powinna być logarytmiczna względem liczby wszystkich elementów w kolejce.
Wskazówka. Zastanów się, jak połączyć dwa kopce lewicowe wzdłuż prawych skrajnych ścieżek.
 5. [*] Dana jest rodzina n jednoelementowych zbiorów rozłącznych. Zbiory te będziemy pamiętali w postaci drzewiastej ze zbalansowanym łączeniem. Skonstruuj ciąg $n - 1$ operacji *union*, by w ich wyniku powstał jeden zbiór reprezentowany przez n elementowe drzewo o wysokości $\lfloor \log n \rfloor$.
 6. **[**] Dany jest graf nieskierowany $G = (V, E)$ z ważonymi krawędziami. Jak za pomocą operacji na zbiorach rozłącznych wyznaczyć minimalne drzewo rozpinające grafu? Omów algorytm rozwiązujący to zadanie i przeanalizuj jego złożoność. Ile razy nastąpi wywołanie *find()* a ile razy *union()* (odpowiedź uzależnij od wielkości $|V|$ i $|E|$)?
 7. **[**] Zaprojektuj efektywną strukturę danych do wykonywania ciągów podanych operacji na dynamicznie zmieniającym się podziale zbioru $\{1, 2, \dots, n\}$:
 - **init()** — utworzenie podziału $\{\{1\}, \{2\}, \dots, \{n\}\}$;
 - **union(A,B)** — połączenie zbiorów A i B bieżącego podziału w jeden zbiór;
 - **find(x)** — wyznaczenie zbioru, do którego należy element x ;
 - **find-smallest()** — wyznaczenie zbioru bieżącego podziału, który ma najmniej elementów.

Pesymistyczna złożoność tych operacji powinna być rzędu $O(\log n)$.
 8. [***] Niech dany będzie ciąg m instrukcji *union* i *find*, w którym wszystkie instrukcje *union* występują przed instrukcjami *find*. Udowodnij, że koszt czasowy wykonania tych operacji na drzewiastych strukturach reprezentujących zbiory rozłączne ze zbalansowanym łączeniem i kompresją ścieżek wynosi $\Theta(m)$.
 9. [***] Operacja *extract(i)* ma usuwać element i -ty ze zbioru, do którego on aktualnie należy i tworzyć nowy jednoelementowy zbiór $\{i\}$. Podaj implementację operacji *extract*, która pozwoli na wykonanie ciągu m instrukcji *union*, *find* i *extract* na n -elementowym zbiorze danych w czasie $O(m \log^* n)$.