

ALGORYTMY I STRUKTURY DANYCH

KOLEJKI PRIORYTETOWE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. [***] Jak odwrócić kolejność zapisanych w pliku wartości (ostatnia wartość w pliku powinna się znaleźć na początku, przedostatnia na drugim miejscu, itd.)? Twój algorytm może korzystać z trzech plików i stałej ilości komórek pamięci operacyjnej; jego czas działania powinien być rzędu $O(n \log n)$ dla pliku, w którym zapisano n wartości.
2. [***] W pliku zapisany jest ciąg n liczb rzeczywistych. Początkowo wszystkie liczby w pliku były posortowane. Następnie następnie w pliku tym zmodyfikowano 7 wartości. Zadanie polega więc na posortowaniu n liczb zapisanych w pliku z co najwyżej stałą ilością $O(1)$ zaburzeń. Podaj algorytm rozwiązujący ten problem w liniowym $O(n)$ czasie i używający tylko kilku $O(1)$ dodatkowych komórek pamięci operacyjnej.
3. [**] Podaj iteracyjną i rekurencyjną implementację operacji siania w górę `sift-up()` i w dół `sift-down()`.
4. [*] Czy odwrotnie posortowana tablica jest kopcem?
5. [**] Na wykładzie zdefiniowany został kopiec binarny oraz przedstawiona została jego implementacja tablicowa. Uogólnij pojęcie kopca na kopce k -arne? Jak taki kopiec włożyć w tablicę i jak potem wyliczać indeksy synów i ojca dla danego elementu? Może przykład kopca 3-arnego pozwoli lepiej zrozumieć działanie ogólnych mechanizmów...
Wskazówka: Użyj tablicy indeksowanej od 0.
6. [***] *Drzewem lewicowym* nazywamy drzewo binarne, w którym dla każdego wierzchołka x prawa skrajna ścieżka od x do liścia jest najkrótszą ścieżką od x do liścia w poddrzewie zakorzenionym w x . Opracuj implementację kolejki priorytetowej, korzystając z drzew lewicowych. Złożoność operacji kolejkowych `insert` i `extract-max` powinna być logarytmiczna względem liczby wszystkich elementów w kolejce.
Wskazówka: Zastanów się, jak połączyć dwa kopce lewicowe wzdłuż prawych skrajnych ścieżek.
7. [*] Wykaż, że *heap-sort* nie jest stabilny.