

Algorytmy i Struktury Danych II, Zestaw 2

KOLEJKI PRIORYTETOWE

Przedmiotem zadania jest zaimplementowanie kolejki priorytetowej za pomocą kopca binarnego i wykorzystanie jej do sortowania tablicy zawierającej zmienne typu `int`.

1. Proszę zapoznać się z definicją i własnościami kopca binarnego.
2. Proszę zaimplementować funkcję `BudujKopiec` która wymusza na wyjściowej liście `T[i]` spełnienie własności kopca $T[i] \leq T[\text{indeks ojca } i]$. W tym celu należy rekurencyjnie sprawdzić, czy wartość `T[i]` dla synów jest mniejsza lub równa niż dla ich ojca. Jeśli warunek ten nie jest spełniony, to należy zamienić ojca z większym z synów.
“Sama metoda przywracania własności kopca polega na sprawdzeniu, czy dane poddrzewo (powiedzmy, o korzeniu o indeksie `i`-tym) spełnia cechy sterty – tzn. czy element pod indeksem `i`-tym jest nie mniejszy niż dwaj jego synowie. Jeśli tak nie jest, to należy zamienić ojca z większym z synów (czyli zachodzi „spłynięcie w dół” elementu), a następnie należy po raz kolejny sprawdzić, czy struktura kopca jest spełniona – tym razem dla poddrzewa o indeksie korzenia, gdzie znajduje się element, który przed chwilą przesunął się z poziomu o 1 wyższego. Aby cała sarta posiadała własność kopca, tę własność musi mieć każde jej poddrzewo.”
3. Proszę zaimplementować funkcję `PobierzNajwiekszy` która pobiera korzeń drzewa i zastępuje go najbardziej prawym z liści najniższego poziomu drzewa, po czym przywracana jest własność kopca.
4. Proszę zaimplementować funkcję `WstawElement` która dodaje element jako nowy liść na najniższym poziomie drzewa. Następnie, w razie potrzeby, element ten jest przesuwany na wyższe poziomy (zamieniając z kolejnymi ojcami), aż do momentu, gdy całe drzewo będzie posiadało własność kopca.
5. Proszę zaimplementować funkcję `HeapSort` sortowania kopcowego. “Procedura sortowania kopcowego najpierw buduje kopiec z losowych danych zapisanych w tablicy `T`, następnie bierze największy element kopca (korzeń) i zamienia go z ostatnim węzłem kopca (o największym indeksie). W tym momencie pozostaje już tylko zmniejszyć rozmiar sterty i przywrócić jej prawidłową strukturę. Pobieranie kolejnych największych wartości z kopca oraz przywracanie własności kopca wykonuje się w pętli do momentu, gdy zostanie nam kopiec 2-elementowy – następuje ostatnia zamiana `T[1]` z `T[2]`, po której wszystkie dane tablicy `T` będą już posortowane.

Materiały pomocnicze:

[Link 1](#)

[Link 2](#)