

Algorytmy i Struktury Danych

Lista zadań 4 - sortowanie, drzewa, rekurencja

1. Napisz procedurę, `counting_sort(int t[], int n, int c)`; kóra posortuje metodą przez zliczanie liczby w tablicy `t[]` względem cyfry `c`. $c = 0$ oznacza cyfrę jedności, $c = 1$ cyfrę dziesiątek itd...
2. Posortuj metodą sortowania pozycyjnego liczby: 1, 34, 123, 321, 432, 132, 543, 651, 91, 32, 987, 910, 643, 641, 12, 342, 498, 987, 965, 122, 121, 431, 350.
3. Które z procedur sortujących: (a) insertionsort (przez wstawianie), (b) quicksort (szybkie), (c) heapsort (przez kopcowanie), (d) mergesort (przez złączanie), (e) counting sort (przez zliczanie) (f) radix sort (pozycyjne), (g) bucket sort (kubelkowe) są stabilne. W każdym przypadku uzasadnij stabilność lub znajdź konkretny przykład danych, dla których algorytm nie zachowa się stabilnie.
4. Wykonaj operację Union dla kopców dwumianowych o 7 i 5 węzłach.
5. Narysuj przykładowy kopiec dwumianowy o 17 kluczach. Wykonaj 4 razy operację Delete_min.
6. Do pustego drzewa czerwono-czarnego wstaw kolejno 20 przypadkowych liczb. Następnie usuń je w tej samej kolejności w jakiej wstawiałeś.
7. Jaka może być minimalna, a jaka maksymalna ilość kluczy w drzewie czerwono-czarnym ustalonym o czarnej wysokości równej h_B ?
8. Jak jest minimalna a jaka maksymalna liczba kluczy w drzewie czerwono-czarnym o czarnej wysokości h_b .
9. Znajdź maksymalną i minimalną wartość stosunku ilości węzłów czerwonych do czarnych w drzewie czerwono-czarnym?
10. Korzystając z twierdzenia o rekursji uniwersalnej rozwiąż następujące zależności:
 - (a) $T(N) = 5T(n/3) + n$,
 - (b) $T(N) = 4T(n/2) + n^2$,
 - (c) $T(N) = 9T(n/3) + n^2$,
 - (d) $T(N) = 6T(n/3) + n^2$,
 - (e) $T(N) = 3T(n/3) + n$,
 - (f) $T(N) = 5T(n/2) + n^2$,
 - (g) $T(N) = T(n/2) + 1$.
11. Jaka będzie złożoność algorytmu jeżeli:
 - (a) $T(1) = 1, T(n) = 4T(\lceil \frac{n}{2} \rceil)$,
 - (b) $T(1) = 1, T(n) = 8T(\lceil \frac{n}{2} \rceil)$.
12. Jaka jest liczba ruchów potrzebna do przełożenia n krążków różnej wielkości z patyczka A na patyczek C, jeśli można używać też patyczka B, a krążek większy nie może nigdy leżeć na mniejszym? Napisz program który wypisze wszystkie konieczne ruchy w postaci:
1. AC 2. AB 3. CB ...
13. Napisz program znajdujący wszystkie ustawienia 8 hetmanów na szachownicy takie, że żaden z nich nie szachuje innego. Oszacuj złożoność twojego rozwiązania.