SPRAWDZIAN III

Imię i nazwisko:

Nr indeksu:

Nr grupy:

Uwaga! Sprawdzian jest testem wielokrotnego wyboru, gdzie wszystkie możliwe kombinacje odpowiedzi są dopuszczalne (tj. zarówno wszystkie odpowiedzi poprawne, część odpowiedzi poprawna jak i brak odpowiedzi poprawnych). Poprawne odpowiedzi należy zaznaczyć, z lewej strony kartki, symbolem "+". Natomiast symbol "-" jak i brak symbolu przy odpowiedzi oznacza odpowiedź niepoprawną. Pytanie jest uznane za poprawnie rozwiązane (tj. +1pkt) wtedy i tylko wtedy gdy wszystkie jego odpowiedzi zaznaczone są poprawnie. Życzymy powodzenia ...

1. Które z poniższych zdań jest zawsze prawdziwe w dziedzinie słowników:
   (a) [–] $\mathit{empty}\,(d) \Leftrightarrow \mathit{empty}\,(\mathit{delete}\,(\mathit{insert}\,(\mathit{insert}\,(d,e)\,,e)\,,e))$,
   (b) [–] $\neg\mathit{member}\,(\mathit{insert}\,(\mathit{delete}\,(d,e)\,,e)\,,e)$,
   (c) [+] $\mathit{member}\,(d,e) \Rightarrow \neg\mathit{member}\,(\mathit{delete}\,(d,e)\,,e)$.

2. Niech T będzie drzewem BST powstałym przez kolejne wstawianie wierzchołków o etykietach $1,7,3,2,5,8$ do początkowo pustej struktury, wtedy:
   (a) [–] korzeniem drzewa T jest wierzchołek o etykiecie 3,
   (b) [+] rezultatem działania algorytmu PreOrder dla drzewa T jest ciąg etykiet $1,7,3,2,5,8$,
   (c) [–] usunięcie wierzchołka o etykiecie 1 z drzewa T wymaga określenia jego bezpośredniego następnika w rozważanym drzewie.

3. Niech T będzie drzewem AVL powstałym przez kolejne wstawianie wierzchołków o etykietach $1,7,3,2,5,8$ do początkowo pustej struktury, wtedy:
   (a) [–] korzeniem drzewa T jest wierzchołek o etykiecie 5,
   (b) [–] rezultatem działania algorytmu PostOrder dla drzewa T jest ciąg etykiet $2,1,3,5,8,7$,
   (c) [+] usunięcie wierzchołka o etykiecie 3 z drzewa T wymaga określenia jego bezpośredniego poprzednika/następnika w rozważanym drzewie.

4. Niech T będzie drzewem AVL składającym się z n wierzchołków, gdzie $n > 10^6$, wtedy:
   (a) [+] koszt operacji *member* na drzewie T jest nie większy niż $\sqrt{n}$ porównań etykiet wierzchołków drzewa,
   (b) [–] realizacja operacji *insert* w drzewie T może spowodować wykonanie więcej niż 6-ciu co najwyżej podwójnych rotacji,
   (c) [–] realizacja operacji *delete* dla pewnego wierzchołka drzewa T nie może spowodować wykonania więcej niż 6-ciu co najwyżej podwójnych rotacji.

5. Niech drzewo binarne T będzie implementacją n-elementowego słownika d, wtedy złożoność czasowa operacji:
   (a) [+] *member* dla słownika d jest $O\,(\lg n)$, jeżeli T jest drzewem AVL,
   (b) [–] *insert* dla słownika d jest $\Theta\,(\lg n)$, jeżeli T jest drzewem BST,
   (c) [+] *delete* dla słownika d jest $\Omega\,(1)$, jeżeli T jest drzewem AVL.

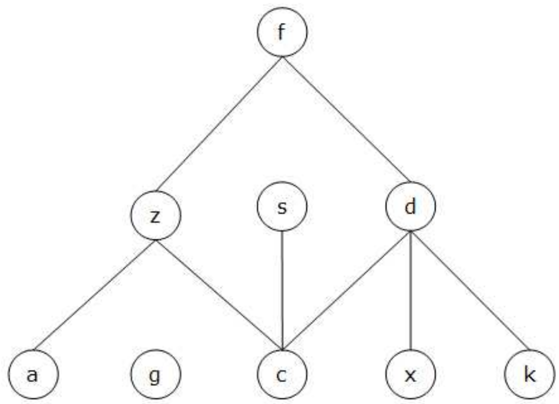
6. Które z poniższych zdań jest zawsze prawdziwe w dziedzinie kolejek priorytetowych:
   (a) [+] $min\,(pq) = min\,(insert\,(pq, min\,(pq)))$,
   (b) [–] $min\,(pq) \neq min\,(insert\,(delmin\,(pq)\,, min\,(pq)))$,
   (c) [+] $empty\,(pq) \Rightarrow empty\,(delmin\,(insert\,(pq, e)))$.
7. Niech H będzie 10^6-elementowym kopcem binarnym zaimplementowanym w drzewie binarnym T, wtedy:
   (a) [+] wysokość drzewa T jest rzędu $\lg 10^6$,
   (b) [–] drzewo T ma co najwyżej $\sqrt{10^6}$ wierzchołków wewnętrznych,
   (c) [+] liczba wierzchołków liści na ostatnim poziomie drzewa T jest równa co najmniej 1.
8. Rozważmy ciąg liczb $\alpha = 0, 4, 1, 2, 5$, wtedy:
   (a) [–] tablica reprezentująca kopiec binarny utworzony z elementów ciągu α przez kolejne wystawianie elementów do początkowo pustego kopca ma postać $[0, 1, 2, 4, 5]$,
   (b) [–] tablica reprezentująca kopiec binarny utworzony z elementów ciągu α przez zastosowanie szybkiej procedury budowy kopca (tj. HeapConstruct) ma postać $[0, 1, 2, 4, 5]$,
   (c) [+] tablica reprezentująca kopiec binarny utworzony z elementów ciągu α przez zastosowanie szybkiej procedury budowy kopca (tj. HeapConstruct) ma postać $[0, 2, 1, 4, 5]$.
9. Rozważmy ciąg n liczb naturalnych, do którego zastosowano algorytm HeapSort, wtedy:
   (a) [+] koszt budowy kopca binarnego wynosi $\Omega\,(n)$,
   (b) [+] koszt rozebrania kopca binarnego wynosi $O\,(n \lg n)$,
   (c) [+] algorytm HeapSort jest optymalnym algorytmem sortującym przez porównania.
10. Rozważmy kopiec binarny-drzewo H powstały przez kolejne wstawianie liczb $2, 3, 4, 5, 6$ do początkowo pustej struktury, wtedy:
    (a) [–] etykiety drzewa odczytane w porządku PostOrder tworzą ciąg $6, 5, 4, 3, 2$,
    (b) [+] jeżeli wykonamy ciąg operacji $insert\,(H, 1)$, $min\,(H)$, to etykiety drzewa odczytane w porządku InOrder tworzą ciąg $5, 3, 6, 1, 4, 2$,
    (c) [+/–] koszt operacji $insert$ na strukturze H jest rzędu liniowego względem liczby wierzchołków drzewa.
11. Rozważmy kopiec binarny-drzewo H powstały przez kolejne wstawianie liczb $6, 5, 4, 3, 2$ do początkowo pustej struktury, wtedy:
    (a) [+] etykiety drzewa odczytane w porządku PreOrder tworzą ciąg $2, 3, 6, 4, 5$,
    (b) [+] jeżeli wykonamy ciąg operacji $delmin\,(H)$, $delmin\,(H)$, to etykiety drzewa odczytane w porządku InOrder tworzą ciąg $6, 4, 5$,
    (c) [+] koszt operacji $delmin$ na strukturze H jest rzędu liniowego względem wysokości drzewa.
12. Rozważmy graf pełny z wagami G składający się z n wierzchołków, wtedy:
    (a) [–] koszt pamięciowy macierzy sąsiedztwa grafu G jest rzędu $O\,(n)$,
    (b) [–] koszt pamięciowy tablicy list incydencji grafu G jest rzędu $\Theta\,(n)$,
    (c) [+] koszt pamięciowy macierzy sąsiedztwa grafu G jest rzędu kosztu pamięciowego tablicy list incydencji grafu G.

13. Rozważmy graf $G = (V, E)$ przedstawiony na poniższym rysunku, wtedy:

    (a) [+] kolejność odwiedzania wierzchołków grafu algorytmem DFS jest zgodna z porządkiem a, z, f, d, x, k, c, s jeżeli wierzchołkiem startowym jest wierzchołek a,

    (b) [–] kolejność odwiedzania wierzchołków grafu algorytmem DFS jest zgodna z porządkiem z, s, d, f, x, k, s, c jeżeli wierzchołkiem startowym jest wierzchołek c,

    (c) [+] maksymalna wysokość stosu pomocniczego w algorytmie DFS zastosowanym do wierzchołka startowego s jest równa 3.



14. Rozważmy graf $G = (V, E)$ z zadania 13-tego, wtedy:

    (a) [+] kolejność odwiedzania wierzchołków grafu algorytmem BFS jest zgodna z porządkiem f, d, z, c, k, x, a, s jeżeli wierzchołkiem startowym jest wierzchołek f,

    (b) [+] kolejność odwiedzania wierzchołków grafu algorytmem BFS jest zgodna z porządkiem x, d, c, f, k, s, z, a jeżeli wierzchołkiem startowym jest wierzchołek x,

    (c) [+] złożoność czasowa algorytmu BFS dla rozważanego grafu jest rzędu $O(|V| + |E|)$.

15. Niech G będzie pewnym grafem skierowanym, wtedy algorytm sortowania topologicznego grafu G:

    (a) [–] działa poprawnie wttw, gdy graf ten jest grafem spójnym,

    (b) [–] działa poprawnie wttw, gdy graf ten jest grafem z wagami,

    (c) [–] ma złożoność rzędu $O(|V|)$, gdzie V jest zbiorem wierzchołków rozważanego grafu.

16. Zapis autentycznej rozmowy radiowej przeprowadzonej między amerykańskim okrętami a Kanadyjczykami. Miała ona miejsce w październiku 1995r. u wybrzeży Nowej Funlandii.

    - *Kanadyjczycy:* Proszę o zmianę kursu o 15 stopni na południe w celu uniknięcia kolizji.
    - *Amerykanie:* Radzimy wam zmienić kurs o 15 stopni na północ, aby uniknąć kolizji.
    - *Kanadyjczycy:* To niemożliwe. To wy będziecie musieli zmienić kurs o 15 stopni na południe, aby uniknąć kolizji.
    - *Amerykanie:* Mówi kapitan okrętu wojennego Stanów Zjednoczonych. Powtarzam ponownie: wy zmieńcie kurs.
    - *Kanadyjczycy:* Nie. Powtarzam: zmieńcie kurs, aby uniknąć kolizji.
    - *Amerykanie:* Mówi kapitan lotniskowca USS "Lincoln" - drugiego pod względem wielkości okrętu bojowego amerykańskiej marynarki wojennej floty atlantyckiej. Towarzyszą nam trzy niszczyciele, trzy krążowniki i wiele innych okrętów wspomagania. Domagam się, abyście to wy zmienili kurs o 15 stopni na północ. W innym przypadku podejmiemy kontrdziałania w celu obrony grupy!
    - *Kanadyjczycy:* Mówi latarnia morska: wasz wybór!

    Jak skończyła się owa historia:

    (a) dobrze,

    (b) źle,

    (c) tego nie wiedzą nawet najstarsi górale.