

Structuri de Date si Algoritmi

Examen (A)

14 iunie 2019

Timp de lucru: 150 min

Nume student _____

Grupa _____

1. (10 puncte) Se da heap-ul din imaginea de mai jos (reprezentarea fizica).
 - (a) Dati reprezentarea logica (de arbore) a heap-ului.
 - (b) Desenati heap-ul dupa aplicarea fiecareia din operatiile urmatoare: Heap-extract-min(), Heap-push(4), Heap-extract-min(), Heap-extract-min().

1	3	1	5	9	7	3	7	5	9
---	---	---	---	---	---	---	---	---	---

2. (10 puncte) Calculati numarul de apeluri ale functiei $op()$ pentru fiecare din secventele urmatoare de cod, in functie de N . Presupunand ca $op()$ are complexitatea $O(1)$ in cazul defavorabil, care este complexitatea fiecarui algoritm de mai jos, in cazul defavorabil?

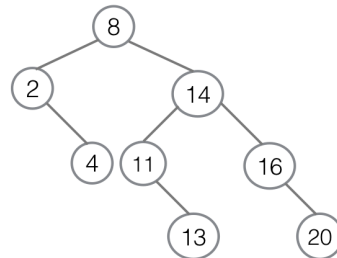
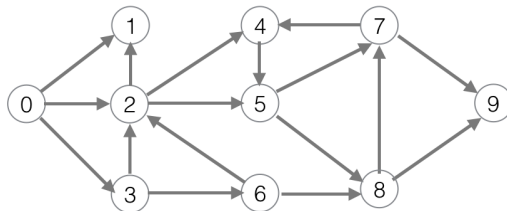
(a) $f1(int\ N)$
 for $i = 0$ to $N/2 + 100$
 $op()$

(b) $f2(int\ N)$
 for $i = 0$ to $N*N - 3$
 for $j = 0$ to i
 $op()$

(c) $f3(int\ N)$
 if $N = 1$ then
 $op()$
 else
 $f3(N/2)$

(d) $f4(int\ N)$
 if $N = 3$ then
 $op()$
 else
 $f4(N-1) + f4(N-1)$

3. (10 puncte) Se da graful de mai jos, stanga.
 - (a) Dati reprezentarea prin liste de adiacenta a grafului.
 - (b) Aplicati algoritmul $BFS(0)$. Desenati arborele rezultat.
 - (c) Este graful bipartit? **Motivati.**



4. (10 puncte) Pe graful de la problema 3:
 - (a) Desenati arborele rezultat in urma aplicarii algoritmului $DFS\ Visit(0)$.
 - (b) Marcati, pe arbore, timpii de descoperire si de finalizare ai fiecarui nod parcurs.
 - (c) Se poate construi o sortare topologica pe graful dat? **Motivati.**
5. (10 puncte) Se da arborele binar de cautare din imaginea de mai sus (dreapta).
 - (a) Scrieti secventele de chei care rezulta la parcurgerile in pre-, respectiv post-ordine ale arborelui.
 - (b) Este arborele un arbore AVL? **Motivati.**
 - (c) Desenati arborele, dupa aplicarea fiecareia din operatiile succesive: Tree-Delete($T, 8$), Tree-Insert($T, 10$), Tree-Insert($T, 8$).
6. (10 puncte) Se da o tabela de dispersie de dimensiune $m = 13$, ce utilizeaza adresare deschisa si verificare patratica, cu functia de dispersie: $h(key, i) = (key \% m + 3 * i + i^2) \% m$.
 - (a) Desenati cum arata tabela la finalul secventei de operatii: hash-insert(3), hash-insert(13), hash-insert(42), hash-insert(6), hash-insert(52), hash-insert(55), hash-insert(5).
 - (b) Cate celule se acceseaza la fiecare din cautarile urmatoare: hash-search(55), hash-search(12), hash-search(29)?

7. (10 puncte) Comparati si contrastati tehnica Greedy cu Branch & Bound (aplicabilitate, strategii generale, complexitate, caracteristicile solutiilor gasite, etc).
8. (10 puncte) Presupuneti ca aveti o stiva in care adaugati (*push*) elementele de la a la f in ordine, dar puteti sterge (*pop*) un element la orice moment. Dati secventa de operatii *push* si *pop* pe o stiva initial goala, pentru a obtine fiecare din urmatoarele ordini de stergere a elementelor (daca nu exista o astfel de secventa, motivati de ce nu exista): (i) b, d, e, f, c, a (ii) a, f, e, d, b, c, (iii) a, c, f, e, d, b .
9. (10 puncte) Mai jos aveti mai multe secvente de siruri de caractere; coloana din stanga - (0) reprezinta secventa originala, cea din dreapta - (5) reprezinta secventa sortata; coloanele (1)-(4) reprezinta continutul secventelor la un pas intermediar (nu neaparat acelasi) al unuia din cei patru algoritmi de sortare numerotati de la (a) la (d) - vezi mai jos. Potriviti fiecare algoritm cu secventa intermediara corespunzatoare.

Original (0)	(1)	(2)	(3)	(4)	Sortat (5)
sansa	bran	arya	bran	bran	arya
tormund	jon	bran	cersei	brienne	bran
bran	ned	brienne	dani	cersei	brienne
ned	sansa	cersei	jon	dani	cersei
jon	tormund	dani	ned	jaime	dani
dani	brienne	davos	sansa	jon	davos
tyrion	cersei	euron	tormund	grey-worm	euron
cersei	dani	grey-worm	tyrion	arya	grey-worm
brienne	jaime	tormund	brienne	jorah	jaime
jaime	tyrion	jaime	jaime	davos	jon
jorah	jorah	jorah	jorah	melisandre	jorah
theon	melisandre	theon	theon	euron	melisandre
sandor	misandei	sandor	sandor	misandei	misandei
melisandre	sandor	melisandre	melisandre	ned	ned
misandei	theon	misandei	misandei	samwell	samwell
grey-worm	arya	ned	grey-worm	sandor	sandor
arya	davos	sansa	arya	sansa	sansa
samwell	euron	samwell	samwell	theon	theon
davos	grey-worm	jon	davos	tormund	tormund
ygritte	samwell	ygritte	ygritte	tyrion	tyrion
euron	ygritte	tyrion	euron	ygritte	ygritte

(a) SelectionSort (b) BubbleSort (c) MergeSort (d) InsertionSort

10. (10 puncte) Problema *rucsacului*: Se da un rucsac a carui capacitate este $W = 100$, si obiectele din tabelul de mai jos, fiecare avand greutatea w_i si profitul p_i . Presupunand o solutie bazata pe **branch and bound** pentru aceasta problema, desenati cum arata spatiul de cautare pentru problema data:

Obiect	A	B	C	D	E	F
Greutate(w_i)	75	60	25	20	10	5
Profit (p_i)	25	12	12.5	4	10	2

11. (20 puncte) Descrieti un algoritm liniar care verifica daca un arbore binar dat este arbore binar de cautare. Motivati (analizati) complexitatea algoritmului propus.
12. (20 puncte) Lui Doru ii place sa joace baschet, si in acelasi timp sa si castige. El are foarte multi prieteni care stiu si ei sa joace baschet. Cu cat un jucator de baschet este mai inalt, cu atata el este mai bun. Fiecare echipa este compusa din k jucatori. Pe masura ce prietenii lui Doru afla ca acesta si-a facut echipa de baschet, ei vin sa se inscrie in echipa lui. Dati un algoritm care sa il ajute pe Doru sa poata alege cea mai buna echipa posibila, la orice moment. Analizati complexitatea algoritmului propus.
13. (20 puncte) Doua siruri de caractere sunt *echivalente* daca unul este anagrama a celuilalt. O clasa de echivalenta este o multime in care orice doua elemente sunt echivalente. De exemplu in sirul $A = ["aeb", "aba", "aab", "eba", "baa"]$ avem doua clase de echivalenta: $["aeb", "eba"]$ si $["aba", "aab", "baa"]$. Descrieti un algoritm care primeste un sir A de siruri de caractere si gaseste numarul de clase de echivalenta din acesta. Analizati complexitatea algoritmului propus. Nota: pentru credit maxim, algoritmul vostru ar trebui sa aiba complexitatea $O(NM(\log N + \log M))$ in cazul defavorabil, unde N este lungimea sirului A si M este lungimea maxima a unui sir de caractere.