ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

EE2008 / IM1001

NANYANG TECHNOLOGICAL UNIVERSITY

SEMESTER 2 EXAMINATION 2015-2016

EE2008 / IM1001 - DATA STRUCTURES AND ALGORITHMS

April / May 2016

Time Allowed: 21/2 hours

INSTRUCTIONS

- 1. This paper contains 4 questions and comprises 4 pages.
- 2. Answer ALL questions.
- 3. All questions carry equal marks.
- 4. This is a closed-book examination.
- 5. Unless specifically stated, all symbols have their usual meanings.
- 1. (a) Determine the asymptotic upper bound for the number of times the statement "y = y + 2" is executed in each of the following algorithms.

(i) for
$$i=1$$
 to n
for $j=i+1$ to n
for $k=j+1$ to n
 $y=y+2$

(ii) for
$$i=1$$
 to n
for $j=i$ to $2i$
 $y=y+2$

(10 Marks)

Note: Question No. 1 continues on page 2.

EE2008 / IM1001

- (b) (i) Solve the following recurrence relation: $b_n = \frac{b_{n-1}}{1+b_{n-1}}$, $b_0 = 1$, where n is a non-negative integer.
 - (ii) Determine whether the following statement is true or false. Justify your answer.

If
$$f(n) = \Theta(g(n))$$
 then $g(n) = \Theta(f(n))$.

(8 Marks)

(c) A pointer *start* points to the first element of a doubly-linked list L. Write an algorithm that reverses the elements of L. You are not allowed to use any additional data structure in your solution.

(7 Marks)

- 2. (a) (i) Let Q be a non-empty queue and S be an empty stack. Using the stack and queue ADT functions and the stack S, write an algorithm to reverse the order of the elements in Q.
 - (ii) Draw the 7-item hash table resulting from hashing the keys 19, 26, 13, 48, and 17 using the hash function $h(x) = x \mod 7$. Assume that collisions are handled by double hashing using a second hash function $h^2(x) = 5 (x \mod 5)$.

(9 Marks)

(b) Assume that the LIST ADT is implemented using a doubly linked list. Using pseudo-code, describe the implementation of the method insertBefore(p,e) of the LIST ADT.

(6 Marks)

(c) Suppose that the data stored at each node in a binary search tree is a positive integer. Write a recursive algorithm that finds the sum of all values, which are less than a given value x in the binary search tree.

(10 Marks)

EE2008 / IM1001

3. (a) Show clearly what the following array looks like in each step of the siftdown algorithm when applied at the index i=2, assuming that we are restoring a maxheap and the index of the array starts at 1.

(b) Continuing from your answer obtained in part (a), show each step in the heapsort algorithm.

(12 Marks)

(c) Write an algorithm using counting sort to sort an array of integers in the range [-k, k]. You may make use of the functions discussed in lecture to construct your algorithm.

(8 Marks)

4. (a) Use the depth first search (dfs) algorithm starting at vertex 1 to perform topological sorting of the directed acyclic graph shown in Figure 1. Explain each step clearly by drawing the dfs trees generated and the output array at each step.

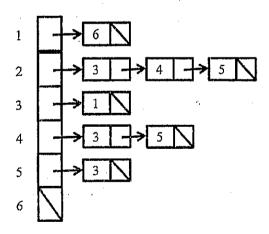


Figure 1

(10 Marks)

Note: Question No. 4 continues on page 4.

Mary Sanda Control

EE2008 / IM1001

(b) Write an algorithm that finds the sum of the in-degrees of all the vertices in a directed graph. Assume that the directed graph is represented by an adjacency list.

(10 Marks)

(c) What is the time complexity of your algorithm in part (b) in terms of the number of vertices and edges? Justify your answer.

(5 Marks)

END OF PAPER

n n	SEMESTER Z EXAMINATION 2015-2016 (1/9
***************************************	EE2008 / IM/001 - PATA STRUCTURES AND ALGORITHMS	No.
ارم	(i) = = = = = = = = = = = = = = = = = = =	A A A A A A A A A A A A A A A A A A A
The state of the s	= = = = = = = = = = = = = = = = = = =	
	$\int_{-\frac{1}{2}+1}^{N} (n-j) = n - (i+1) + n - (i+2) + n - (i+3) + \cdots + n - (i+n-i)$	
hanny ammunine Mahaliani dalah	$= n - 1 - 1 + n - 1 - 2 + n - 1 - 3 + n - 1 - 4 + \cdots + n - 1 - (n - 1)^{2} - (1 + 2 + 3 + \cdots + n - 1)$)
		V
3	$\frac{N}{\sum_{i=1}^{N}(N-i)^{2}} = \frac{1}{\sum_{i=1}^{N}N^{2} + \frac{N}{\sum_{i=1}^{N}2^{N}i} \leq N^{3} + \frac{N(N+1)(2N+1)}{6} = 0$	(n³)
1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 -	(ii) $\frac{2}{2} = \frac{2}{2} (2i-i+1) = \frac{2}{2} (i+1) = \frac{2}{2} + \frac{2}{2} = \frac{2}{2} + 1$	ハ=0(い2) 井
1167	(i) bn = bn-1 bo=1 n is a non-negative integer.	
	$b_0 = 1$ $b_1 = \frac{b_1 - b_2}{1 + b_2} = \frac{b_0}{1 + b_2} = \frac{1}{1 + b_2} = \frac{1}{1 + b_2}$	
	$b_1 = \frac{b_1}{b_1} = \frac{b_1}{b_2} = \frac{b_1}{b$	
	63=	
· Marine and the second	bn= bn-1 bn-1 = bn-2	7
- non-reconstruction (Assessment of the Assessment of the Assessme	6h-2 [+bh-2 DN-2 L	
	= bn-2 = >bn-2 +1 : bn-2 +1 bn-3	2000 etc.
The state of the s	= <u>bn-3</u> = 3bn-3+1	2 C C C C C C C C C C C C C C C C C C C
, and the same of	= bn-4	
- Name -	नियमना	near of day
	T T	
	(ii) If $f(\alpha) = \Theta(g(\alpha))$	J. Order
- Millians consequence of the	in f(n) = k, g(n) for all n≥no and kpo	1. Company
	f(n) > k2 g(n) for all n>no and k20	
	prove g(n) = ksf(n) & g(n) & k4f(n)	
	for $f(n) \leq k_1 g(n)$ for all $n \geq N_0$ and $k_1 > 0$ $k_1 f(n) \leq g(n)$	
	shows that gin> (s+ch)	AZONE
		* * * * * * * * * * * * * * * * * * * *

	Dale: No.
	for fin>kigen for all h>no and k2>0
·	g(n) < K2 f(n)
14	$\frac{g(n) \leq k_2 \int_{C} f(n)}{f(n)}$
 ;	Shows that g(n) < K4-f(n)
	Thus, if fin=0g(n) then g(n)=60fcn. #
	Next Next Next
	NI 2 3 4 L'élements
	prev. prev prev
	Prev Prev prev
	NITT BIT 14N
	Next Wext Wext
	While (Start. next!= null) {
	Swap (Start next clata, Start prev. data)
<u> </u>	Start = Start next
	return;
	- - - -
	3

The idea is let all values in Q insert into S firstly. (Q is FIFO & S is FILO) Secondly, put all values back into Q but will be reversed. (2/4)

•		Date: No.
2(a)	(i) Q > Non-empty queue of @ B	OO: Y
· J		
- INWINI	Es-> Empty Stack /	0 1 2 S. bength -1
	EMPTY S	
		1
_Step 1	. While (r!=-1) {	Now t.
	1.1 front() {	Quene of empty
	if (empty())	
	throw QEmptyException	Stock &
	else return Qifj }	Step 2.
ĺ	val = QTf] // Set val equals to	
	front value of Qu	
	1.2 push (val) {	if Siempty 1) then
	>	Hyraw Empty Stack Etragion
Jan State St	if t == Sibergth - 1 thon	else
<u>98 j. a. 4.</u>	throw Full Stanck Exception	return SEt] }
<u> </u>	else , Ll	Val2 = 57t] // Last itom
10 (11 (S) 10 (S)		ualue in the Stack
115 242 1 194 1 1 1 2 1		
Marks of the Section of the Control	} // Insert the Ual into Stack	2.2 enqueue (valz) {
		A (outby)
	1.3 dequeue(){	r=f=0
	if (empty)	
	throw Q Empty Exception	relse §
	else {	r=r+1 if(r==0.512e) r=0
	大(L==+)	ifr==f
prácolní de distribution de la constantina della	r=f=-1	-thru Full & Exception
	else {	-tunu full & txception
	f =ft1	g[r]=val2
	if (f== a. size) f=0	
·	<u> </u>	3 // Insert the val to Q
· · · · · · · · · · · · · · · · · · ·		2.3 70709
11.	3 // Debete the value in Q	if S. enjoyu then throw Empty Fack Exception
- 13.	} // While Q is not empty, repec	
, , , , , , , , , , , , , , , , , , ,	the stops until all elements	else t=t-13
-	One inserted into S.	// Remove the value in S } // Transfer all values in S to Q.
		Step3. return Q #

			· · · · · · · · · · · · · · · · · · ·					1	
			huxs = x mo	1d 7 h1(x)=	5- (Xmoo	(5)	132	Na	
2 (Oy	(11)	k	hlkj	d(K)	Pinha	· · · · · · · · · · · · · · · · · · ·		No.	
	7.7	19	٤	1	5	<u> </u>			
· •		26	<u>ئى</u>	4	(-	2 1	 		
		13	<u> </u>	2					
		48	6	2	6	, [
		17	<u> </u>	3	3	<u>,l</u>			
		1			1 3				
					`	·			
	19:	h(K):	=19 mod 7= 5	= d(k)= ((19 m	od 5):	=		
				= t ol(k)=					
				=6 d(k)					
	48	: hik	1 = 48 moo	17=6 dek	1 = F- (18 moo	(x)=2		
				7=3 24					
	17.	- / []	, , , , , , , , , , , , , , , , , , , ,			. 1 - 00	<u> </u>		
	コン	Fen ho	rdn table:						
		<u> </u>							
. 1.			48 26	17 19	13		丑		
**		0	- 1	3 4 5	6		11	······································	
· 1:			· · · · · · · · · · · · · · · · · · ·				 		
≥ (b)	inser	t.Before	(P,e)}	man and the second seco	· · · · · · · · · · · · · · · · · · ·			·	
.,			ew node		· r.			· · · · · · · · · · · · · · · · · · ·	.,
		•	ita= e	· · · · · · · · · · · · · · · · · · ·		•		1	
	11 * 1 * 1 * 1 * 1 * 1 * 1 * 1 * 1 * 1		reu= p. prev			·			
		•	ext = >		,		· · · · · · · · · · · · · · · · · · ·		
			rev. n'ext=(y			'.		·
			nev=v						
	.			#	·		,		
ري)د	Su	m (root	t,x)?						
) i	Croot = =	null)					
			return	0					
		رھ		ot oloita < X)					
				root. data t	Sum (root	, left, x) + sum	(root. right	/X)
		: .	else v	return sum (root let	$t_{\lambda}x)+\zeta$	Cum (roo	f.right, xy	
			3	* *	· · · · · · · · · · · · · · · · · · ·			J	
	•		J·	#				A'ZC	NE
				4.₹					

	Date: No.
3(a)	A 102 20 69 67 33 58 65 23 15
	1123456789
	Step 1 (102)
	1. Find larger Child 2. Swap
are one contributed data are	(69)
	5 6 7
Chila	(62) (33) (18) (2)
	8 7 9
	(23) (15) Sift down
·	Step 2. (102)
	2
1000 1000 1000 1000 1000 1000 1000 100	(67) (69) processor p
The state of the s	102 67 69 20 33 58 65 4 5 6/ >
	23 15 (26) (33) (38 (64)
	Child 8
	=2i(23)(5)
A Market as 15	
- marked by Berkel state of the	Step 3 2 (102), Siftdown
	67 69
	4 / 5 / 7 7
	(23) (33) (58) (65)
and the second s	8
	$(20)(15)^{1}$ $n=9$.
•••	

	A 102 67 69 23 33 58 65 20 15
	1,1,23456789
14.4	
L.	
•	
	A SANTA A TANDAR A TA

January Commence

 $\frac{1}{2} \left(\frac{1}{2} \left(\frac{1}{2} \right) + \frac{1}{2} \left(\frac{1}{2} \right) \right) = \frac{1}{2} \left(\frac{1}{2} \left(\frac{1}{2} \right) + \frac{1}{2} \left(\frac{1}{2} \right) \right)$

	1 2 3 4 5 6 7 pm 9 Date: No.
3(b)	1 2 3 4 5 6 7 8 9 Date: No. A 1/02 67 69 23 33 58 65 20 15
	Exchange A[1] with A[n] (A[1] with A[9])
	Treat A [1,, N-1] As a new heap. (A [1,2, 8] As a new heap)
A	15 67 69 23 33 58 65 20 102
,	(B) 69 (9)
	64 (6) 15 => 67 (D15 => 69 (G) Siftdown
	13 33 to 65 23 33 to 0 40 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	λ ₀ 26 8 6 5
/	A 69 67 65 23 33 58 15 20 102 C
<u>(2)</u>	CWAP ACIT & AT&J. Treat ATI, z 7. J as a new heap.
A	
	200 67
	67 65 => 63 65 => 33 63 Siftdown. 23 33 58 15 25 33 58 15 63 63 63 63
	25 33 58 15 23 33 58 15 63 23 68 (3)
	67 33 65 23 20 58 15 69 102
(g)	Swap A[1] &A[7]. Treat A[1,2,",6] As a new heap
	A 15 33 65 23 20 58 67 69 62
= =	38 (D => 33 (D => 35 (D) Siftdown
	38 (5) => 33 (G) => (5) Ftdown
<u> </u>	23 20 58 25 20 68 4 3 66
	65 33 58 23 20 10 67 69 102
	Swap A[1] with A[6]. Treat A[1, ", 1] as a new heap
(4)	
A	
,,	35 (5) => 2(3) (5) Siftdown
	23 20 (23 (2) 3
A	1 8 33 15 23 20 65 67 69 102
	AZONE

		Date: No.
	Swap A [1] with A [5] Treat A [1,, 4]	as a new heap
	A 20 33 15 23 58 65 67 69 102 -	
	2(2) 33 (33)	
,	(3) 15 => 200 15 =7 (3) (3)	Siftdown
		311,000.
- Andreas - 1	23 23 4	
	A 33 23 15 20 58 65 67 69 102 6	
<u>(6</u>	Swap A[1] with A[4]. Treat A[1,,3] as	a new heap
	A 20 23 15 23 58 65 67 69 102	
	7(29 (23)	
		sittolown
	<u> </u>	51,0003
	1 2 1 2 1 5 1 5 2 1 6 2 1	
	A 28 20 15 33 58 65 67 69 602	dise 1/4 est
a de la companya de l		
A Arthur Anna Anna Anna Anna Anna Anna Anna Ann		The second secon
·	Swap ACIJ & AC3]	
	A [15 20 23 33 68 65 67 69 102	#
	for (j= 1 to K) {	
3(0	Counting_sort (A,B,K) {	
	$n = A \cdot last$	
	for i=1 to k?	
	Countij=0	-for(i=-k to1){
4-9-19-19-19-19-19-19-19-19-19-19-19-19-1	7	
	-for j=1 ton ?	Counting_sort (A,B,K),
	Count CAC) 1 = Count[AC])]+1	>
	fon i=1 tok?	100 E 11110 -
	Count [i]= Count [i-] + Count [i]	return;
· · · · · · · · · · · · · · · · · · ·		
	for j=n downto 19	——————————————————————————————————————
	B [court ACj] = ACj]	
	Count [ACj]] = Count [ACj]]-1	
	*	
	3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4	
	But the state of t	

Section of the property of the

	Date: No.
419	Implementation = PFS
	dfs(1) 50 (2)
·	dfs (6)
	(3) (3) (3)
	6 🕏
	Sort: 1,6
4	
	(2) dfs(2) 5
of a love of	
	4/3
	(G) Sort. 3,1,6
	(2) d(m(t)) (D)
	(3) dfs(5)
	6 3 E D Sort: \$,3,1,6
المنتبية الماديات الماديات	D diciti Os as
	\$\Phi dfs(4) \Pi
i	43 Sort: 4,5,3,1,6.
ali in der	
	(P) dk(12) (P)
	6 //
	O 43 3
	sort: 2,4,5,3,16
	6 Sort Will be 2,4,5,3,1,6
and an arms	
·	

	Dola: No.
4(6)	Sum_indegroes (adj) {
	n= adj. last < sum = 0
	for i=1 to n
,	in degree [i]=0
	for i=1-bons
	hef = adj [i]
	white (ref!= null) {
	indegree [ref. data] = indegree [ref. data] +1
	ref = ref. next
	3
	for (i=1 ton) {
	0.00.00
	Sum = Sum + Sum [i]
	3
	Retnun Sum #
4(0)	The state of the Alice
<u> HO)</u>	Time complexity is O(n)
	for i=1 to n {
	ref = adjīli]
Total Need	- While (ref!=null)?
1 times	Happened indegree [nef. data] = indegree Tref. data] +1
	once for
	each i = lef : hext
	51= n. i Time complexing is O(N)
	T=1 4
·	7
	the state of the s

