NATIONAL UNIVERSITY OF SINGAPORE
SCHOOL OF COMPUTING

TERM TEST FOR
Semester 2, AY2018/2019

CS3230 – DESIGN AND ANALYSIS OF ALGORITHMS

9 Mar 2019 Time Allowed: 2 hours

Instructions to Candidates:

1. This paper consists of **FOUR** questions and comprises **TWELVE (12)** printed pages, including this page.
2. Answer **ALL** questions.
3. Write ALL your answers in this examination book.
4. This is an **OPEN BOOK** examination.

Matric. Number: ____________________
Tutorial Group Number: ____________________

QUESTION	POSSIBLE	SCORE
Q1	20	
Q2	20	
Q3	30	
Q4	20	
TOTAL	**90**	

IMPORTANT NOTE:

- *You can* ***freely quote*** *standard algorithms and data structures covered in the lectures and homeworks. Explain* ***any modifications*** *you make to them.*
- *Unless otherwise specified, you are expected to* ***prove (justify)*** *your results.*

Q1. (20 points) Sorting out order of growth rates

(a) Rank the following functions in *increasing order of growth*; that is, if function $f(n)$ is *immediately* before function $g(n)$ in your list, then it should be the case that $f(n)$ is $O(g(n))$.

$g_1(n) = \sum_{i=2}^{n} \frac{n^3}{i(i-1)}$	$g_2(n) = n^2 \lg\lg n$	$g_3(n) = n!$	$g_4(n) = n^{\lg n}$
$g_5(n) = 2^{2\cdot 2^{\lg\lg n}}$	$g_6(n) = (\lg n)^n$	$g_7(n) = n^3$	$g_8(n) = 2^n$

To simplify notations, we write $f(n) << g(n)$ to mean $f(n) = o(g(n))$ and $f(n) \equiv g(n)$ to mean $f(n) = \Theta(g(n))$. For example, the four functions n^2, n, $(2013n^2 + n)$ and n^3 could be sorted in increasing order of growth as follows: ($n << n^2 \equiv (2013n^2 + n) << n^3$). *Proofs are not required for this problem.*

(b) Can you show that $n \lg\lg n = O(n \lg n)$ by limit method?

Q1. (continued…)

(c) Can you show that $3n^2 + 2^{3 \lg n}\left(1 + \cos\left(\frac{n\pi}{2}\right)\right) = O(n^3)$ by definition? Please show the steps and state clearly what are c and n_0.

Q2. (20 points) Solving recurrence

(i) $T(n) = 2\,T(n/2) + n \lg \lg n$
(ii) $T(n) = 4\,T(n/2) + n$
(iii) $T(n) = 4\,T(n/2) + n \lg \lg n$
(iv) $T(n) = 4\,T(n/2) + n^2$
(v) $T(n) = 4\,T(n/2) + n^2 \lg \lg n$
(vi) $T(n) = 4\,T(n/2) + n^3$
(vii) $T(n) = 4\,T(n/2) + n^3 \lg \lg n$

(a) Among the above recurrences, please indicates those that can be solved by master theorem. For each recurrence that is solvable by master theorem, please indicate which case it belongs to (either case 1, 2, or 3) and state the time complexity.

Q2. (continued…)

(b) For recurrences that are not solvable by master theorem, please solve them.

Q2. (continued…)

Q3. (30 points) Divide-and-conquer

You are given an unsorted integer array A[1..n] of length n. Assume all integers in A[1..n] are distinct. Consider an array of m indices $s[1..m]$ such that $1 < s[1] < s[2] < \cdots < s[m] < n$. We would like to return the $s[k]$-th smallest elements of A[1..n] for all $k \in \{1, 2, \ldots, m\}$.

(a) Consider the following algorithm Select(A[1..n], s[1..m]) that return D[1..m] where D[k] is the s[k]-th smallest element in A[1..n]. Can you show that the algorithm is correct?

Select(A[1..n], s[1..m])

1. If (m==0) then return null;
2. Let $k = \lfloor (1 + m)/2 \rfloor$;
3. By linear select, we obtain x = the s[k]-th smallest element in A[1..n];
4. D[k] = x
5. D[1..k-1] = Select($A[1..n], s[1..k-1]$)
6. D[k+1..m] = Select ($A[1..n]$, s[k+1..m])
7. Return D[1..m];

Q3. (continued…)

(b) Can you give the time complexity of Select(A[1..n], s[1..m])?

(c) Can you propose an efficient algorithm DIVIDE(r, A[1..n]) that computes (x, $B[1..r-1]$, $C[1..n-r]$)?

a. x is the r-th smallest element in $A[1..n]$,
b. $B[1..r-1]$ is an array that contains all elements in A smaller than x, and
c. $C[1..n-r]$ is an array that contains all elements in A bigger than x.

What is the running time of DIVIDE(r, A[1..n])?

Q3. (continued…)

(d) Refer to part (c), if y is the q-th smallest element in A[1..n], can you answer the following queries?

i. If q>r, what is the rank of y in $C[1..n-r]$? (note: y is rank j if y is the j-th smallest element in $C[1..n-r]$.)

ii. If q<r, what is the rank of y in $B[1..r-1]$?

Q3. (continued…)

(e) Can you propose an $O(n \lg m)$-time algorithm for Select(A[1..n], S[1..m])?

Q4. (20 points) Randomized Algorithms

(a) On a rainy day in Singapore, n students enter a restaurant, each depositing an umbrella in a bin. The umbrellas get shuffled around, so that when the students come out, the receptionist hands out the umbrellas in random order. What is the expected number of students who get back their own umbrella?

Q4. (continued…)

(b) Xin Yi goes to a casino. She starts out with a net profit of 0 and plays a game with a sequence of rounds. In each round, her profit increases by 1 with probability 1/9 and decreases by 1 with probability 8/9. Each round is independent.
Give an upper-bound on the expected number of rounds in which her net profit is positive.

-- End of Paper ---