Xi'an Jiaotong-Liverpool University

西交利物浦大學

Paper CODE	EXAMINER	DEPARTMENT	TEL
CSE 204		Computer Science and Software Engineering	

2nd SEMESTER 2018/19 RESIT EXAMINATION

Undergraduate – Year 3

Complexity of Algorithms

TIME ALLOWED: 2 Hours

INSTRUCTIONS TO CANDIDATES

1. **This is a closed-book examination, which is to be written without books or notes.**
2. **Total marks available are 100. This accounts for 80% of the final mark.**
3. **The number in the column on the right indicates the marks for each question.**
4. **Answers should be written in the answer booklet(s) provided.**
5. **Only solutions in English are accepted.**
6. **All materials must be returned to the exam supervisor upon completion of the exam. Failure to do so will be deemed academic misconduct and will be dealt with accordingly.**

Notes:

- **To obtain full marks for each question, relevant and clear steps should be included in the answers.**
- **Partial marks may be awarded depending on the degree of completeness and clarity.**

Question 1: Algorithm Analysis [30 marks]

a) What is the asymptotic value of the expression $\sum_{i=1}^{n} \log_2 i$ as a function of n by using the big-Theta notation? [4 marks]

b) Give a tight bound of the runtime complexity class for each of the following two code fragments in Big-Oh notation, in terms of the variable N. Justify your answers.

i. [4 marks]

```
int sum = 0;
for (int i = 1; i <= N - 5; i++) {
   for (int j = 1; j <= N - 5; j = j * 2) {
     sum++;
   }
}
```

ii. [4 marks]

```
int sum = N;
for (int i = 0; i < 1000; i++) {
   for (int j = 1; j <= i; j++) {
      sum += N;}
   for (int j = 1; j <= i; j++) {
      sum += N;}
   for (int j = 1; j <= i; j++) {
      sum += N;}
}
```

c) The worst-case running time T(N) of Merge-Sort on an input sequence of size N can be characterized by the following recurrence equation, wherein a>0 and b>0 are constants:

$$T(N) = \begin{cases} a & \text{if } N = 1 \\ 2T(\frac{N}{2}) + bN & \text{if } N > 1 \end{cases}$$

i. Explain why the above recurrence equation can characterize the worst-case running time of Merge-Sort. [4 marks]

ii. Solve the above recurrence equation and express the time complexity of Merge-Sort using Big-Oh notation. [4 marks]

d) Given a binary search tree of height h, we wish to find out the value of its k^{th} element.

i. Assuming we have a function that computes the size of a tree in O(1), complete the following *findKth* function (using pseudo-code), which finds the k^{th} element with a time complexity in terms of the tree's height. [7 marks]

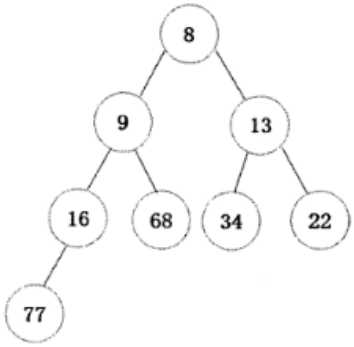
ii. Compute the time complexity of *findKth* function by using Big-Oh notation. [3 marks]

```
findKth(T; k)
Input: a binary search tree T and an integer k
Output: the kth element of T T.value.
1 sLeft=size(T.left);  // T.left denotes the left subtree of T.
2 _____
3 _____
...
```

Question 2: Tree and Graph [30 marks]

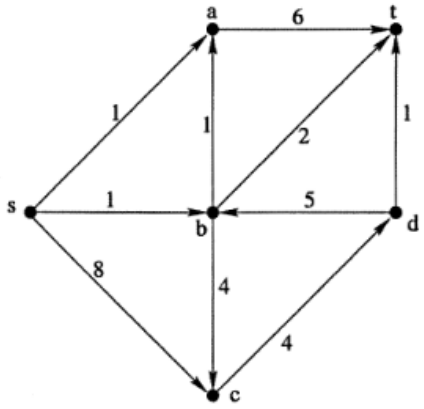
a) Consider a binary heap. Print the keys as encountered in an inorder travel. Is the output sorted? Justify your answer. [4 marks]

b) Delete two minimum numbers on the following min-heap. You do not need to show the array representation of the heap. You are only required to draw intermediate heaps and circle the final step. [8 marks]

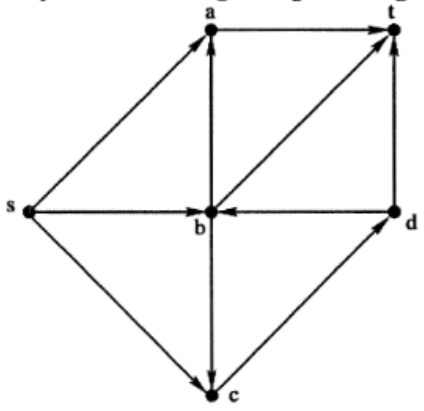


c) What is time complexity of the removal algorithm? [3 marks]

d) Consider the following directed graph. Each edge is labelled with the capacity of that edge. For instance, the edge (s, c) has capacity 8.



i. Find the maximum flow from s to t in this graph. Fill in the graph below with your flow: label each edge with the amount of flow you are sending along that edge. [4 marks]



ii. What is the value of your flow? [4 marks]

iii. Find the minimum-capacity cut between s and t in this graph. Show your answer by drawing a circle around the vertices in the above picture. [4 marks]

iv. What is the capacity of the cut you identified in part iii? [3 marks]

Question 3: Number Theory and Cryptography [15 marks]

a) Evaluate 27^{103} mod143. You may use the binary representation of 103. [3 marks]

b) Consider a cryptosystem wherein n =35 and e=5.

i. Verify that the pair (n, e) is a valid public key for an RSA cryptosystem. [3 marks]

ii. Calculate the associated private key d. [4 marks]

iii. Bob chooses an integer between 0 and 34, then encrypts it and sends the number 26 to Alice. Can you help Alice finding out the original integer chosen by Bob? Justify your answer. [5 marks]

Question 4: NP-Hardness [25 marks]

a) State the definitions of 3-SAT. [5 marks]

b) Show that the 3-SAT problem can be reduced to Vertex Cover problem.
Vertex Cover problem: Given a Graph G(V,E), decide if there is k vertex such that every edge is covered by one of them? [5 marks]

c) Deduce that Vertex Cover is NP-Complete. [5 marks]

d) If we could solve an NP-complete problem in polynomial time, would all other problems in NP necessarily be solvable in polynomial time? Briefly justify your answer. [5 marks]

e) If we could solve an NP-complete problem in time $O(n^{2019})$, would all other problems in NP necessarily be solvable in time $O(n^{2019})$? Briefly justify your answer. [5 marks]

END OF EXAM PAPER