The University of Nottingham Ningbo China

SCHOOL OF COMPUTER SCIENCE

A LEVEL 2 MODULE, FULL YEAR, 2017–2018

ALGORITHMS CORRECTNESS AND EFFICIENCY

Time allowed TWO hours

Candidates may complete the front covers of their answer books and sign their desk cards but must NOT write anything else until the start of the examination period is announced.

Answer all FOUR questions. The total mark is 100.

No calculators are permitted in this examination.

Dictionaries are not allowed with one exception. Those whose first language is not English may use a standard translation dictionary to translate between that language and English provided that neither language is the subject of this examination. Subject-specific translation directories are not permitted.

No electronic devices capable of storing and retrieving text, including electronic dictionaries, may be used.

DO NOT turn examination paper over until instructed to do so

Question 1 Multiple choice questions. For each question, there are one or more correct answers; if *all* correct answers are selected and no incorrect ones are selected, then two marks will be awarded; otherwise, no mark will be awarded. (22 marks)

(a) Choose the class(es) to which the function $S(n) = (2n^2+3n^3)(\log(n^2)+(\log n)^2)$ belongs to.

(i) $\Omega(\log n)$
(ii) $\Theta(n^2 \log n)$
(iii) $O(n^2)$
(iv) $\Theta(n^3(\log n)^2)$
(v) $O(n^3)$

(b) Suppose that an intermixed sequence of (stack) push and pop operations are performed. The push operations push the integers 0 through 5 in order; the pop operations print out the returned value. For the returned values, which of the following sequence(s) cannot occur?

(i) 4 3 2 1 0 5
(ii) 4 5 3 2 0 1
(iii) 1 2 3 4 5 0
(iv) 1 4 5 3 0 2

(c) Let Q be a queue. It is initially empty. A total of 32 *enqueue* operations and 15 *dequeue* operations are performed on Q. The *enqueue* and *dequeue* operations are interleaved. 5 of these *dequeue* operations return null to indicate an empty queue. Select the size of Q after performing all these 47 operations.

(i) 17
(ii) 22
(iii) 27
(iv) 32
(v) None of the above

(d) Which of the following code fragments has $result = n^i$ as a loop invariant? Select all the correct answer(s).

(i)
```
int power(int n, int k){
      int result = 1;
      for (int i = 1; i < k; i++){
            result = result * n;
      }
}
```

(ii)
```
int power(int n, int k){
      int result = n;
      for (int i = 1; i < k; i++){
            result = result * n;
      }
}
```

(iii)
```
int power(int n, int k){
      int result = n;
      for (int i = 0; i < k; i++){
            result = result * n;
      }
}
```

(e) Suppose that you run the code fragment below (generate and then Mergesort an array of random double values) for $N = 10,000,000$ and observe that it takes 5.3 seconds.

```
double[] a = new double[N];
for (int i = 0; i< N; i++){
a[i] = Math.random();
}
MergeSort(a);
```

Assuming you have enough memory, which one of the following is a reasonable predication of its running time (in seconds) for $N = 1,000,000,000$? Pick just one answer.

(i) 340 seconds

(ii) 530 seconds

(iii) 680 seconds

(iv) 1060 seconds

(v) 5300 seconds

(f) Which of the following data structures have reliably efficient (logarithmic) performance for search, insertion and deletion? Select all the correct answer(s).

(i) AVL tree

(ii) ordered array

(iii) red-black tree

(iv) unordered list

(g) The time complexity of depth-first traversal of a graph using a stack is best represented by which one of the following:

(i) $O(1)$

(ii) $O(|E|)$ where $|E|$ is the number of edges

(iii) $O(|V|)$ where $|V|$ is the number of vertices

(iv) $O(|V| + |E|)$

(v) $O(2^{|V|})$

(h) Suppose that the following keys are inserted in *some order* into an initially empty linear probing hash table of size 7, using the following table of hash values:

key	hash value
A	3
B	1
C	4
D	1
E	5
F	2
G	5

Which one of the following could be the contents of the resulting linear-probing array? Pick just one answer.

(i)

0	1	2	3	4	5	6
G	B	D	F	A	C	E

(ii)

0	1	2	3	4	5	6
B	G	D	F	A	C	E

(iii)

0	1	2	3	4	5	6
E	G	F	A	B	C	D

(i) This question is about Boyer-Moore algorithm for pattern matching. Given the character set S = {a, b, c, d, e, f, g, r, s, t}, which one of the following is the last-occurrence function for the pattern "abstract"?

(i)

c	a	b	c	d	e	f	g	r	s	t
L(c)	6	2	7	-1	-1	-1	-1	5	3	8

(ii)

c	a	b	c	d	e	f	g	r	s	t
L(c)	0	1	6	-1	-1	-1	-1	4	2	7

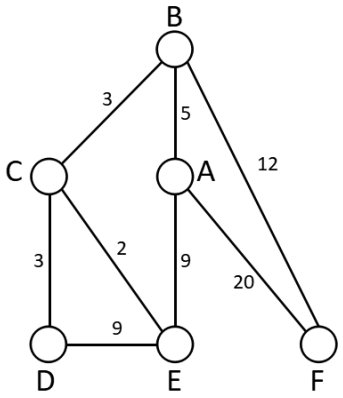
(iii)

c	a	b	c	d	e	f	g	r	s	t
L(c)	1	2	7	-1	-1	-1	-1	5	3	8

(iv)

c	a	b	c	d	e	f	g	r	s	t
L(c)	5	1	6	-1	-1	-1	-1	4	2	7

(j) Given the weighted graph shown below, please find the sum of weights of all the edges in its minimum spanning tree.



(i) 20

(ii) 22

(iii) 25

(iv) 31

(v) 33

(k) This question is about KMP algorithm for pattern matching. Which one of the following is the fail function for the pattern "amalgamation"?

(i)

c	a	m	a	l	g	a	m	a	t	i	o	n
f(c)	-1	-1	0	-1	0	1	2	-1	-1	-1	-1	-1

(ii)

c	a	m	a	l	g	a	m	a	t	i	o	n
f(c)	0	0	1	0	0	1	2	3	0	0	0	0

(iii)

c	a	m	a	l	g	a	m	a	t	i	o	n
f(c)	-1	0	1	0	0	2	1	3	0	0	0	0

(iv)

c	a	m	a	l	g	a	m	a	t	i	o	n
f(c)	0	0	1	0	0	2	1	3	0	0	0	0

Question 2 This question is about algorithm correctness and recursion. (26 marks)

(a) Explain the meaning of 'partial correctness' and 'total correctness' of algorithms. Give an example to illustrate the difference between 'partial correctness' and 'total correctness'. (4 marks)

(b) The following program implements a sorting algorithm:

```
1   public static void sort(int[] arr) {
2        int len = arr.length;
3        for(int i = len - 1; i > 0; i--){
4             for(int j = 0; j < i; j++){
5                  if( arr[j] > arr[j+1]){
6                       int temp = arr[j];
7                       arr[j] = arr[j+1];
8                       arr[j+1] = temp;
9                  }//end if
10            } // end inner for loop
11       }//end outer for loop
12  }
```

(i) The partial correctness of the program above can be proved by proving the loop invariant of the inner for loop and the loop invariant of the outer for loop. What is the loop invariant of the inner for loop? What is the loop invariant of the outer for loop? You may write your answer using either logical expressions or as pseudo-code. (5 marks)

(ii) Prove the loop invariant of the inner for loop by mathematical induction. (8 marks)

(iii) Does the program above satisfy the definition of 'total correctness'? Justify your answer briefly.

(3 marks)

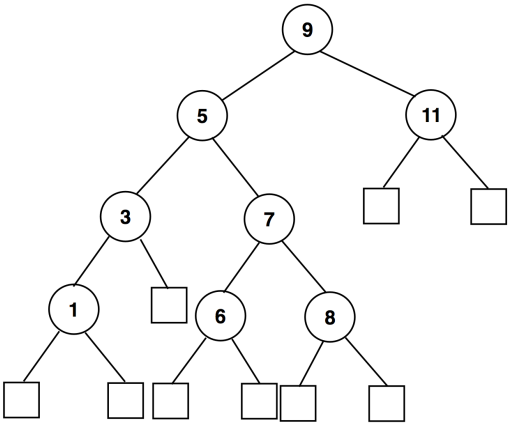
(c) Describe a method in pseudo-code to search for and locate a target value in a sorted array of integers. The method should run in $O(\log n)$ time.

(6 marks)

Question 3 This question is about search tree structures. (26 marks)

(a) What is a binary search tree? (2 marks)

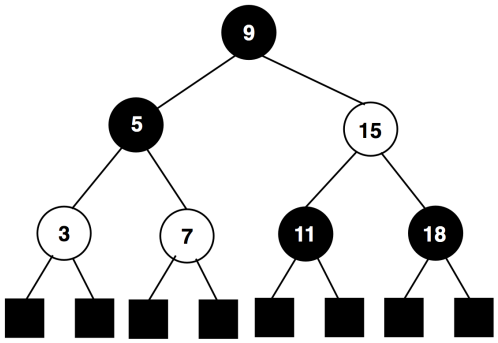
(b) Explain and draw figures to show the process of removing the key 5 from the following binary search tree. (4 marks)



(c) To make the binary search tree shown above more balanced, we could use the 'trinode restructuring' operation. Draw figures to show the resulting tree after applying the 'trinode restructuring' operation *once*. What is the worst-case time complexity of the 'trinode restructuring'? (3 marks)

(d) What is a red-black tree? (4 marks)

(e) Explain and draw figures to show the process of inserting the key 6 into the following red-black tree $\mathcal{T}$. (6 marks)



(f) Explain and draw figures to show the process of deleting the key 15 from the *original* red-black tree $\mathcal{T}$. (7 marks)

Question 4 This question is about graphs. (26 marks)
Consider the graph $\mathcal{G}$ given by the following adjacency lists:

```
A ---> {B, D, E}
B ---> {C, E}
C ---> {F}
D ---> {E}
E ---> {F}
F ---> {}
```

(a) Draw the graph according to the adjacency lists shown above.
Then trace a breadth-first traversal of the graph starting from A using a queue. At each step, show which nodes are in the queue.
List the nodes in the order that they are marked as 'visited'.
(8 marks)

(b) Trace a depth-first traversal of the graph starting from A using a stack.
At each step, show which nodes are in the stack.
List the nodes in the order that they are marked as 'visited'.
(6 marks)

(c) Add one edge to the graph $\mathcal{G}$, from F to E. Show how to detect a cycle using topological sort. At each step, display the state of the output array of nodes and the nodes and edges which remain in the graph.
(4 marks)

(d) Explain how to modify the depth first search (DFS) to detect cycles in a graph. Add one edge to the *original* graph $\mathcal{G}$, from E to B. Show the process of using the modified DFS to detect a cycle in the new graph. (8 marks)