Index Number:							

UNIVERSITY OF MORATUWA

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

BSc Engineering Honours Degree
2013 Intake Semester 2 Examination

CS2022: DATA STRUCTURES & ALGORITHMS (MODEL PAPER)

Time allowed: 2 Hours March 2015

ADDITIONAL MATERIAL: *None*

INSTRUCTIONS TO CANDIDATES:

1. This paper consists of **two (2) Sections** in **14** pages.
2. Answer **all** questions in the question paper itself **in the space provided**. If you make a mistake or need additional space you may attach additional sheets.
3. Section **A** contains **twenty-five (25) MCQ and/or short answer questions**. **Answer these questions in the space provided in the paper**.
4. **For each MCQ question** in section A, there is **only one correct answer** and you get full marks if you mark **only the correct choice**. **Otherwise** you will get **zero** marks (**no negative marks**).
5. Each question in section A carries 2 marks.
6. Section B contains **two (2)** essay type questions.
7. The maximum attainable mark for each question in section B is given in brackets.
8. The maximum attainable mark for this paper is 100. This examination accounts for 60% of the module assessment.
9. This is a closed book examination.

   NB: It is an offence to be in possession of unauthorized material during the examination.

10. Only calculators approved and labelled by the Faculty of Engineering are permitted.
11. Assume reasonable values for any data not given in or with the examination paper. Clearly state such assumptions made on the script.
12. In case of any doubt as to the interpretation of the wording of a question, make suitable assumptions and clearly state them on the script.

This paper should be answered only in English.

SECTION A

1. Which of the following factors affect the running time of a program?

   I. Amount of processing required by the algorithm
   II. Size of the input dataset
   III. Nature of the input dataset

   a) I Only
   b) I and II only
   c) I and III only
   d) All three

2. Which of the following statements are true regarding evaluation of algorithms?

   I. Best-case running time of an algorithm is a very good representation of the efficiency of an algorithm.
   II. Space efficiency is an important consideration in evaluating algorithms.
   III. If two algorithms have same worst case running time, their average case running time must be same.

   a) I Only
   b) II only
   c) I and III only
   d) II and III only

3. Following array is to be sorted using the Bubble sort.

55	66	11	88	44	22	33	77

   The array after first and second iterations of the Bubble sort are shown below.

   After First Iteration:

55	11	66	44	22	33	77	88

   After Second Iteration:

11	55	44	22	33	66	77	88

   What will be the array after the **forth** Iteration of the Bubble sort?

4. A sort which compares adjacent elements in a list and switches where necessary is;

   a) Insertion sort
   b) Heap sort
   c) Quick sort
   d) Bubble sort

5. What are the worst case running times of insertion sort, merge sort and heap sort respectively?

   a) $O(n^2)$, $O(n\log n)$, $O(n)$
   b) $O(n^2)$, $O(n\log n)$, $O(n\log n)$
   c) $O(n\log n)$, $O(n\log n)$, $O(n\log n)$
   d) None of the above

Continued …

6. If $f(x) = 27x^3 + 3x + 1$ and $g(x) = 2x^2 + 2$ and $h(x) = 3x^3 + 2x^2 + 1$ which of the following statements are correct?

   I. $f(x) \in \Omega(g(x))$
   II. $f(x) \in O(h(x))$
   III. $f(x) \in \theta(g(x))$

   a) I Only
   b) I and II only
   c) I and III only
   d) All three

7. Which of the following statements are correct regarding asymptotic notation?

   I. If $f(x) \in \Theta(g(x))$, then $f(x)$ cannot be in the set $\omega(g(x))$
   II. If $f(x) \in o(g(x))$ and $g(x) \in O(h(x))$ then, $f(x) \in o(g(x))$ is **always** true
   III. If $f(x) \in \omega(g(x))$, then $g(x) \in \Theta(f(x))$ can **never** be true

   a) I Only
   b) I and II only
   c) I and III only
   d) All three

8. What is a recursive algorithm?

   ……

   ……

   ……

   ……

9. Which of the following statement is true regarding the data structures?

   I. Push and pop are operations of the stacks.
   II. Queue is a Last-In-First-Out data structure.
   III. A tree is an acyclic graph.

   a) I Only
   b) I and II only
   c) I and III only
   d) All three

Continued …

10. Complete the following pseudocode which implements the post-order traversal of a binary tree.

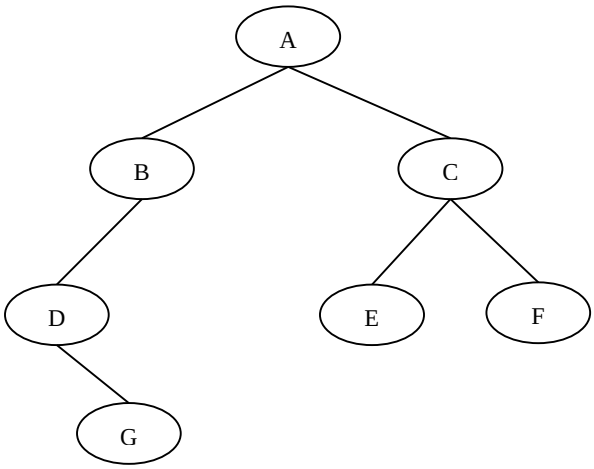
```
Void post-order-traversal (Tree t) {
    if ( t is null)
        return;

    …….
    post-order-traversal (t.root.left)

    …….

}
```

11. Which of the following statements are correct regarding tree implementations?

I. If an array is used to implement binary trees, it will be inefficient of the tree is unbalanced.
II. Binary trees can be implemented using parent, left child and right-sibling representation.
III. Trees with unbounded branching can be implemented using arrays.

a) I Only
b) I and II only
c) I and III only
d) All three

12. Consider the binary tree shown below.



What is the result of post-order traversal of this tree?

……

Continued …

13. Draw the binary tree obtained by inserting (in the same order) 55, 22, 66, and 44 to a binary tree which contains 33.

14. Which of the following is not a limitation of binary search algorithm?

    a) Must use a sorted array
    b) Requirement of sorted array is expensive when a lot of insertion and deletions are needed
    c) There must be a mechanism to access middle element directly
    d) Binary search algorithm is not efficient when the data elements are more than 1000

15. Which of the following is not a collision handling technique in hash tables?

    a) Chaining
    b) Linear probing
    c) Double hashing
    d) All of the above are collision handling techniques

16. Which of the following statements are correct regarding graph searching?

    I. A depth first search (DFS) must contain all the vertices in the graph in its results.
    II. DFS is generally implemented using a stack.
    III. Breadth first search requires backtracking.

    a) Only statements (i) and (ii) are true.
    b) Only statements (ii) and (iii) are true.
    c) Only statements (i) and (iii) are true.
    d) All three statements are true.

Continued …

Use the following description to answer questions 17, 18 and 19.

You are provided with an implementation of a (single) linked list which supports the following operations.

- `LIST-SEARCH(L,k)`: Finds the first node with the key *k* in the list *L*. Returns NIL if no element is found.
- `LIST-HEAD(L)`: Returns the value of the first node in the list.
- `LIST-INSERT(L,x)`: Inserts the value *x* as the first element of the list *L*.
- `LIST-DELETE(L,x)`: Deletes the first node which contains value *x* from the list *L* if it exists.
- `LIST-INSERTAT(L,x,i)`: Inserts the value *x* into the i^{th} location of the list *L*.
- `LIST-DELETEAT(L,i)`: Deletes the node in the i^{th} location of the list *L*.
- `LIST-GET(L,i)`: Returns the value of the node in the i^{th} location of the list *L*.

You are required to implement a queue using the above linked list implementation and you are not allowed to make any changes to the linked list implementation. For this implementation, we are using a record (structure) of the following format.

```
typedef struct {
     linked_list L;
     int size; //number of elements in the queue
} my_queue;
```

The pseudocode for the initialization of queue is shown below.

```
     void init_queue(my_queue Q){
          Q.L = null;
          Q.size = 0;
     }
```

17. The enqueue operation (`void enqueue(my_queue Q, int val)`) which will adds the specified value to the queue needs to be implemented. Complete the pseudocode for enqueue operation shown below.

```
     void enqueue(my_queue Q, int val){
          LIST-INSERT(Q.L, val)

          ..........
     }
```

Continued …

18. The dequeue operation (`int dequeue(my_queue Q)`) which will removes an element from the queue needs to be implemented. Complete the pseudocode for dequeue operation shown below.

```
int dequeue(my_queue Q){
      retval = ……
      Q.size -= 1;
      return retval;
}
```

19. A length operation (`int queue_length(my_queue Q)`), which returns the number of items in the queue needs to be implemented. Complete the pseudocode for length operation shown below.

```
int queue_length (my_queue Q){

      ……
}
```

20. What is the main difference between the divide and conquer approach and dynamic programming approach?

……

……

……

21. What is a max-heap?

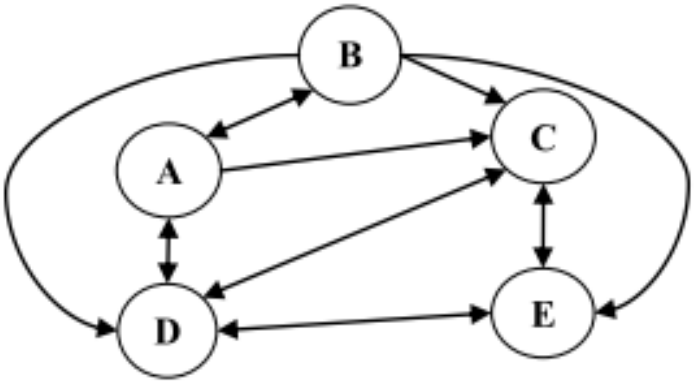
……

……

……

Continued …

Use the following graph to answer questions 22, 23 and 24.



22. Show the adjacency list representation of the above graph.

23. Which of the following is **not** a possible depth first search sequence of the above graph?

a) A, D, C, B, E
b) A, B, C, D, E
c) A, C, D, E, B
d) A, D, E, C, B

24. Write the breadth first search result of the above graph if the search starts from vertex A.

……

25. Which of the following statements are correct regarding graphs?

I. A graph is a collection of nodes (vertices) which may be connected in pairs by line segments called edges.
II. A tree is a hierarchical graph which may contain cycles.
III. Nodes that do not have child nodes are called leaf nodes in trees.

a) Only statements (i) and (ii) are true.
b) Only statements (ii) and (iii) are true.
c) Only statements (i) and (iii) are true.
d) All three statements are true.

Continued …

SECTION B

(Note: These questions are taken from the past exam papers just to show the structure. Actual questions will be based on the things discussed in the class and the ab exercises)

Q1. **[25 *marks*]**

a) Compare Quicksort, Merge sort and Heap sort considering their worst case time complexity and space complexity.

[3 *marks*]

	Quicksort	**Merge sort**	**Heap sort**
Time complexity			
Space complexity			

Let $A[1 \sim n]$ be an array of n distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair (i, j) is called an ***inversion*** of A. (*Note: Pseudocodes for the Insertion sort, Merge Sort and Bubble sort algorithms are given at the end of the paper as resources.*)

b) List four (4) inversions of the array [5, 2, 8, 6, 1]. [2 *marks*]

……..……..……

c)

i. What array with elements from the set $\{1, 2, \ldots, n\}$ has the most inversions? [1 *mark*]

……..……..

ii. How many does it have? [1 *mark*]

……..……..

iii. Explain your answers. [2 *marks*]

……..……..

……..……..

Continued …

……..……..

d) If the **actual number of inversions in the input array** is m, show that the running time of the Insertion sort will be Θ(*m*). [3 *marks*]

……..……..

……..……..

……..……..

……..……..

……..……..

e) What is the relationship between the **number of swaps that will occur in Bubble sort and the number of inversions in the input array**? [2 *marks*]

……..……..

f) Explain your answer for question e. [2 *marks*]

……..……..

g)

i. Give an algorithm that determines the number of inversions in any permutation on *n* elements in Θ(*n* lg *n*) worst-case time. (*Hint:* Modify Merge sort.) [6 *marks*]

……..……

……..……

……..……

……..……

……..……

……..……

……..……

Continued …

……

……

……

……

……

……

……

ii. Show that the worst case running time of the algorithm you proposed in part (e) is Θ(n lg n). [3 *marks*]

……

……

……

……

……

……

……

……

……

……

……

……

Continued …

Q2. **[25 *marks*]**

a)

i. Explain why dynamic programming solutions would run faster compared to divide and conquer or recursive solutions. [4 *marks*]

……..……

……..……

……..……

……..……

……..……

……..……

……..……

……..……

ii. Explain why greedy solutions would run faster compared to dynamic programming solutions. [4 *marks*]

……..……

……..……

……..……

……..……

……..……

……..……

……..……

Continued …

b) Rod cutting problem is defined as given a rod of length n meters and a table of prices p_i for i = 1, 2, …….., n (that represent the prices of pieces of length i for different i values), determine the maximum revenue r_n obtainable by cutting up the rod and selling the pieces.

i. Argue that the problem can be solved using the dynamic programming approach. [2 *marks*]

……..……

……..……

……..……

……..……

ii. Derive the recursive solution to the problem of selecting the optimal set of pieces that will maximize the revenue for a rod of given length. [3 *marks*]

……..……

……..……

……..……

……..……

……..……

iii. Given the following table with prices p_i for i = 2, 3, …….., 8, determine the maximum revenue obtainable by cutting up a 11 meter long rod using the dynamic programming approach. Clearly show the steps you followed. [6 *marks*]

length (i)	2	3	4	5	6	7	8
price (p_i)	5	8	9	10	17	17	20

Continued …

c) Consider a modified version of 1-0 knapsack problem in which the thief is allowed to **pick a part of only a single item.**

i. If **the item of which a part can be picked is fixed**, can the optimal solution be obtained using the greedy approach? [1 *mark*]

……..……

ii. Explain your answer in part (i) [2 *marks*].

……..……

……..……

……..……

iii. If the **thief is given the flexibility to pick which item she wants to pick partially**, can the optimal solution be obtained using the greedy approach? [1 *mark*]

……..……

iv. Explain your answer in part (iii) [2 *marks*].

……..……

……..……

……..……

--- End of the Paper ---