



FINAL
Examination Paper

(COVER PAGE)

Session : April 2014

Programme : Diploma In Information And Communication Technology (DICTN)

Course : ICT2102: Introduction To Data Structure

Date of Examination : July 21, 2014

Time : 2:00pm – 4:00pm Reading Time: Nil

Duration : 2 Hours

Special Instructions :

Answer any **FOUR (4)** questions.

Materials permitted : Standard Calculator

Materials provided : Nil

Examiner (s) : Mr. Ang Chee Huei, Koo Lee Chun.

Moderator : Ms. Siti Hawa Mohamed Said

This paper consists of 5 printed pages, including the cover page.

INTI INTERNATIONAL COLLEGE SUBANG

DIPLOMA IN INFORMATION AND COMMUNICATIONS TECHNOLOGY
PROGRAMME (DICTN)ICT2102: INTRODUCTION TO DATA STRUCTURE
FINAL EXAMINATION: APRIL 2014 SESSION

Instructions: This paper consists of **SIX (6)** questions. Answer any **FOUR (4)** questions in the answer booklet provided. All questions carry equal marks.

QUESTION 1

(a) Fill in the blanks :

- (i) _____ means that a function can be used without taking into account how the function is implemented.
- (ii) A variable that is used to store the address of another variable is called a _____.
- (iii) When a linked-list based stack is empty, the value of the top pointer is _____.
- (iv) Which data structure has the “First in, First out” feature? Answer: _____.
- (v) A _____ is a constructor without arguments.
- (vi) Traversing the nodes of a tree in _____ visits the parent after the children.
- (vii) In a _____ the value stored in any parent node is never greater than the value stored in any right child of the parent.
- (viii) _____ refer to new types of data, defined by the programmer, where the implementation details are hidden from the user, and access to the data is provided through an interface to a specific set of operations.

(8 marks)

(b) Using diagrams, show the detailed workings of an insertion sort in ascending order on the following array.

5	3	1	9	8	2	4	7
---	---	---	---	---	---	---	---

(7 marks)

(c) Write in pseudocode or otherwise, the algorithm for bubble sort. How does it perform when an array becomes sorted at an early stage?

(10 marks)

QUESTION 2

- (a) Show the results of the following sequence of events, by drawing the state of the data structure:

add(4), add(8), add(1), add(6), remove(), remove()

where add and remove are the operations that correspond to the basic operations in :

- (i) stack

(6 marks)

- (ii) queue

(6 marks)

- (b) Consider a linked list :

- (i) Describe an algorithm in plain English, to remove the last item from the list.

(4 marks)

- (ii) Write a function `removeLast()` that implements your algorithm in (i). You can use without need to define all the attributes and functions that are members of the `LinkedList` class discussed in lecture.

(9 marks)

QUESTION 3

- (a) What is a binary search tree? Outline the advantage of it.

(5 marks)

- (b) Show how binary search works when searching for 13 in the following array:

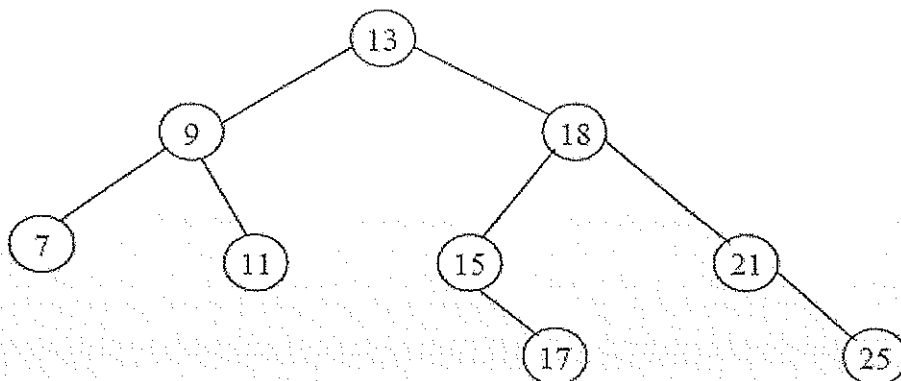
2	3	5	8	10	12	13	15	18	20	23
---	---	---	---	----	----	----	----	----	----	----

(6 marks)

- (c) Write an algorithm for binary search on a sorted array in pseudocode or C++.

(8 marks)

- (d) Show the effect of inserting key 14 and removing key 18 into the following binary search tree.



(6 marks)

QUESTION 4

- (a) Provide a linked list declaration of a Stack in C++, showing the data structures involved and the class interface. You can leave out the actual implementation of the Stack functions. (7 marks)
- (b) With the aid of pointer diagrams, show how the *pop()* operation works on a Stack which has a linked list implementation. Treat the cases of a stack with one value and a stack with more than one value. Is there any difference? (6 marks)
- (c) Give the C++ implementation of the method *pop()*. (6 marks)
- (d) Describe the alternative Stack implementation using an array. (6 marks)

QUESTION 5

- (a) Consider the C++ code fragment below. Write down the output.

```
char str[] = "This is a Test";
int *p;
char *c;
int index = 5;
c = str;
cout << c << endl;
p = &index;
cout << *p << endl;
c = c + *p;
cout << *c << endl;
```

(6 marks)

- (b) The *Node* class below can be used to implement a linked list.

```
class Node {
private:
    int data;
    Node *next;
public:
    Node();
    Node(int x, Node *ptr);
    int getData(); //return the data
    Node *getNext(); //return next Node's pointer
    void setData(int x); //set data of the Node to x
    void setNext(Node *ptr); //set next pointer to Node ptr
};
```

- (i) Write the implementation of the default constructor that will create a *Node* object with the value of *data* set to 0 and the *next* pointer point to nothing.

(4 marks)

(ii) Write the implementation of the constructor `Node(int x, Node *ptr);`
(3 marks)

(iii) Write the implementation of all other member functions :
`int getData();`
`Node *getNext();`
`void setData(int x);`
`void setNext(Node *ptr);`

(12 marks)

QUESTION 6

(a) Define the *queue* data structure and give a real-life example of *queue*.

(5 marks)

(b) The class below is an ADT for a static array based queue:

```
const int MAX=100;
class Queue{
public:
    Queue(length=0; head=0; tail=0);
    void enqueue (char x);
    char dequeue();
    int getHead();
    int getTail();
    int getSize();
    bool isEmpty();
private:
    char qArray[MAX]
    int length;
    int head;
    int tail;
};
```

(i) Provide the full implementation in C++ of the member function `enqueue (char x)`, which insert the data x into the queue. Note that you cannot insert the data if the queue is full.

(8 marks)

(ii) Provide the full implementation in C++ of the member function `dequeue ()`, which remove a data x from the queue. Note that you cannot remove the data if the queue is empty.

(5 marks)

(c) Write the selection sort algorithm for an array *number* of size n .

(7 marks)