UNIVERSITI
TEKNOLOGI
PETRONAS

FINAL EXAMINATION
MAY 2012 SEMESTER

COURSE	:	**TCB1063 ALGORITHM AND DATA STRUCTURE**
DATE	:	**30th AUGUST 2012 (THURSDAY)**
TIME	:	**9.00 AM – 12.00 NOON (3 HOURS)**

INSTRUCTIONS TO CANDIDATES

1. Answer **ALL** questions from the Questions Booklet.
2. Begin **EACH** answer on a new page in the Answer Booklet.
3. Indicate clearly answers that are cancelled, if any.
4. Where applicable, show clearly steps taken in arriving at the solutions and indicate **ALL** assumptions.
5. Do not open this Question Booklet until instructed.

Note : There are **EIGHT (8)** pages in this Question Booklet including the cover page.

1. a. Write a C++ program to store the string "This is a sample" into an array named samtest and displays the values in samtest using for loop that uses a pointer access to each element in the array.

[6 marks]

b. Write a C++ program that asks the user to enter a sequence of strings and stores them in a vector. Call a function named reverse that reverses the elements in that vector. The function should receive a reference parameter and should not return any value.

Sample of output:

```
Enter character(s)(q to stop): For Someone Fantastic q
The reverse of the vector:Fantastic Someone For
Press any key to continue . . .
```

[8 marks]

c. What is code tuning?

[2 marks]

d. How do you minimize the work performed inside the loop below?

```
num = 1;
while(num <= total/2)
{
      emSal[num] = 1.5 * emSal[num];
      num = num + 1;
}
```

[2 marks]

e. How do you unroll the loop so that the number of iteration can be reduced?

```
num = 1;
while(num < n)
{
      a[num] = num;
      num = num + 1;
}
```

[2 marks]

2. a. Assume the content of an array named `myArray` is,

`20 30 10 5 50 70 65 3 10`

Perform each of the following sorting by indicating the content of `myArray` for each pass:

i. Selection sort

[5 marks]

ii. Bubble sort

[5 marks]

iii. Insertion sort

[5 marks]

b. Consider the following list:

`3 11 18 46 50 56 69 86 93 100 119`

Perform binary search algorithm, to find `119`. Show the values of first, last and middle for each iteration of the loop.

[5 marks]

3. a. Trace the evaluation of the following expression using postfix notation. Show the operands stack and each time operators stack is executed.

```
8 2 * 10 / 2 1 24 * + -
```

[6 marks]

b. Consider the following statements. Trace and give the output of the following segment of code.

```
stackType myStack;
int m, n;

myStack.initializeStack();
m = 87;
n = 2;

myStack.addStack (m);
myStack.addStack (n);
m = myStack.front();
myStack.deleteStack ();
myStack.addStack (m+33);
myStack.addStack (11);
myStack.addStack (m);
myStack.addStack (n-2);
cout<<"Stack Elements: ";
while(!myStack.isEmptyStack ()){
     cout<<" "<<myStack.front();
     myStack.deleteStack ();
}
cout<<endl;
```

[6 marks]

c. Write a function that reads a line and reverses the words in the line (not the characters) using a stack. For example, given the following input:

to? up you are What you! Hey

You should get the following output:

Hey you! What are you up to?

[8 marks]

4. a. Assume that `my_queue` is an instance of a class that implements `queue<string>` and `my_queue` is an empty queue. Explain the effect of each line of the following code.

```
1.  my_queue.push("Hello");
2.  my_queue.push("Bye");
3.  cout << my_queue.front();
4.  my_queue.pop();
5.  my_queue.push("Welcome");
6.  if (!my_queue.empty()) {
7.      cout << my_queue.front();
8.      my_queue.pop();
9.      cout << "Size"<< my_queue.size()<< endl;
10.     cout<<"Front item"<< my_queue.front()<<;}
```

[6 marks]

b. Write a new queue function called `move_to_rear` that moves the element currently at the front of the queue to the rear of the queue. The element that was second in line will be the new front element. Do this using the `push`, `front`, and `pop` functions.

[6 marks]

c. **FIGURE Q4c** shows a linked list with `list`, as the head pointer, `X` and `Y` as the pointers pointing to the respective node. Each node consists of `info` of type integer and a `link`, to point to the next address of the node.

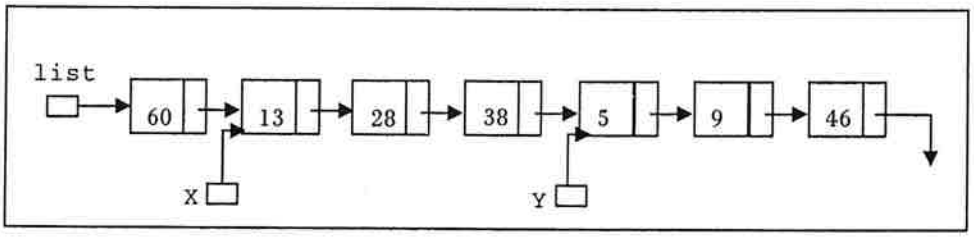

FIGURE Q4c

Determine the output of the following statements:

i. `cout << list->info;`

[1 mark]

ii. `cout << X->link->info;`

[1 mark]

iii. `cout << Y->info;`

[1 mark]

iv. `cout << list->link->link->info;`

[1 mark]

d. Based on **FIGURE Q4c**, write the code statement for each of the following:

i. Make `X` point to the node containing `info 38`.

[2 marks]

ii. Make `Y` point to the last node in the list.

[2 marks]

5. a. i. Draw **ALL** of the binary trees with 4 nodes.

[6 marks]

ii. How many different binary trees are there with six nodes?

[2 marks]

b. Consider the Adeĺson-Veĺskiĭ-Landis (AVL) tree in **Q5b**:

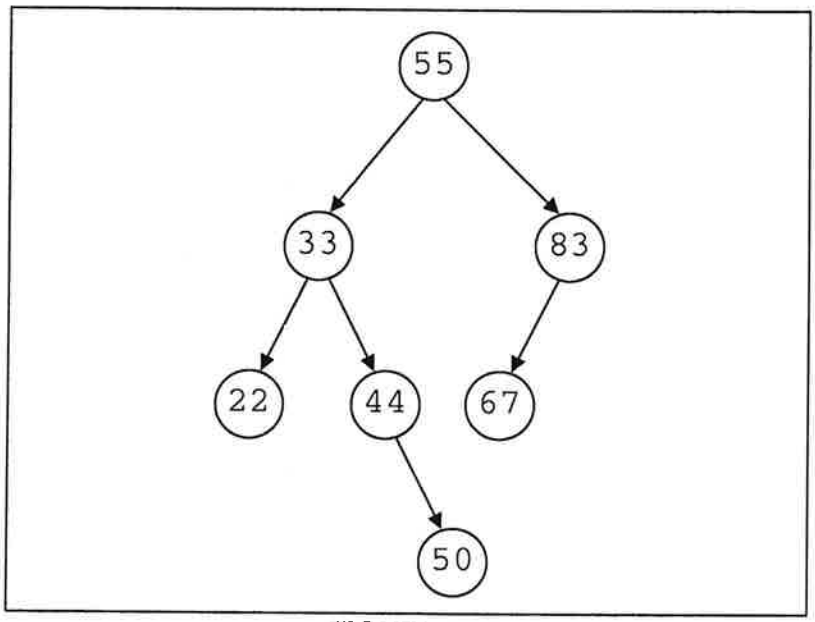

FIGURE Q5b

i. Insert `47` in the AVL tree and draw the resulting tree.

[2 marks]

ii. What is the balance factor at the root node after the insertion?

[2 marks]

c. Build an AVL tree to insert the following numbers `33, 44, 11, 22, 99, 88, 77, 98, 13, 72` in the given order.

[8 marks]

- END OF PAPER -