

PROGRAMMING QUESTIONS

QUESTION ONE

- b) Given the `TreeNode`, `BSTEmployee` and `Employee` ADTs

```
public class Employee
{
    private String ID;
    private String name;
    private String status;
    private String position;
    private double salary;

    public Employee () {}
    public Employee(String a,String b,String c,String d,
    double e)

    private String getID (){};
    private String getName (){};
    private String getStatus (){};
    private String getPosition (){};
    private double getSalary (){};
}

public class TreeNode
{
    TreeNode left;
    Employee data;
    TreeNode right;
    //definition for other methods
}

public class BSTEmployee
{
    TreeNode root;
    public BSTEmployee();
    public double calcTotSalary (){};
    public void calcAveSalary(){};
    public void countMStatus(String status) {};
    //definition for all methods
}
```

The following table contains the information of TechNoArt Sdn. Bhd employees

ID	Name	Position	Marital Status	Salary (RM)
13124	Nik Mond Fathi Nik Mai	Marketing Officer	Single	4494
27421	NoorShuhada Ibrahim	Operating Officer	Married	8627
21584	Anumugam Senthivel	Clerk	Married	2090
16145	Shi Nuraid Abu Bakar	Manager	Single	9060
26121	Ahmad Huzni Zamri	Financial Officer	Single	5964
24737	Ahmad Shafiq Abdul Razak	Clerk	Married	1629

- Based on the information in the table, draw a Binary Search Tree (BST) according to employee salary (3 marks)
- Write the definition of method `countMStatus (String status)` and its recursive method to count and display the number of employees with the marital status given by the parameter. (6 marks)
- Write the definition of method `calcAveSalary()` and necessary recursive method(s) to calculate and display the average salary of all employees. (7 marks)



## QUESTION TWO

Given the definition of SpaQaseh, TreeNode and BSTSpaQaseh ADTs :

TreeNode ADT	BSTSpaQaseh ADT
<pre>TreeNode left SpaQaseh data TreeNode right  Public TreeNode(Object)</pre>	<pre>TreeNode root  public BSTSpaQaseh {} public void displaySpaQaseh() public void calcPrice(String) public void countTreatment(String)</pre>

SpaQaseh ADT
<pre>String treatmentName double price String room int hour  public SpaQaseh(String, double, String, int)</pre>

Based on the following information in the table shows data of SpaQaseh

Treatment Name	Room	Price (RM)	Hour
Facial	Rose	100	2
Pedicure and Manicure	Vanilla	90	1
Full Body Massage	Tulip	250	2.5
Leg Massage	Daisy	150	1.5
Wedding Package	Carnation	350	4
Hair Spa	Hydrangea	150	1.5
Hot Stone Therapy	Lily	250	2.5
Head Massage	Orchid	100	1.5

- Differentiate between a Binary Tree and a Binary Search Tree (3 marks)
- Write a definition for method countTreatment() and its recursive method to calculate the number of treatment based on the hour which is takes up more than 2 hours (5 marks)
- Write method definition for displaySpaQaseh() and its recursive method to display all treatments where the price for each treatment is more than RM 100 and hours of treatment is less than 3 (5marks)
- Write method definition for calcPnce() and its recursive method to calculate the total price based on the treatments the customer takes (5 marks)

### QUESTION THREE

Given the following `KumonTuition`, `TreeNode`, `BSTKumonTuition` ADTs and Table 1:

```
public KumonTuition
{
    private String subjectName;
    private double fee;
    private double hour;
    private String room;
    private int noStudent;

    public KumonTuition(String className, double hour, String room, int
        noStudent){...}
    ...
}

public class TreeNode
{
    protected TreeNode left;
    protected TreeNode right;
    protected KumonTuition data;

    public TreeNode(Object data){...}
    ...
}

public class BSTKumonTuition
{
    protected TreeNode root;

    public BSTKumonTuition(){...}
    public void displayKumonTuition(){...}
    public int countStudent(){...}
    public void calcFees(){...}
}
```

Table 1 describes the information about Kumon Tuition such as the subjects taken by students, the room numbers, the fee charged for each subject, the number of students per subject and the tuition hour.

Table 1 – The information about Kumon Tuition

Subject	Room	Fee (RM)	No of student per Subject	Hour
English	B1	100	20	1
Bahasa Malaysia	B3	100	20	1
Mathematics	B5	150	15	2
Additional Math	B8	150	15	2
Biology	B4	180	15	1.5
Chemistry	B11	180	15	1.5
Physics	B9	180	15	1.5
History	B10	100	20	1

- Based on the subject name, draw a binary search tree diagram.  
(3 marks)
- Write a method definition for `countStudent()` using a recursive technique to calculate the number of students based on the subject names e.g. English, Biology, etc.  
(5 marks)
- Write a method definition for `calcFees()` using a recursive technique to calculate the total fee of all subjects taken by students.  
(5 marks)
- Write a method definition for `displayKumonTuition()` using a recursive technique to display all tuition classes based on the tuition hour is less than 2 hours and the number of students in a class is more than 15 students.  
(5 marks)

TECHNIQUES RELATED QUESTIONS

QUESTION ONE

- a) There are three different traversals of a binary tree which are inorder traversal, preorder traversal and postorder traversal. Write the recursive definition of the method to implement the postorder traversal algorithm

(3 marks)

- b) Explain the **FOUR (4)** cases that must be taken into count when the deletion operation is done on a binary search tree.

(4 marks)

- c) Construct an AVL tree by using the following list of values and show the balance factor after each insertion :

342 206 444 523 607 301 142 183 172 157 149

(8 marks)

QUESTION TWO

- a) Consider the product of a tree traversal on a binary tree as follows:

Inorder:      **O A K N L S G M Y R P**

Preorder:    **S A O N K L Y G M P R**

Draw the tree.

(5 marks)

- b) Consider the **postfix** expression below and draw the **expression tree**.

**G A - B C + D + / E F + H - +**

(5 marks)

- c) Answer the following questions on **AVL Tree**.

- i) What is AVL tree?

(2 marks)

- ii) Construct an AVL tree by inserting these primary keys. Show step-by-step construction.

**BL56, QA32, RD77, EU98, DD61, FY49**

(8 marks)

QUESTION THREE

- a) The height of a binary tree is the number of nodes on the longest path from the root to a leaf. From the definition above, how many nodes does the full binary tree of height 5 have?  
(2 marks)
- b) Given the following list of item, construct a binary search tree.

**ITS432 MAT560 CSC541 CTU153 QMT300 CSC438 BEL324 ENT650**

(4 marks)

- c) Find the infix, prefix and postfix expressions based on the binary search tree above.  
(6 marks)
- d) Draw an expression tree based on the infix expression below :

$$3 * ((7 + 1) / 4) + (17 - 5)$$

(3 marks)