

# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

[Dashboard](#) / [My courses](#) / [CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES](#) / [Exam](#) / [Exam Paper](#)



Second Semester Examination  
2021/2022 Academic Session

July/August 2022

## CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

Duration: 2 hours 45 mins

This examination paper is divided into **THREE (3) sections**.

**Part A** consists of TEN (10) questions. For every correct answer, you will be given 1 mark and for every wrong answer, 0.25 marks will be deducted from your total marks.

**Part B** also consists of TEN (10) questions. For every correct answer, you will be given 2 marks and for every wrong answer, approximately 0.5 mark will be deducted from your total marks.

No penalties will be imposed to you if you left the question unanswered.

**Part C** is the description and programming section. You are required to type your answer inside the answer box given.

All the best!

Dr Hana & Pn Maziani

This quiz has been configured so that students may only attempt it using the Respondus LockDown Browser.

This quiz closed on Sunday, 31 July 2022, 5:45 PM

The button Launch Lockdown Browser will appear when the time exam is started, please refresh your browser if the button does not appear

CANDIDATES CAUGHT COPYING/CHEATING IN THE EXAMINATION AND FOUND GUILTY MAY BE EXPELLED FROM THEIR STUDIES AT USM

Attempts: 190

### Summary of your previous attempts

Attempt	State	Review
Preview	Finished Submitted Sunday, 31 July 2022, 5:41 PM	Not permitted

Back to the course





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

## Exam Paper

This examination paper is divided into **THREE (3) sections**.

**Part A** consists of TEN (10) questions. For every correct answer, you will be given 1 mark and for every wrong answer, 0.25 marks will be deducted from your total marks.

**Part B** also consists of TEN (10) questions. For every correct answer, you will be given 2 marks and for every wrong answer, approximately 0.5 mark will be deducted from your total marks.

No penalties will be imposed to you if you left the question unanswered.

**Part C** is the description and programming section. You are required to type your answer inside the answer box given.

All the best!

Dr Hana & Pn Maziani

## Instructions

This exam has a time limit of 2 hours 45 mins. Time will count down from the moment you start your attempt and you must submit before it expires. Are you sure that you wish to start now?

Start attempt

Cancel





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **1**

Not yet answered

Marked out of 1.00

Flag question

Edit question

A doubly linked list keeps track of the next node in the list. State which other node it keeps track as well.

- ☐ a. the head node
- ☐ b. the previous node
- ☐ c. the tail node
- ☐ d. itself

Question **2**

Not yet answered

Marked out of 1.00

Flag question

Edit question

Select the correct statement about destructor:

- ☐ a. automatically called when the main program end
- ☐ b. is called from the member of the class
- ☐ c. need to be called explicitly in the main function
- ☐ d. triggered when we finish using all the functions of the class

Next page

Quiz navigation

Part A

1

2

3

4

5

6

7

8

9

10

Part B

11

12

13

14

15

16

17

18

19

20

Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **3**

Not yet answered

Marked out of 1.00

Flag question

Edit question

What is the output of the program for the following input?

5 2 \* 3 3 2 2 + \* + - \*

☐ a.

 Imbalance postfix notation

☐ b.

 21☐ c.☐ d.☐ e.

Question **4**

Not yet answered

Marked out of 1.00

Flag question

Edit question

Identify when the dynamic memory allocation occurs in programming.

☐ a.

 When a pointer is assigned an incorrect address

☐ b.

 When a new variable is created by the compiler☐ c.☐ d.

Previous page

Next page

Quiz navigation

Part A

1

2

3

4

5

6

7

8

9

10

Part B

11

12

13

14

15

16

17

18

19

20

Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **5**

Not yet answered

Marked out of 1.00

Flag question

Edit question

Let  $T$  be a binary search tree with 15 nodes. The minimum possible height of  $T$  is:  
*Note: The height of a tree with a single node is 0*

- ☐ a. 5
- ☐ b. 7
- ☐ c. 4
- ☐ d. 3

Question **6**

Not yet answered

Marked out of 1.00

Flag question

Edit question

In linked list implementation of a queue, front and rear pointers are tracked. Select which of these pointers will change during an insertion into a nonempty queue.

- ☐ a. Only front pointer
- ☐ b. No pointer will be changed
- ☐ c. Only rear pointer
- ☐ d. Both front and rear pointer

Previous page

Next page

Quiz navigation

Part A

1

2

3

4

5

6

7

8

9

10

Part B

11

12

13

14

15

16

17

18

19

20

Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **7**

Not yet answered

Marked out of 1.00

Flag question

Edit question

Among the following list, which one is the best data structure to check whether an arithmetic expression has balanced parenthesis.

- ☐ a. Queue
- ☐ b. List
- ☐ c. Stack
- ☐ d. Tree

Question **8**

Not yet answered

Marked out of 1.00

Flag question

Edit question

Select the statement which is **FALSE** regarding object-oriented programming concepts.

- ☐ a. Object encapsulates both the data and the functions that operate on the data
- ☐ b. A private member function is useful for tasks that are internal to the class, but it is not directly called by statements outside of the class
- ☐ c. Class objects cannot be defined prior to the class declaration
- ☐ d. You must declare all data members of a class before you declare member functions

Quiz navigation

Part A

1

2

3

4

5

6

7

8

9

10

Part B

11

12

13

14

15

16

17

18

19

20

Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **9**

Not yet answered

Marked out of 1.00

Flag question

Edit question

Which of the following operator cannot be overloaded in a friend function?

- ☐ a. ()
- ☐ b. ==
- ☐ c. --
- ☐ d. +

Question **10**

Not yet answered

Marked out of 1.00

Flag question

Edit question

Select the statement which is **TRUE** regarding classes in programming.

- ☐ a. The constructor function must not accept arguments
- ☐ b. More than one constructor functions may be defined for a class
- ☐ c. More than one destructor functions may be defined for a class
- ☐ d. A destructor function can have zero to many parameters

Previous page

Next page

Quiz navigation

Part A

1

2

3

4

5

6

7

8

9

10

Part B

11

12

13

14

15

16

17

18

19

20

Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **11**

Not yet answered

Marked out of 2.00

Flag question

Edit question

- Compute how many steps are involved in the process of deleting a node from a linked list?
- ☐ a. two: remove the node without breaking links, then delete it from memory
  - ☐ b. four: create a new node, insert the new node before the node being deleted, remove the node being deleted, delete the new node
  - ☐ c. one: delete the node from memory
  - ☐ d. three: create a new node, remove the node being deleted, insert the new node

Question **12**

Not yet answered

Marked out of 2.00

Flag question

Edit question

- Choose all the is **TRUE** about inheritance
- ☐ a. A protected-type inheritance will make all member of the based class as protected member of the derived class
  - ☐ b. Inheritance class is a specialized version of the based class
  - ☐ c. A protected-type inheritance will make all protected member of the based class as a protected member of the derived class
  - ☐ d. Must have a consistent class access specifier for all based class
  - ☐ e. The private-type inheritance can allow the based class members to be accessed by any of the function of the based class

Quiz navigation

Part A

1

2

3

4

5

6

7

8

9

10

Part B

11

12

13

14

15

16

17

18

19

20

Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview

Previous page

Next page





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:

This exam is not currently available

Question **13**

Not yet answered

Marked out of 2.00

Flag question

Edit question

Choose all that is **TRUE** about classes and inheritance

- ☐ a. Scope resolution operator is needed when the definition of the method/function is done outside of the class
- ☐ b. Operator overloading can be used to compare selected variables that belong to the respective objects variables.
- ☐ c. When using inheritance, if a method/function of the derived class is having similar name, the computer will choose the based class method/function definition
- ☐ d. to access identifiers local to the class, scope resolution operator is needed for a program with multiple classes
- ☐ e. Operator overloading can be used to compare the variables that belong to the same class.
- ☐ f. to access identifiers local to the class, scope resolution operator is needed for a program with single class

Question **14**

Not yet answered

Marked out of 2.00

Flag question

Edit question

Given the following partial class definition **MysteryList** and function definition **mystery()**, show the functionality of **mystery()**.

```
class MysteryList
{
    private:
        struct ListNode
        {
            int value;
            struct ListNode *next;
        };
        ListNode *head;
        void destroy();// Destroy function

    public:
        // Constructor and Destructor
        void mystery();
};

void MysteryList::mystery ()
{
    ListNode *newHead = nullptr, *newNode, *nodePtr, *tempPtr;

    nodePtr = head;
    while (nodePtr)
    {
        newNode = new ListNode;
        newNode->value = nodePtr->value;
        newNode->next = nullptr;
        if (newHead != nullptr)
        {
            tempPtr = newHead;
            newHead = newNode;
            newNode->next = tempPtr;
        }
        else
        {
            newHead = newNode;
        }
        nodePtr = nodePtr->next;
    }
    destroy();
    head = newHead;
}
```

- ☐ a. Insert a new node at the end of the list
- ☐ b. Reverse the nodes in the list
- ☐ c. Insert a new node at the beginning of the list
- ☐ d. Delete a node at the end of the list

Quiz navigation

### Part A

1

2

3

4

5

6

7

8

9

10

### Part B

11

12

13

14

15

16

17

18

19

20

### Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:

This exam is not currently available

Question **15**

Not yet answered

Marked out of 2.00

Flag question

Edit question

Select the correct deletion case based on the following C++ codes segment on deleting a node from an ordered doubly linked list.

```
nodePtr = head;
while(nodePtr != nullptr && !found)
{
    if (nodePtr->value >= delItem)
    {
        found = true;
    }
    else
    {
        nodePtr = nodePtr->next
    }
    if (nodePtr == nullptr)
    {
        cout<<"Item not in list";
    }
    else
    {
        if (nodePtr->value == delItem)
        {
            {
                trailPtr = nodePtr->previous;
                trailPtr->next = nodePtr->next;
                if (nodePtr->next != nullptr)
                {
                    nodePtr->next->previous = trailPtr;
                }
                if (nodePtr == end)
                {
                    end = trailPtr
                }
                delete nodePtr
            }
        }
        else
        {
            cout<<"Item not in list";
        }
    }
}
```

- ☐ a. Case 2: The item to be deleted is in the first node of the list
- ☐ b. Case 3: The item to be deleted is somewhere in the list
- ☐ c. Case 4: The item to be deleted is not in the list
- ☐ d. Case 1: The list is empty, cannot delete the item

Question **16**

Not yet answered

Marked out of 2.00

Flag question

Edit question

Given the following partial class definition **Node** and **Queue**. Show the purpose of function **fun()**.

```
template <class T>
class Node
{
public:
    T value;
    Node<T> *next;
};

template <class T>
class Queue
{
private:
    Node<T> *front;
    Node<T> *rear;
    int numItems;
public:
    Queue();
    ~Queue();
    bool isEmpty()const;
    void fun(T &);
};

template <class T>
void Queue<T>::fun(T &item)
{
    Node<T> *temp = nullptr;
    if (isEmpty())
    {
        cout << "Are we having fun? ";
    }
    else
    {
        {
            item = front->value;
            temp = front;
            front = front->next;
            delete temp;
            numItems--;
        }
    }
}
```

- ☐ a. Insert value at the front of the queue and copies it into **item**
- ☐ b. Insert value at the rear of the queue and copies it into **item**
- ☐ c. Remove value at the front of the queue and copies it into **item**
- ☐ d. Remove value at the rear of the queue and copies it into **item**

## Quiz navigation

### Part A

1

2

3

4

5

6

7

8

9

10

### Part B

11

12

13

14

15

16

17

18

19

20

### Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:

This exam is not currently available

Question **17**

Not yet answered

Marked out of 2.00

🚩 Flag question

⚙ Edit question

Given the **IntList** class definition, demonstrate which one of the following that correctly insert a value **x** at position **y** in a linked list?

```
class IntList
{
private:
    struct ListNode
    {
        int value;
        struct ListNode *next;
    };
    ListNode *head;
    void destroy();

public:
    IntList()
    { head = nullptr; }
    IntList(const IntList &);
    ~IntList();
    void insert(int, int);
};
```

- ☐ a. `void IntList::insert(int x, int y)`
- ```
{
    ListNode *newNode = new ListNode;
    newNode->value = x;
    newNode->next = nullptr;

    if (head == nullptr)
    {
        head = newNode;
        return;
    }
    if (y == 0)
    {
        newNode->next = new Node;
        head = newNode->next;
        return;
    }

    ListNode *p = head;
    int num = 1;
    while (num <= y)
    {
        if (p->next == nullptr || num == y)
        {
            ListNode *tempPtr = p->next;
            p->next = newNode;
            newNode->next = tempPtr;
            return;
        }
        p = p->next;
        num++;
    }
}
```
- ☐ b. `void IntList::insert(int x, int y)`
- ```
{
    ListNode *newNode = new ListNode;
    newNode->value = x;
    newNode->next = nullptr;

    if (head == nullptr)
    {
        head = newNode;
        return;
    }
    if (y == 0)
    {
        newNode->next = head;
        head = newNode;
        return;
    }

    ListNode *p = head;
    int num = 1;
    while (num <= y)
    {
        if (p->next == nullptr || num == y)
        {
            ListNode *tempPtr = p->next;
            p->next = newNode;
            newNode->next = tempPtr;
            return;
        }
        p = p->next;
        num++;
    }
}
```
- ☐ c. `void IntList::insert(int x, int y)`
- ```
{
    ListNode *newNode = new ListNode;
    newNode->value = x;
    newNode->next = nullptr;

    if (head == nullptr)
    {
        head = newNode;
        return;
    }
    if (y == 0)
    {
        newNode->next = head;
        head = newNode;
        return;
    }

    ListNode *p = head;
    int num = 1;
    while (num <= y)
    {
        if (p->next == nullptr)
        {
            ListNode *tempPtr;
            p->next = newNode;
            newNode->next = tempPtr;
            return;
        }
        p = p->next;
        num++;
    }
}
```
- ☐ d. `void IntList::insert(int x, int y)`
- ```
{
    ListNode *newNode = new ListNode;
    newNode->value = x;
    newNode->next = nullptr;

    if (head == nullptr)
    {
        head = newNode;
        return;
    }
    if (y == 0)
    {
        newNode->next = head;
        head = newNode;
        return;
    }

    ListNode *p = head;
    int num = 1;
    while (num <= y)
    {
        if (p->next == nullptr || num == y)
        {
            ListNode *tempPtr = newNode;
            p->next = newNode;
            newNode->next = tempPtr;
            return;
        }
        p = p->next;
        num++;
    }
}
```

Question **18**

Not yet answered

Marked out of 2.00

🚩 Flag question

⚙ Edit question

Assuming a series of characters is pushed into a stack in the order of A, B, C, D and E. Which of the following is the possible answer(s) sequence that can be obtained from the series of the stack operation?

Choose all correct combinations.

- ☐ a. C, B, D, E, A
- ☐ b. A, B, C, E, D
- ☐ c. E, D, C, B, A
- ☐ d. E, D, C, A, B
- ☐ e. C, D, E, A, B
- ☐ f. E, A, B, C, D

Quiz navigation

Part A

1

2

3

4

5

6

7

8

9

10

Part B

11

12

13

14

15

16

17

18

19

20

Section C

21

22

23

24

25

26

[Finish attempt ...](#)

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **19**

Not yet answered

Marked out of 2.00

Flag question

Edit question

Given a binary search tree which contains the values 33, 39, 40, 41, 52, 57, 60, 63 and 65. Which of the following is possible preorder traversal.

- ☐ a. 57, 41, 33, 39, 40, 52, 60, 63, 65
- ☐ b. 57, 41, 33, 39, 52, 40, 60, 65, 63
- ☐ c. 57, 41, 33, 39, 52, 40, 60, 63, 65
- ☐ d. 57, 41, 33, 39, 40, 60, 52, 65, 63

Question **20**

Not yet answered

Marked out of 2.00

Flag question

Edit question

Choose all that is **FALSE** about overloading constructor:

- ☐ a. The number of parameters of the overloading constructor may not necessarily match the number of variables of the class.
- ☐ b. Overloading constructor will be called after the object trigger the default constructor.
- ☐ c. An overloading constructor can have multiple parameters passing or no parameter at all.
- ☐ d. Overloading constructor can be triggered using operator =
- ☐ e. An overloading constructor can be the only constructor available of a given class.

Previous page

Next page

Quiz navigation

Part A

123456789

10

Part B

111213141516171819

20

Section C

212223242526

Finish attempt ...

Start a new preview



# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **21**

Not yet answered

Marked out of 15.00

Flag question

Edit question

Analyse the following program. Answer the following questions without changing any member access specifier of both classes.

```
6 class Course{
7     private:
8         string courseCode;
9         double fee;
10    public:
11        Course(){courseCode=""; fee=0.0;}
12        ~Course(){courseCode=""; fee=0.0;}
13        void set(string cCode, double charge){
14            courseCode=cCode; fee=charge;}
15        void getCourse(string &cCode, double &charge){
16            cCode=courseCode; charge=fee;}
17    };

7 class Student{
8     private:
9         string name;
10        int matric;
11        Course C[5];
12        int courseCount;
13
14    public:
15        Student(){name=""; matric=0; courseCount=0;}
16        ~Student(){name=""; matric=0; courseCount=0;}
17        void setPersonalDetails(string, int);
18        void getPersonalDetails(string&, int&);
19
20        void regCourse(string, double);
21        double courseFee();
22        void removeCourse();
23        void print();
24    };
```

- Construct a complete removeCourse() method/function. This function is able to identify the most probable course to be removed from the list of registered courses by making sure that the maximum total fee to be paid is not exceeding RM500 for a student. **Note: Assume the function courseFee() will return the total fee a student needs to pay.** (10 marks)
- Write an operator "==" overloading function to identify if two students are taking the same courses AND have the same total fees. If the fees are not the same, the operator overloading function needs to print out appropriate messages. (5 marks)

<p>  
Index Number:  
IC Number:  
</p>

## Quiz navigation

### Part A

1

2

3

4

5

6

7

8

9

10

### Part B

11

12

13

14

15

16

17

18

19

20

### Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview

Question **22**

Not yet answered

Marked out of 5.00

Flag question

Edit question

Analyse the following program. Answer the following questions without changing any member access specifier of both classes.

```
10 class Salary
11 {
12     double basic_salary;
13     double allowance;
14
15    public:
16        Salary(){basic_salary=0; allowance=0;}
17        Salary(int b){
18            basic_salary = b;
19            allowance = 0.5 * b;
20        }
21        void setSalary( int b){
22            basic_salary = b;
23            allowance = 0.5 * b;
24        }
25        void getSalary(){
26            cout << "\nBasic Salary = $" << basic_salary << endl;
27            cout << "Allowance = $" << allowance << endl;
28        }
29    };

31 class Freelance
32 {
33     int no_of_jobs;
34     // assuming that the person can't do more than 20 freelance jobs
35     double income[20];
36
37    public :
38        void assignJobs(){
39            cout << "\nEnter number of jobs done : ";
40            cin >> no_of_jobs;
41            for( int i = 0 ; i < no_of_jobs ; i++ ){
42                cout << "\tEnter income from Job-" << i + 1 << " : ";
43                cin >> income[i];
44            }
45        }
46        void printJobs(){
47            cout << endl;
48            for ( int i = 0 ; i < no_of_jobs ; i++ )
49                cout << "Income from Job- " << i + 1 << " = $" << income[i] << endl;
50        }
51    };
```

- Write the friend function prototype total\_income in both classes. (1 marks)
- Write a friend function called total\_income to calculate the income for both salary and freelance of a person. (4 marks)

<p>  
Index Number:  
IC Number:  
</p>



# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:  
This exam is not currently available

Question **23**

Not yet answered

Marked out of 20.00

Flag question

Edit question

Analyse the following situation.

During a recent reunion gathering, the guests form themselves into two groups and play the broken telephone game. It is a game where the first person in line come up with a phrase and then whisper it into the ear of the next person in line. Then, this person has to whisper what he or she heard in the next person's ear. This continues until the last person has heard the phrase. The trick is that each person only has one chance to whisper the phrase, no repeating is allowed. When the phrase comes around to the last person in the line, he or she need to say it out loud. The game finishes by comparing the original phrase to what the last person heard and said.

The winner is the group that keeps the original phrase until the last person in the line. If both groups did not maintain the original phrase, then the winner is the group that has shorter line from the location of the last person to the person who started the wrong phrase.

(a) Based on the situation given above, determine the most suitable data structure to implement this game. (4 marks)

(b) Demonstrate the C++ code to track the person who started the wrong phrase. (8 marks)

(c) Each group wants to invite a new guest into their group and play again. Construct the function in C++ code to add a new person at the end of the line. (8 marks)

↓

A ▾

B

I

☰

☷

✂

↺

H-P

<p>

Index Number:

IC Number:

</p>

Question **24**

Not yet answered

Marked out of 15.00

Flag question

Edit question

Analyse the following situation:

A new local burger restaurant recently open in town and business is booming. Many customers come and place their orders at the counter, make payments and wait for their food to be served. In order to keep up with the crowd, the manager needs to keep track of the orders by making sure waiting list of each customer is handled fairly.

(a) Determine the suitable data structure for keeping track of customers' orders. (3 marks)

(b) Construct the class definition in C++ codes for the data structure answered in question (a) that includes the necessary variables and function prototypes. (6 marks)

(c) Manager wants to display all the orders currently waiting to be served to customers. Construct the function in C++ using the class definition answered in question (b). (6 marks)

↓

A ▾

B

I

☰

☷

✂

↺

H-P

<p>

Index Number:

IC Number:

</p>

Quiz navigation

Part A

1

2

3

4

5

6

7

8

9

10

Part B

11

12

13

14

15

16

17

18

19

20

Section C

21

22

23

24

25

26

Finish attempt ...

Start a new preview





# CPT113-PROGRAMMING METHODOLOGY & DATA STRUCTURES

You can preview this quiz, but if this were a real attempt, you would be blocked because:

This exam is not currently available

Question **25**

Not yet answered

Marked out of 7.00

Flag question

Edit question

Evaluate the following shape:

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*  
\*\*  
\*  
\*  
\*  
\*\*  
\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

Without using any loop, write a C++ program to print the diagram recursively.

<p>  
Index Number:  
IC Number:  
</p>

Question **26**

Not yet answered

Marked out of 8.00

Flag question

Edit question

The following three are the traversal sequence for preorder, inorder and postorder of a binary tree.

- I. 33 27 77 25 55 53 67 43 88 99
- II. 99 55 27 33 25 77 88 43 53 67
- III. 27 33 55 77 25 99 88 53 43 67
- a) Analyse which one is preorder, inorder and postorder. Explain and show your work. (4 marks)
- b) Based on question 6, build the tree and answer the following questions: (4 marks)

I. What is the node of level 1 left subtree

ii. What is the node of level 1 right subtree

iii. List all parents with only right leaf

iv. List all leaves of the tree

<p>  
Index Number:  
IC Number:  
</p>

Quiz navigation

Part A

123456789

10

Part B

111213141516171819

20

Section C

212223242526

Finish attempt ...

Start a new preview