ISC · CLASS 12 · COMPUTER SCIENCE

ISC Class 12 Computer Science Important Questions 2027

45 question patterns that repeated across past papers · the years show every time each was asked

Scan for the online version

More free papers, important questions & topics:
guesspaper.app/isc/class-12/computer-science/important-questions/

Propositional Logic

1. Verify by truth table whether a proposition is valid / a tautology, contradiction or contingency

Asked 6× · Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 2/3 marks

Draw a truth table to verify if the following proposition is a Tautology, a Contradiction or a Contingency.
(A ∧ ~B) => (~A ∨ B)

2. Converse / inverse / contrapositive of an implication

Asked 6× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2025 · 1 marks

Consider the two propositions given below:
A = You use ecofriendly methods
B = Pollution is reduced
If A implies to B, then write its Contrapositive statement.

3. Assertion-reason: equivalence laws of propositions (implication, De Morgan)

Asked 2× · Board 2025, Board 2026 · 1 marks

Study the given propositions and the statements marked Assertion and Reason that follow. Choose the correct option based on your analysis.
P = You practise regularly
Q = You become skilled
S1 = P => Q
S2 = ~P ∨ Q
Assertion: S1 and S2 are logically equivalent.
Reason: A conditional statement P => Q can be expressed as ~P ∨ Q
(a) Both Assertion and Reason are true and Reason is the correct explanation for Assertion.
(b) Both Assertion and Reason are true but Reason is not the correct explanation for Assertion.
(c) Assertion is true and Reason is false.
(d) Both Assertion and Reason are false.

Boolean Algebra

4. MCQ: complement of a Boolean expression (De Morgan's theorem)

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 1 marks

The complement of the Boolean expression a • b' + a' + a' • b is:
(a) a + b' • a' • a' + b
(b) (a' + b) • a • (a + b')
(c) (a + b') • a' • (a + b')
(d) a' • b + a + a • b'

5. MCQ: dual of a Boolean equation (principle of duality)

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 1 marks

According to the Principle of Duality, the Boolean equation Q' • 0 + P' • Q' + P' • Q = P' will be equivalent to:
(a) Q • 0 + P • Q + P • Q' = P
(b) Q' • 1 + P' • Q' + P • Q' = P'
(c) (Q' + 1) • (P' + Q') • (P' + Q) = P'
(d) (Q' + 0) • (P' + Q') • (P' + Q) = P'

6. Truth table from a word problem (eligibility criteria) and write its SOP/POS expression

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 5 marks

According to the ancient laws of the Valley of Peace, a candidate can become the Dragon Warrior only if they satisfy any one of the following conditions:
• The candidate belongs to the Panda Clan and has been trained for more than 5 years under the Grand Master
OR
• The candidate possesses the Secret Chi Power but does not belong to the Panda Clan
OR
• The candidate is recommended by the Grand Master, but neither belongs to the Panda Clan nor has been trained for more than 5 years
The inputs are:
C : Belongs to the Panda Clan
T : Trained for more than 5 years
P : Possesses Secret Chi Power
R : Recommended by the Grand Master
(In all the above cases, 1 indicates YES and 0 indicates NO)
Output: X - Denotes eligibility to become the Dragon Warrior (1 = eligible and 0 = not eligible)
Draw the truth table for the inputs and outputs given above. Write the SOP expression for X(C, T, P, R).

7. MCQ: identify the Boolean law (distributive, absorption, associative, idempotence...)

Asked 6× · Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 1 marks

The Boolean equations a + 1 = a and a • 0 = 0 correspond to:
(a) Involution Law
(b) Law of Identity
(c) Distributive Law
(d) Law of Complements

8. Reduce F = π(...) by 4-variable K-map and draw the logic gate diagram

Asked 6× · Board 2023, Board 2024, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 5 marks

(a) Reduce the Boolean function F(A,B,C,D) = π (0,1,2,3,7,8,9,10,11,12,13,15) by using 4-variable Karnaugh map, showing the various groups (i.e., octal, quads and pairs). [4]
(b) Draw the logic gate diagram for the reduced expression. Assume that the variables and their complements are available as inputs. [1]

9. Reduce the word-problem expression X by 4-variable K-map and draw the gate diagram

Asked 4× · Board 2024, Board 2025, Sample 2024, Sample 2025 · 5 marks

[context: X (A, S, C, L) is the POS expression from the truth table of part (i) - superhero access to the Avengers facility: Avengers membership, S.H.I.E.L.D. permit, clearance badge, recognised ally]
Reduce the above expression X (A, S, C, L) by using 4-variable Karnaugh map, showing the various groups (i.e., octal, quads and pairs).
Draw the logic gate diagram using NOR gates only for the reduced expression. Assume that the variables and their complements are available as inputs.

10. Canonical / cardinal form of a Boolean function (1 mark)

Asked 4× · Board 2024, Board 2025, Sample 2023, Sample 2024 · 1 marks

The canonical expression of F(P, Q, R) = π (2, 5, 7) is:
(a) (P + Q' + R) • (P' + Q + R') • (P' + Q' + R')
(b) (P • Q' • R) + (P' • Q • R') + (P' • Q' • R')
(c) (P' + Q + R') • (P + Q' + R) • (P + Q + R)
(d) (P' • Q • R') + (P • Q' • R) + (P • Q • R)

11. Write the minterm / maxterm for given variable values (1 mark)

Asked 4× · Board 2023, Board 2024, Board 2025, Sample 2024 · 1 marks

Write the minterms in canonical form for the Boolean Function X (A, B), from the truth table given below:

A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

12. Write maxterm and minterm / convert cardinal form to canonical form (2 marks)

Asked 4× · Board 2023, Board 2024, Board 2026, Sample 2023 · 2 marks

Given that P = 0, Q = 1, R = 0, S = 0, write its:
(a) Maxterm [1]
(b) Minterm [1]

13. Reduce F = Σ(...) by 4-variable K-map and draw the logic gate diagram

Asked 3× · Board 2023, Board 2025, Sample 2023 · 5 marks

(a) Reduce the Boolean function F(P, Q, R, S) = Σ(0,1,2,5,7,8,9,10,13,15) by using 4-variable Karnaugh map, showing the various groups (i.e., octal, quads and pairs). [4]
(b) Draw the logic gate diagram using NAND gates only for the reduced expression. Assume that the variables and their complements are available as inputs. [1]

14. MCQ: reduced form of a Boolean expression

Asked 2× · Board 2023, Sample 2025 · 1 marks

The complement of the reduced expression of F(A,B) = Σ (0,1,2,3) is:
(a) 1
(b) A•B
(c) 0
(d) A' + B'

Guesspaper.app

15. Simplify / complement and reduce an expression using Boolean laws

Asked 2× · Board 2024, Sample 2023 · 2 marks

Find the complement of the following expression and reduce it by using Boolean laws.
P • (P + Q) • Q • (Q + R')

Computer Hardware

16. Encoder / decoder / multiplexer: define, differentiate and draw the logic circuit

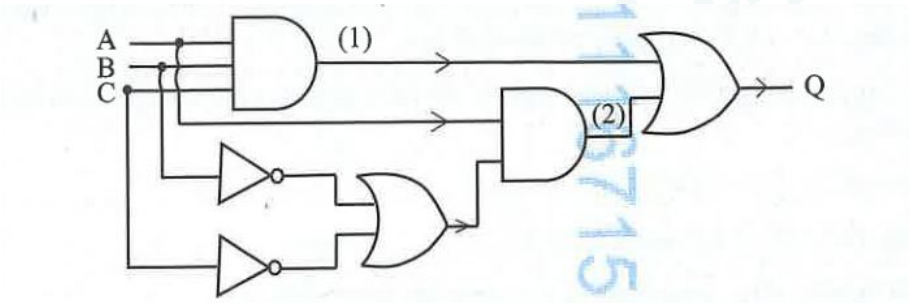
Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 5 marks

(a) What is a decoder? [1]
(b) Draw the logic gate diagram for decoding the binary numbers {0011, 0100, 0101, 0111, 1011, 1110} to hexa-decimal numbers. [3]
(c) State any one application of a multiplexer. [1]

17. Derive (and simplify) the Boolean expression from a given logic gate diagram

Asked 5× · Board 2023, Board 2025, Board 2026, Sample 2024, Sample 2025 · 3/5 marks

From the logic gate diagram given below, derive the Boolean expression for (1), (2) and Q. Reduce the derived expression.
[Logic diagram with inputs A, B, C: an AND gate gives (1); NOT gates and an OR gate feed a second AND gate giving (2); (1) and (2) feed an OR gate giving Q - see figure]



18. Half adder / full adder: expressions and logic circuit (NAND only or using two half adders)

Asked 3× · Board 2023, Board 2026, Sample 2023 · 3 marks

Construct the logic gate diagram for a Full Adder using two Half Adders.

19. Half adder and full adder: applications / gates required (1 mark)

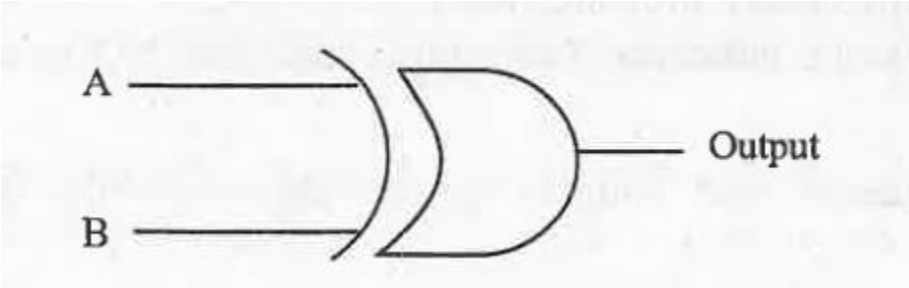
Asked 2× · Sample 2024, Sample 2025 · 1 marks

A full adder needs five gates and those are 3 AND gates, 1 OR gate and 1 XOR gate. When a full adder is constructed using 2 half adders, it also requires 5 gates. State the names along with the quantity those gates.

20. Output of a given logic gate for given inputs

Asked 2× · Board 2025, Sample 2025 · 2 marks

Study the logic gate diagram given below and answer the questions that follow:
[Logic gate symbol with inputs A and B and one Output - see figure]
What will be the output of the above gate when:
(a) A = 1, B = 0 [1]
(b) A = 1, B = 1 [1]



21. Draw a basic gate (OR / AND) using only NAND or only NOR gates

Asked 2× · Board 2024, Board 2026 · 2/3 marks

Draw the logic gate diagram for 2-input AND gate using NOR gates only. Show the expression at each step.

22. Gate equivalent to logic gates connected in series

Asked 2× · Board 2024, Sample 2023 · 1 marks

When a sequence of OR, NOT, NOR are connected in series, the logic gate obtained is:
(a) AND
(b) NOT
(c) OR
(d) XOR

Primitive Values, Variables and Statements

23. Dry-run an iterative (loop) function and state what it computes

Asked 2× · Board 2023, Sample 2023 · 3 marks

With reference to the code given below, answer the questions that follow along with dry run / working.

```
boolean num(int x)
{ int a=1;
   for (int c=x; c>0; c/=10)
      a *= 10;
   return (x*x%a)==x;
}
```

(a) What will the function num( ) return when the value of x=25? [2]
(b) What is the method num( ) performing? [1]

Methods

24. Fill in the missing code ?1? ?2? ?3? in a method (digit / number logic)

Asked 5× · Board 2024, Board 2025, Board 2026, Sample 2024, Sample 2025 · 3 marks

The following function isTech( ) is a part of some class which is used to check if a given number is a Tech number or not. There are some places in the code marked by ?1?, ?2?, ?3? which may be replaced by a statement / expression so that the function works properly.
A number is Tech number if the count of digits is even and the square of the sum of its two equal halves is equal to the number itself.
Example: $2025 = 20 + 25 = (45)^2 = 2025$

```
boolean isTech(int n)
{
   String s = String.valueOf(n);
   int len = ?1?;
   if(len % 2 != 0)
      return ?2?;
   int first = Integer.parseInt(s.substring(0, len / 2));
   int second = Integer.parseInt(s.substring(len / 2));
   int sum = first + second;
   return ?3? == n;
}
```

What are the expressions or statements at ?1?, ?2? and ?3?

25. MCQ: which statements are valid for the given code (static final member / this)

Asked 3× · Board 2025, Board 2026, Sample 2025 · 1 marks

Consider the following statement written in class Student where school_Name is its data member.

```
static final String school_Name = "Co-Ed School";
```

Which of the following statements are valid for school_Name?
I. All objects of class Student share the same value of school_Name.
II. The value of school_Name cannot be changed during program execution.
III. The keywords static and final cannot be used together for a variable.
(a) Only I and II
(b) Only II and III
(c) Only I and III
(d) Only III

26. State the relevance of static / this keywords; class vs instance variable

Asked 3× · Board 2023, Board 2024, Sample 2024 · 1 marks

State any one difference between instance variable and class variable.

Arrays

27. Find the address / base address of a 2-D array element (row major or column major)

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 2 marks

A matrix G[3...7, 2...5] is stored in the memory, with each element requiring 4 bytes of storage. If the address of G[5][4] is 6000, find the base address when the matrix is stored Column Major Wise.

28. Class program: combine the arrays of two objects (object passed and returned)

Asked 3× · Board 2026, Sample 2023, Sample 2025 · 10 marks

A class TimeOp has been defined to add any two accepted time periods.
Example: Time A = 6 hours 35 minutes 40 seconds
Time B = 7 hours 45 minutes 30 seconds
Time A + Time B = 14 hours 21 minutes 10 seconds
(where 60 minutes = 1 hour and 60 seconds = 1 minute)
The details of the members of the class are given below:
Class name : TimeOp
Data member/instance variable:
  arr[ ] : integer array to hold three elements (hours, minutes and seconds)
Methods/Member functions:
  TimeOp( ) : default constructor
  void readTime( ) : to accept the elements of the array
  TimeOp addTime(TimeOp tt) : to add the time of the parameterised object tt and the current object, to store it in a local object and return it
  void dispTime( ) : to display the array elements in hours:minutes:seconds format
Specify the class TimeOp giving the details of the constructor( ), void readTime( ), TimeOp addTime(TimeOp) and void dispTime( ). Define the main( ) function to create objects and call the functions accordingly to enable the task.

29. Class program on a matrix (2-D array): transpose, fill, compare columns

Asked 3× · Board 2023, Board 2025, Sample 2024 · 10 marks

Design a class Colsum to check if the sum of elements in each corresponding column of two matrices is equal or not. Assume that the two matrices have the same dimensions.
Example:
Input:
MATRIX A MATRIX B
2 3 1 7 4 2
7 5 6 1 3 1
1 4 2 2 5 6
Output: Sum of corresponding columns is equal.
The details of the members of the class are given below:
Class name : Colsum
Data members/instance variables:
  mat[ ][ ] : to store the integer array elements
  m : to store the number of rows
  n : to store the number of columns
Member functions/methods:
  Colsum(int mm, int nn) : parameterised constructor to initialise the data members m = mm and n = nn
  void readArray( ) : to accept the elements into the array
  boolean check(Colsum A, Colsum B) : to check if the sum of elements in each column of the objects A and B is equal and return true otherwise, return false
  void print( ) : to display the array elements
Specify the class Colsum giving details of the constructor(int, int), void readArray( ), boolean check(Colsum, Colsum), and void print( ). Define the main() function to create objects and call the functions accordingly to enable the task.

Strings

30. Class program: rearrange / process the characters of a word (flip halves, pendulum, mask removal, ASCII codes)

Asked 4× · Board 2024, Board 2025, Board 2026, Sample 2025 · 10 marks

Design a class PendulumS to perform an operation on a word containing alphabets in upper case only. Rearrange the word by putting the lowest ASCII value character at the centre and the second lowest ASCII value character to its right and the third to its left and so on.
Example 1 : Input : COMPUTER Output : TPMCEORU
Example 2 : Input : SCIENCE Output : SIECCEN
The details of the members of the class are given below:
Class name : PendulumS
Data members/instance variables:
  wrd : to store the original word
  newwrd : to store the rearranged word
Methods/Member functions:
  PendulumS(String k) : parameterised constructor to initialise wrd = k and newwrd = ""
  int minCharIndex(String str) : to find the index of the minimum ASCII value character in str and return it
  void arrange( ) : to rearrange the characters of wrd as per the given instructions and store it in newwrd by invoking minCharIndex( )
  void display( ) : to display the original word and the rearranged word
Specify the class PendulumS giving the details of the constructor( ), int minCharIndex(String), void arrange( ) and void display( ). Define the main( ) function to create an object and call the functions accordingly to enable the task.

31. Class program: replace each vowel in a word by the next vowel

Asked 2× · Sample 2023, Sample 2024 · 10 marks

A class Encode has been defined to replace only the vowels in a word by the next corresponding vowel and form a new word.
i.e. A → E, E → I, I → O, O → U, U → A and a → e, e → i, i → o, o → u, and u → a
Example: Input: Institution
Output: Onstotatoun
Some of the members of the class are given below:
Class name : Encode
Data members/instance variables:
  word : to store a word
  length : integer to store the length of the word
  new_word : to store the encoded word
Methods / Member functions:
  Encode( ) : default constructor to initialize data members with legal initial values
  void acceptWord( ) : to accept a word
  void nextVowel( ) : to replace only the vowels from the word stored in 'word' by the next corresponding vowel and to assign it to 'newword', with the remaining alphabets unchanged
  void display( ) : to display the original word along with the encrypted word
Specify the class Encode giving details of the constructor( ), void acceptWord( ), void nextVowel( ) and void display( ). Define a main( ) function to create an object and call the functions accordingly to enable the task.

Recursion

32. Dry-run a recursive function: find its output and state what it does

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 3 marks

The following function workOut( ) is a part of some class. Assume 'n' is a positive integer.

```
String workOut (int n)
{ if (n == 0)
      return "";
   int rem = n % 16;
   char cr = (rem < 10) ? (char)(rem + '0') : (char)(rem - 10 + 'A');
   return workOut (n / 16) + cr;
}
```

Answer the questions given below with the dry run / working.

(a) What will the function workOut(220) return? [2]

(b) What is the function workOut( ) performing apart from recursion? [1]

33. Class program with a recursive method on a number (Dudeney, Pronic, Evil, Pernicious, Trimorphic, LCM, decimal to hex)

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 8/10 marks

A class Trimorphic has been defined to accept a positive integer from the user and display if it is a Trimorphic number or not.

[A number is said to be Trimorphic if the cube of the number ends with the number itself.]

Example 1: 24^3 = 13824 ends with 24

Example 2: 5^3 = 125 ends with 5

The details of the members of the class are given below:

Class name : Trimorphic

Data members/instance variables:

n : to store the number

cube : to store the cube of the number

Methods/Member functions:

Trimorphic( ) : constructor to initialise the data members with legal initial values

void accept( ) : to accept a number

boolean check(int num, long c) : to compare num with the ending digits of c using recursive technique

void result( ) : to check whether the given number is a trimorphic number by invoking the function check( ) and to display an appropriate message

Specify the class Trimorphic giving the details of the constructor( ), void accept( ), boolean check( ) and void result( ). Define the main( ) function to create an object and call the functions accordingly to enable the task.

34. Assertion-reason: recursion, memory and the call stack

Asked 2× · Board 2024, Sample 2025 · 1 marks

Given below are two statements marked Assertion and Reason. Read the two statements carefully and choose the correct option.

Assertion: Recursion utilises more memory as compared to iteration.

Reason: Time complexity of recursion is higher due to the overhead of maintaining the function call stack.

(a) Both Assertion and Reason are true and Reason is the correct explanation for Assertion.

(b) Both Assertion and Reason are true but Reason is not the correct explanation for Assertion.

(c) Assertion is true and Reason is false.

(d) Assertion is false and Reason is true.

Inheritance, Interfaces and Polymorphism

35. Inheritance program: define the subclass (constructor, compute method, show) given the super class

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 5 marks

A superclass Hotel has been defined to store the details of a hotel. Define a subclass Customer to calculate the total bill for a customer as per the following criteria:
Room type | Additional Amount
Executive | 10% of room rent
Suite | 20% of room rent
The details of the members of both the classes are given below:
Class name : Hotel
Data members/instance variables:
  hname : to store hotel name
  roomrent : to store the rent per day
Methods/Member functions:
  Hotel(...) : parameterised constructor to assign values to its data members
  void show( ) : to display hotel details
Class name : Customer
Data members/instance variables:
  cname : to store customer name
  days : to store the number of days of stay
  type : to store the room type
  surcharge : to store the additional amount
  amt : to store the total amount
Methods/Member functions:
  Customer(...) : parameterised constructor to assign values to data members of both the classes
  void compute( ) : to calculate the surcharge based on the room type as given above. Also, to calculate the total amount as: (room rent + surcharge) × days
  void show( ) : to display hotel and customer details
Assume that the superclass Hotel has been defined. Using the concept of Inheritance, specify the class Customer, giving details of constructor(...), void compute( ) and void show( ).
The super class, main function and algorithm need NOT be written.

36. Interfaces in Java: use / purpose / properties

Asked 5× · Board 2023, Board 2024, Board 2026, Sample 2023, Sample 2024 · 1 marks

Given below are two statements marked, Assertion and Reason. Read the two statements carefully and choose the correct option.
Assertion: An interface in Java contains abstract and non-abstract methods.
Reason: All methods in an interface must be implemented by any class that extends this interface.
(a) Both Assertion and Reason are true and Reason is the correct explanation for Assertion.
(b) Both Assertion and Reason are true but Reason is not the correct explanation for Assertion.
(c) Assertion is true and Reason is false.
(d) Both Assertion and Reason are false.

37. MCQ: keyword / type of inheritance (multilevel, extends)

Asked 2× · Board 2023, Sample 2023 · 1 marks

The keyword that allows multi-level inheritance in Java programming is:
(a) implements
(b) super
(c) extends
(d) this

Stacks and Queues

38. Convert an infix expression to postfix / prefix notation

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 2 marks

Convert the following infix notation to prefix form.
P / Q + (S * F + X / R)

39. Queue / circular queue / dequeue class: write the add and remove functions and name the entity

Asked 5× · Board 2023, Board 2025, Sample 2023, Sample 2024, Sample 2025 · 5 marks

A circular queue is a linear data structure that allows data insertion at the rear and removal from the front, with the rear end connected to the front end forming a circular arrangement.
The details of the members of the class are given below:
Class name : CirQueue
Data members/instance variables:
  Q[ ] : array to hold integer values
  cap : maximum capacity of the circular queue
  front : to point the index of the front
  rear : to point the index of the rear
Methods/Member functions:
  CirQueue(int n) : constructor to initialise cap = n, front = 0 and rear = 0
  void push(int v) : to add integers from the rear index if possible else display the message "QUEUE IS FULL"
  int remove( ) : to remove and return the integer from front if any, else return -999
  void print( ) : to display the elements of the circular queue in the order of front to rear
(i) Specify the class CirQueue giving the details of the functions void push(int) and int remove( ). Assume that the other functions have been defined. The main( ) function and algorithm need NOT be written. [4]
(ii) State one application of a circular queue. [1]

40. Stack class: write the push and pop functions and name the entity (LIFO)

Asked 2× · Board 2024, Board 2026 · 5 marks

In any internet browser, a user can visit new webpages and go back to previously visited webpages. Each new webpage URL is stored in browser's memory such that when the user clicks 'Back' button, the previous webpage gets displayed.
The details of the members of the class are given below:
Class name : Browser
Data members/instance variables:
  pages[ ] : an array to hold the URLs of visited webpages
  max : to store the maximum capacity of the array
  top : to point to the index of the last visited webpages
Methods/Member functions:
  Browser(int cap) : constructor to assign max = cap and top = -1
  void visit(String url) : to add URL of a new webpage if possible, else display the message "Browser history full"
  String back( ) : to remove and return the last visited webpage URL, if present, else to return the message "No previous browser history"
(i) Specify the class Browser giving details of the functions void visit(String) and String back( ). Assume that the other functions have been defined. [4]
(ii) Name the entity described above and state its principle. [1]

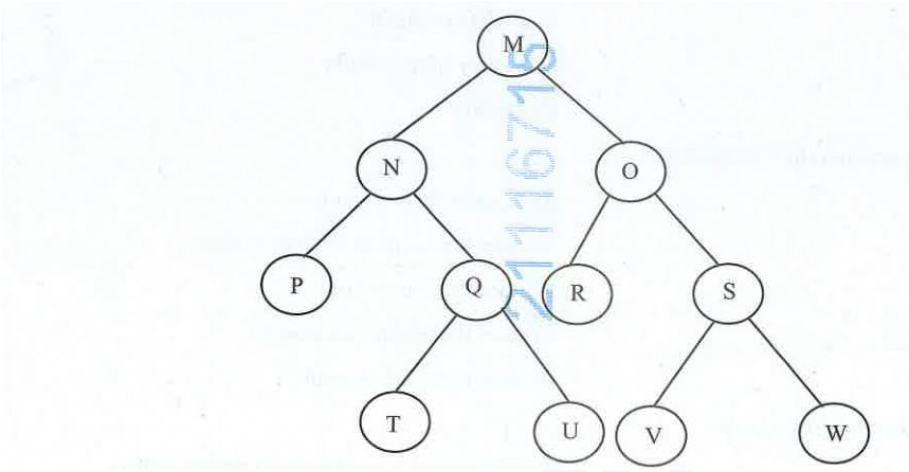
Linked Lists and Binary Trees

41. Binary tree diagram: traversal (pre/in/post-order) and terms (level, depth, size, leaves, siblings)

Asked 7× · Board 2023, Board 2024, Board 2025, Board 2026, Sample 2023, Sample 2024, Sample 2025 · 3 marks

Answer the following questions based on the diagram of a Binary Tree given below:

(a) Write the pre-order traversal of the above tree. [1]
(b) State the level of Node Q, when the root is at level 0. [1]
(c) State the size of the left subtree and the right subtree. [1]



42. Write an algorithm or method on a singly linked list (insert, search, count, sum)

Asked 5× · Board 2023, Board 2025, Board 2026, Sample 2023, Sample 2025 · 2 marks

A linked list is formed from the objects of the class Word. The structure of the class Word is given below:

```
class Word
{
   String value;
   Word next;
}
```

Write an Algorithm OR a Method to count and display the number of nodes whose value starts with a consonant.
The method declaration is as follows:

```
void countConsonant(Word first)
```

43. What is a linked list / importance of its reference part

Asked 2× · Board 2023, Sample 2023 · 1 marks

What is the importance of the reference part in a Linked List?

Complexity and Big O Notation

44. Big O (worst case) of a given loop / nested-loop code segment

Asked 3× · Board 2025, Board 2026, Sample 2025 · 1 marks

The worst case complexity for following code segment is:

```
for(int i=1; i<= n; i++)
{
   for(int j=1; j<= i; j++)
   {
      statement;
   }
}
```

(a) O(n+i)
(b) O(n×i)
(c) O(n)
(d) $O(n^2)$

45. Explain dominant term / constant factor in complexity with an example

Asked 2× · Board 2024, Sample 2024 · 2 marks

With the help of an example, briefly explain the dominant term in complexity.

guesspaper.app