

Enrollment No./Seat No.:

## GUJARAT TECHNOLOGICAL UNIVERSITY

**DIPLOMA IN ENGINEERING - SEMESTER - III EXAMINATION - WINTER 2025**

**Subject Code: DI03016031**

**Date: 08-12-2025**

**Subject Name: Data Structures with Python**

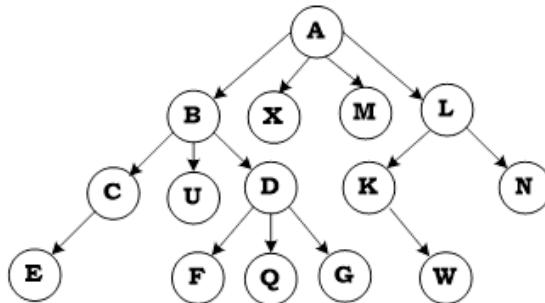
**Time: 10:30 AM TO 01:00 PM**

**Total Marks: 70**

**Instructions:**

- 1. Attempt all questions.**
- 2. Make Suitable assumptions wherever necessary.**
- 3. Figures to the right indicate full marks.**
- 4. Use of simple calculators and non-programmable scientific calculators are permitted.**
- 5. English version is authentic.**

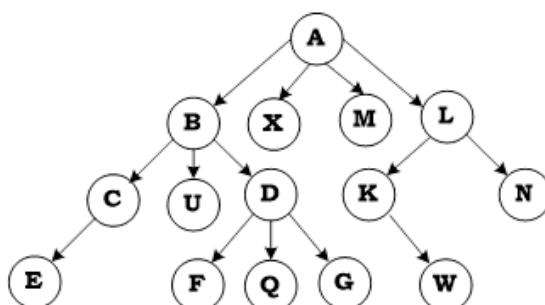
|                                                                                                                  | <b>Marks</b> |
|------------------------------------------------------------------------------------------------------------------|--------------|
| <b>Q.1 (a)</b> Find the in-degree and out-degree of the given nodes from the following tree:<br>A, B, U, K, W, L | <b>03</b>    |



**(અ) નીચેના ટ્રી ના આપેલ નોડ માટે ઇન-કીગ્રી અને આઉટ-કીગ્રી શોધો.**

A, B, U, K, W, L

**03**



**(b) List out different types Data Structures. Explain each with suitable example.**

**04**

**(અ) વિવિધ પ્રકારની ડેટા સ્ટ્રક્ચર્સની યાદી બનાવો. દરેકને યોગ્ય ઉદાહરણ સાથે સમજાવો.**

**04**

**(c) Define: a) Polymorphism b) Abstraction.**

**07**

Write a python program to create a base class **Animal** with a method **make\_sound()** that prints "Some generic sound". Create subclasses **Dog** and **Cat** that override **make\_sound()** to print "**Bark**" and "**Meow**" respectively. Instantiate each class and call **make\_sound()** to demonstrate method overriding.

- (ક) વ્યાખ્યાયિત કરો: a) પોલીમોર્ફિઝમ b) એબસ્ટ્રેક્શન.07  
 પાયથન પ્રોગ્રામ લખો જે એક બેઝ કલાસ **Animal** બનાવે, જેમાં એક મથોડ **make\_sound()** હોય જે "Some generic sound" પ્રિન્ટ કરે. ત્યારબાદ **Dog** અને **Cat** નામની સબકલાસ બનાવો, જે **make\_sound()** મથોડને ઓવરરાઇટ કરે અને ક્રમશ: "Bark" અને "Meow" પ્રિન્ટ કરે. દરેક કલાસના ઓફજેક્ટ બનાવો અને **make\_sound()** કોલ કરો જેથી મથોડ ઓવરરાઇટિંગ દર્શાવવામાં આવે.

### OR

- (c) Define: Inheritance. List out types of it. Explain Multiple and Hierarchical inheritance with suitable example. 07  
 (ક) વ્યાખ્યાયિત કરો: ઇનહેરિટન્સ. તેના પ્રકારો ની ચાદી આપો. અનુરૂપ ઉદાહરણ સાથે મહિંપલ ઇનહેરિટન્સ અને હાયરાર્કિલ ઇનહેરિટન્સ સમજાવો. 07

**Q.2 (a)** Differentiate: Simple Queue vs Circular Queue. 03

(અ) તફાવત કરો: સિમ્પલ ક્યૂ vs સરક્યૂલર ક્યૂ. 03

(બ) Define: Queue. Write and Explain Insert algorithm of Simple Queue. 04

(બ) વ્યાખ્યાયિત કરો: ક્યૂ. સિમ્પલ ક્યૂ પર ઇન્સર્ટ ઓપરેશન માટે અલ્ગોરિધમ લખો અને સમજાવો. 07

(ચ) Perform following operations on given expression: 07

$$(5 + 2) * 3 + 24 / (6 + 3 * 2 * 3) + 5$$

I) Convert infix expression to postfix expression using stack.

II) Evaluate that postfix expression using stack.

(ક) આપેલ સમીકરણ પર નીચેના ઓપરેશન કરો: 07

$$(5 + 2) * 3 + 24 / (6 + 3 * 2 * 3) + 5$$

I) સ્ટેકનો ઉપયોગ કરીને ઇન્ડિક્સ એક્સપ્રેશનને પોસ્ટફિક્સ એક્સપ્રેશનમાં કન્વર્ટ કરો.

II) સ્ટેકનો ઉપયોગ કરીને તે પોસ્ટફિક્સ સમીકરણની કિંમત શોધો.

### OR

(એ) List out and explain applications of Queue. 03

(અ) ક્યૂ ની એપ્લિકેશન ની ચાદી બનાવો અને સમજાવો. 03

(બ) Define: Circular Queue. Explain Concept of Circular Queue with example. 04

(બ) વ્યાખ્યાયિત કરો: સકર્યુલર ક્યૂ. સકર્યુલર ક્યૂ નો કન્સેપ્ટ ઉદાહરણ સાથે સમજાવો. 07

(ચ) Define: Stack. Write and Explain following algorithms. 07

I) An algorithm to convert infix expression to postfix expression using stack.

II) Push algorithm of stack.

(ક) વ્યાખ્યાયિત કરો: સ્ટેક. નીચેના અલ્ગોરિધમ લખો અને સમજાવો. 07

I) સ્ટેકનો ઉપયોગ કરીને ઇન્ડિક્સ એક્સપ્રેશનને પોસ્ટફિક્સ એક્સપ્રેશનમાં કન્વર્ટ કરવા માટે અલ્ગોરિધમ.

II) સ્ટેકનો પુશ અલ્ગોરિધમ.

**Q.3 (એ)** Explain asymptotic notation with its types. 03

(અ) એસિમ્પ્ટોટિક નોટેશનને તેના પ્રકારો સાથે સમજાવો. 03

(બ) Define: Constructor. Explain parameterized constructor with suitable example. 04

(બ) વ્યાખ્યાયિત કરો: કન્સ્ટ્રક્ટર. પેરામીટરાઈડ કન્સ્ટ્રક્ટરને ચોગ્ય ઉદાહરણ સાથે સમજાવો. 07

(ચ) Write an algorithm for selection sort. Give the trace to sort following numbers using selection sort method. 07

**38, 25, 64, 88, 55, 72, 98, 11**

(ક) સિલેક્શન સોર્ટ માટે અલ્ગોરિધમ લખો. સિલેક્શન સોર્ટ પદ્ધતિનો ઉપયોગ કરીને નીચેના નંબરોને સોર્ટ કરવા માટે ટ્રેસ આપો. 07

**38, 25, 64, 88, 55, 72, 98, 11**

**OR**

(એ) Explain any three array operations with suitable example. 03

(અ) એરેના કોઈપણ ત્રણ ઓપરેશનને ચોગ્ય ઉદાહરણ સાથે સમજાવો. 03

(બ) Define: Class method. Explain use of class method with suitable example. 04

(ચ) વ્યાખ્યાયિત કરો: કલાસ મેથડ. કલાસ મેથડ નો ઉપયોગ ચોગ્ય ઉદાહરણ સાથે સમજાવો. 07

(દ) Write an algorithm for bubble sort. Give the trace to sort following numbers using bubble sort method. 07

**38, 25, 64, 88, 55, 72, 98, 11**

(ક) બબલ સોર્ટ માટે અલ્ગોરિધમ લખો. બબલ સોર્ટ પદ્ધતિનો ઉપયોગ કરીને નીચેના નંબરોને સોર્ટ કરવા માટે ટ્રેસ આપો. 07

**38, 25, 64, 88, 55, 72, 98, 11**

**Q.4** (એ) Write merge sort algorithm. 03

(અ) મર્જ સોર્ટ અલ્ગોરિધમ લખો. 03

(બ) Consider the list  $I1 = [18, 25, 35, 45, 55, 68, 81, 91, 98, 100]$ . Search 81 from this list with the help of binary search and state required number of iterations. 04

(ચ) લિસ્ટ  $I1 = [18, 25, 35, 45, 55, 68, 81, 91, 98, 100]$  ને દ્યાનમાં લો. બાયનરી સર્ચ ની મદદથી આ લિસ્ટ માંથી 81 શોધો અને ઈટેશન ની આવશ્યક સંખ્યા જણાવો. 07

(દ) Define: Linked List. Write and explain following algorithms. 07

I) Insert a new node at the End in a singly linked list.

II) Delete first node from a singly linked list.

(ક) વ્યાખ્યાયિત કરો: લિંકડ લિસ્ટ. નીચેના અલ્ગોરિધમ્સ લખો અને સમજાવો. 07

I) સિંગલી લિંકડ લિસ્ટ માં અંતમાં નવો નોડ દાખલ કરવા માટેનો અલ્ગોરિધમ.

II) સિંગલી લિંકડ લિસ્ટ માંથી પહેલા નોડને ડિલિટ કરવા માટેનો અલ્ગોરિધમ.

**OR**

(એ) Differentiate: Linear Search vs Binary Search. 03

(અ) તફાવત આપો: લીનીયર સર્ચ vs બાયનરી સર્ચ. 03

(બ) Give the trace to sort following numbers using insertion sort method. 04

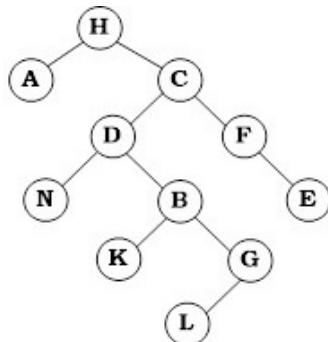
**66, 39, 35, 19, 57, 27, 20**

(ક) ઇન્સરશન સોર્ટ પદ્ધતિનો ઉપયોગ કરીને નીચેના નંબરોને સોર્ટ કરવા માટે ટ્રેસ આપો. 07

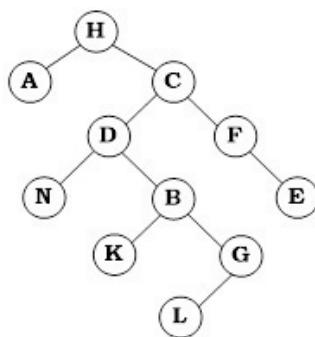
**66, 39, 35, 19, 57, 27, 20**

- (c) List out types of linked list. Write and explain following algorithms. 07  
 I) Insert a new node in the beginning of the singly linked list.  
 II) Delete the last node from a singly linked list.
- (ક) લિંકડ લિસ્ટના પ્રકારોની સૂચિ બનાવો. નીચેના અલ્ગોરિધમ્સ લખો અને સમજાવો. ૦૭  
 I) સિંગલી લિંકડ લિસ્ટ ના શરૂઆતમાં નવો નોડ દાખલ કરવા માટેનો અલ્ગોરિધમ.  
 II) સિંગલી લિંકડ લિસ્ટ માંથી છેલ્લા નોડને ડિલીટ કરવા માટેનો અલ્ગોરિધમ.

**Q.5 (a)** Traverse the nodes of following tree in pre-order, in-order and post-order. 03



- (ા) નીચેના ટ્રી ના નોડને પ્રિઓર્ડર, ઇનાર્ડર અને પોસ્ટાર્ડર માં ટ્રાવર્ઝ કરો. ૦૩



- (b) Explain Doubly Linked List with its advantages and disadvantages. 04  
 (બ) ડબલી લિન્કડ લિસ્ટ ને સમજાવો અને તેના ફાયદા અને નુકસાન જણાવો. ૦૪  
 (c) Describe: Binary Search Tree. Construct the Binary Search Tree using following data. Show each steps. 07

**32, 45, 12, 11, 13, 92, 78, 66, 17, 70, 98, 108.**

- (ક) વર્ણન કરો: બાઈનરી સર્ચ ટ્રી. નીચેના ડેટાનો ઉપયોગ કરીને બાઈનરી સર્ચ ટ્રી બનાવો. દરેક સ્ટેપ્સ બતાવો. ૦૭

**32, 45, 12, 11, 13, 92, 78, 66, 17, 70, 98, 108.**

**OR**

- (a) Define following terms: 03  
 I) Binary Tree  
 II) Internal Node  
 III) Path

|                                                                                    |           |
|------------------------------------------------------------------------------------|-----------|
| <b>(અ)</b> નીચેના શબ્દો વ્યાખ્યાયિત કરો:                                           | <b>૦૩</b> |
| I) બાઈનરી ટ્રી                                                                     |           |
| II) ઈન્ટરનલ નોડ                                                                    |           |
| III) પાથ                                                                           |           |
| <b>(બ)</b> Differentiate: Singly Linked List vs Circular Linked List.              | <b>૦૪</b> |
| <b>(બ)</b> તફાવત કરો: સિંગલી લિંક્ડ લિસ્ટ વિસ્ત સક્ર્યુલર લિંક્ડ લિસ્ટ.            | <b>૦૪</b> |
| <b>(ચ)</b> Define: Completer binary tree. Write and explain following algorithms   | <b>૦૭</b> |
| I) an algorithm to Insert a new node in binary search tree.                        |           |
| II) In-order traversal algorithm in binary search tree.                            |           |
| <b>(ઝ)</b> વ્યાખ્યાયિત કરો: કંપલેટ બાઈનરી ટ્રી. નીચેના અન્ગોરિધમ્સ લખો અને સમજાવો. | <b>૦૭</b> |
| I) બાઈનરી સર્ચ ટ્રીમાં નવો નોડ દાખલ કરવાનો અન્ગોરિધમ                               |           |
| II) બાઈનરી સર્ચ ટ્રીમાં ઇન ઓર્ડર ટ્રાવર્સલ નો અન્ગોરિધમ                            |           |

\*\*\*