--Question Starting--  
Match the following concepts in fuzzy logic with their appropriate descriptions:  
1. Concept Description  
I. Membership Functions A. Process of translating crisp values into degrees of membership  
II. Fuzzification B. Uses IF-THEN rules for decision making  
III. Fuzzy Inference C. Mathematical curves that represent how each point in the input space is mapped to a membership value  
IV. Fuzzy Control Systems D. System that modifies operations based on fuzzy logic  
Choose the correct answer from the options given below:  
(1) I-C, II-A, III-B, IV-D  
(2) I-B, II-D, III-A, IV-C  
(3) I-A, II-C, III-D, IV-B  
(4) I-D, II-B, III-C, IV-A  
Answer Key: 1   
Solution:  
• Membership Functions: These are mathematical curves that determine how each input value corresponds to a degree of membership in a fuzzy set.  
• Fuzzification: This is the process where crisp numbers are converted into fuzzy values based on the degree of membership defined by membership functions.  
• Fuzzy Inference: This system uses a set of fuzzy rules, typically in the form of IF-THEN statements, to derive conclusions from fuzzy inputs.  
• Fuzzy Control Systems: These systems apply fuzzy logic to control mechanisms, adjusting outputs based on fuzzy inputs and rules.  
Hence, Option (1) is the right answer.  
  
--Question Starting--  
Match the following elements of C++ programming with their correct functionalities:  
1. Element Functionality  
I. Virtual Functions A. Allows objects to be initialized when they are created  
II. Constructors B. Enables runtime polymorphism  
III. Overloading C. Facilitates the use of an operator or function in different ways  
IV. Templates D. Allows generic programming by letting classes and functions operate with generic types  
Choose the correct answer from the options given below:  
(1) I-B, II-A, III-C, IV-D  
(2) I-A, II-D, III-B, IV-C  
(3) I-C, II-B, III-D, IV-A  
(4) I-D, II-C, III-A, IV-B  
Answer Key: 1   
Solution:  
• Virtual Functions: These are used in base classes to ensure that the correct function is called for an object, regardless of the type of reference (or pointer) used for function call, thus supporting runtime polymorphism.  
• Constructors: These are special member functions of classes that are executed whenever a new object of that class is created, primarily to initialize variables.  
• Overloading: This feature allows functions or operators to perform differently based on the parameters passed to them, enhancing the flexibility of the language.  
• Templates: Templates support generic programming, allowing functions and classes to operate on generic types, which helps in writing code applicable to any data type.  
Hence, Option (1) is the right answer.  
  
--Question Starting--  
Match the following components of distributed systems with their corresponding attributes:  
1. Component Attribute  
I. Network Structure A. Concerns the stability and continuity of services  
II. Distributed File Systems B. Handles the layout and physical components of the network  
III. Design Issues C. Manages data storage across multiple machines  
IV. Robustness D. Includes considerations like scalability, security, and fault tolerance  
Choose the correct answer from the options given below:  
(1) I-D, II-B, III-C, IV-A  
(2) I-B, II-C, III-D, IV-A  
(3) I-A, II-D, III-B, IV-C  
(4) I-C, II-A, III-B, IV-D  
Answer Key: 1   
Solution:  
• Network Structure: This involves the physical and logical layout of the network, influencing how components communicate and are interconnected.  
• Distributed File Systems: These systems manage the storage and retrieval of data across multiple physical locations, ensuring consistency and reliability.  
• Design Issues: These are critical considerations in distributed systems, including factors like scalability, security, and how the system handles failures.  
• Robustness: This attribute pertains to the system's ability to remain stable and continue functioning in the face of failures or unexpected conditions.  
Hence, Option (1) is the right answer.  
  
--Question Starting--  
Match the following topics related to software requirements with their correct definitions:  
1. Topic Definition  
I. Eliciting Requirements A. Document detailing the functions and environment of a system  
II. SRS Document B. Techniques and practices for gathering requirements from stakeholders  
III. Use Cases C. Describes typical user interactions and system responses  
IV. Requirement Analysis D. Critical analysis to determine stakeholder needs and ensure alignment with system capabilities  
Choose the correct answer from the options given below:  
(1) I-B, II-A, III-C, IV-D  
(2) I-C, II-D, III-A, IV-B  
(3) I-A, II-C, III-B, IV-D  
(4) I-D, II-B, III-A, IV-C  
Answer Key: 1  
Solution:  
• Eliciting Requirements: This involves the use of various techniques to gather clear and concise requirements from stakeholders and users.  
• SRS Document: This is a formal document that outlines everything the software will do and how it will perform in various environments.  
• Use Cases: These provide a description of how users will interact with the system and how the system responds, facilitating a better understanding of the system's functional requirements.  
• Requirement Analysis: This process involves a detailed examination of the requirements to ensure they are feasible, necessary, and aligned with the business goals.  
Hence, Option (1) is the right answer.  
  
--Question Starting--  
Match the following aspects of semantic analysis with their correct functionalities:  
1. Aspect Functionality  
I. Attribute Grammar A. Defines how attribute values are obtained  
II. Syntax Directed Definitions B. Provides a framework for generating values based on syntax trees  
III. Inherited Attributes C. Attributes that are passed down from parent nodes to child nodes in a syntax tree  
IV. Type-Checking D. Ensures that operations in a program are applied to compatible data types  
Choose the correct answer from the options given below:  
(1) I-A, II-B, III-C, IV-D  
(2) I-C, II-D, III-A, IV-B  
(3) I-B, II-A, III-D, IV-C  
(4) I-D, II-B, III-A, IV-C  
Answer Key: 1  
Solution:  
• Attribute Grammar: This is a formal way to define attributes associated with programming language constructs, specifying how to derive attribute values from syntax trees.  
• Syntax Directed Definitions: These are rules that define how values of attributes are computed based on the structure of the parse tree.  
• Inherited Attributes: These are attributes whose values are passed from parent nodes to child nodes, helping to determine contextual conditions.  
• Type-Checking: This process involves verifying that the types of operands in expressions are operationally compatible, preventing type errors during execution.  
Hence, Option (1) is the right answer.