42 Lecture - CS201

Important Mcqs

1. What is a class template in C++?

- a. A class that is used to create multiple instances of a class
- b. A class that can be used with multiple data types
- c. A class that is used to define a single data type
- d. A class that cannot be instantiated

Answer: b

2. How do you declare a class template in C++?

a. By using the keyword "class" followed by the template parameter list and the class declaration

b. By using the keyword "template" followed by the template parameter list and the class declaration

c. By using the keyword "typedef" followed by the template parameter list and the class declaration

d. By using the keyword "typename" followed by the template parameter list and the class declaration

Answer: b

3. What is the purpose of a template parameter in a class template?

- a. To represent a data type that can be used by the class
- b. To represent a constant value that can be used by the class
- c. To define the size of the class
- d. To determine the access level of the class members

Answer: a

4. How do you instantiate a class template in C++?

- a. By defining a new class with the same template parameters as the original class
- b. By creating an object of the class with the desired data type as the template argument
- c. By using the keyword "using" followed by the name of the class template
- d. By using the keyword "typename" followed by the name of the class template

Answer: b

5. How does template specialization work in class templates?

- a. It allows you to create a specialized version of the class for a specific data type or value
- b. It allows you to define a new template parameter for the class
- c. It allows you to restrict the data types that can be used with the class
- d. It allows you to override the default implementation of the class

Answer: a

6. How do you define member functions for a class template in C++?

- a. By defining the functions outside the class definition
- b. By defining the functions inside the class definition
- c. By using the keyword "template" before the function definition
- d. By using the keyword "typename" before the function definition

Answer: b

7. How do you overload a class template in C++?

- a. By defining a new class with the same name but different template parameters
- b. By defining a new member function with the same name but different template parameters
- c. By defining a new member function with a different name but the same template parameters
- d. By defining a new class with a different name but the same template parameters

Answer: b

8. What are the advantages of using class templates in C++?

- a. They provide code reusability and improve code quality
- b. They reduce development time and cost
- c. They allow for generic programming and flexible data structures
- d. All of the above

Answer: d

9. What are the potential drawbacks of using class templates in C++?

- a. They can lead to longer compilation times
- b. They can be difficult to understand for novice programmers
- c. They can be prone to errors and bugs
- d. All of the above

<mark>Answer: d</mark>

10. Can class templates be used with user-defined data types?

- a. Yes, as long as the data type is specified as a template parameter
- b. No, class templates can only be used with built-in data types
- c. It depends on the complexity of the user-defined data type
- d. It depends on the implementation of the class template

Answer: a