35 Lecture - CS304

Important Mcqs

1. What are member templates in C++?

- A. Templates defined inside a class or struct
- B. Templates defined outside a class or struct
- C. Templates that cannot be used with any data type
- D. Templates that can only be used with integer data types

Answer: A. Templates defined inside a class or struct.

What is the advantage of using member templates?

- A. Increased reusability and adaptability of code
- B. Improved code performance
- C. Reduced code complexity
- D. All of the above

Answer: D. All of the above.

Can member templates access the data members and methods of the class they are defined in?

A. Yes

B. No

Answer: A. Yes.

Can member templates be used to provide generic constructors?

A. Yes

B. No

Answer: A. Yes.

What is the syntax for defining a member template?

A. template<class T> void myFunction(T arg);

B. template<typename T> void myFunction(T arg);

C. template<class T> struct MyClass {...};

D. All of the above

Answer: D. All of the above.

Can member templates be specialized for specific data types?

A. Yes

B. No

Answer: A. Yes.

What is the purpose of a member function template?

- A. To provide a generic member function that can work with any data type
- B. To provide a specialized member function for a specific data type
- C. To provide a constructor for a class or struct
- D. None of the above

Answer: A. To provide a generic member function that can work with any data type.

What is the advantage of using a member function template over a regular member

function?

- A. Increased reusability and adaptability of code
- B. Improved code performance
- C. Reduced code complexity
- D. All of the above

Answer: D. All of the above.

What is the difference between a member function template and a regular member function?

- A. A member function template can work with any data type, whereas a regular member function can only work with specific data types
- B. A member function template is defined inside a class or struct, whereas a regular member function is defined outside the class or struct
- C. A member function template cannot access the data members and methods of the class it is defined in, whereas a regular member function can
- D. None of the above

Answer: A. A member function template can work with any data type, whereas a regular member function can only work with specific data types.

What is the purpose of template specialization?

- A. To provide a generic template that can work with any data type
- B. To provide a specialized template for a specific data type
- C. To provide a constructor for a class or struct
- D. None of the above

Answer: B. To provide a specialized template for a specific data type.