# 27 Lecture - CS304

## Important Mcqs

1. **Which of the following is a restriction of specialization in C++?**
   A) You can partially specialize function templates
   B) You can specialize function templates for built-in types
   C) You can specialize function templates for any type
   D) Specialization can lead to code duplication and maintenance issues
   Answer: B

   **What is specialization in C++?**
   A) A mechanism that allows programmers to define a different implementation of a template or function for a specific set of arguments
   B) A way to restrict access to certain parts of a program
   C) A technique used to improve the performance of a program
   D) None of the above
   Answer: A

   **Which of the following is not a restriction of specialization in C++?**
   A) You cannot partially specialize function templates
   B) You cannot specialize function templates for built-in types
   C) You cannot specialize function templates for any type
   D) None of the above
   Answer: D

   **When is specialization useful in C++?**
   A) When the default behavior of a template or function is not suitable for a particular data type or value
   B) When you want to restrict access to certain parts of a program
   C) When you want to improve the performance of a program
   D) None of the above
   Answer: A

   **Can you partially specialize function templates in C++?**
   A) Yes
   B) No
   Answer: B

   **Can you specialize function templates for a built-in type such as int or double in C++?**
   A) Yes
   B) No
   Answer: B

   **What are some restrictions of specialization in C++?**
   A) You can partially specialize function templates
   B) You can specialize function templates for any type

C) Specialization can lead to code duplication and maintenance issues
D) None of the above
Answer: C

**How does specialization help in C++?**
A) By allowing programmers to define a different implementation of a template or function for a specific set of arguments
B) By restricting access to certain parts of a program
C) By improving the performance of a program
D) None of the above
Answer: A

**What is the syntax for specialization in C++?**
A) template <> function_name<>(){}
B) template <> function_name<>{}()
C) template <typename T> function_name<T>(){}
D) None of the above
Answer: A

**Is overuse of specialization a good practice in C++?**
A) Yes
B) No
Answer: B