

32 Lecture - CS304

Important Subjective

1. What is generic programming?

Answer: Generic programming is a programming paradigm that emphasizes writing reusable code that can be used with different data types.

What is a template in C++?

Answer: A template is a mechanism in C++ that allows generic programming. Templates define a blueprint for a class or function that can be used with different data types.

How do templates work in C++?

Answer: Templates work by defining a generic class or function that can be used with different data types. The template is instantiated with a specific data type when it is used in code.

What are the advantages of generic programming?

Answer: The advantages of generic programming include increased code reusability, improved code maintainability, and reduced code complexity.

What is the difference between generic programming and object-oriented programming?

Answer: Generic programming focuses on writing reusable code that can be used with different data types, while object-oriented programming emphasizes encapsulation, inheritance, and polymorphism.

What is a generic algorithm?

Answer: A generic algorithm is an algorithm that is written to work with different data types. The algorithm is typically written as a function or class template.

What is type erasure in Java?

Answer: Type erasure is a process in Java where the generic type information is removed from a generic class or method during compilation. This is done to maintain backward compatibility with older Java code.

What is a type parameter in Java generics?

Answer: A type parameter in Java generics is a placeholder for a specific data type that is used by a generic class or method.

What is the syntax for defining a generic class in C++?

Answer: The syntax for defining a generic class in C++ is:

```
arduino
Copy code
template<typename T>
class MyClass {
    // Class definition here
};
```

What is the syntax for defining a generic method in Java?

Answer: The syntax for defining a generic method in Java is:

typescript

Copy code

```
public <T> void myMethod(T myParam) {  
    // Method code here  
}
```