37 Lecture - CS201

Important Subjective

1. **What is the purpose of overloading insertion and extraction operators**?
   Answer: Overloading these operators enables custom input and output of user-defined types.
2. **How do you overload the insertion operator in C++?**
   Answer: The insertion operator is overloaded using the syntax: ostream& operator<<(ostream& os, const Object& obj)
3. **How do you overload the extraction operator in C++?**
   Answer: The extraction operator is overloaded using the syntax: istream& operator>>(istream& is, Object& obj)
4. **What is the return type of the overloaded insertion operator?**
   Answer: The return type of the overloaded insertion operator is ostream&.
5. **What is the return type of the overloaded extraction operator?**
   Answer: The return type of the overloaded extraction operator is istream&.
6. **How do you define an insertion operator for a class in C++?**
   Answer: You define an insertion operator for a class in C++ by declaring a friend function of the class.
7. **How do you define an extraction operator for a class in C++?**
   Answer: You define an extraction operator for a class in C++ by declaring a friend function of the class.
8. **Can you overload the insertion and extraction operators for built-in types in C++?**
   Answer: No, these operators cannot be overloaded for built-in types in C++.
9. **What is the purpose of the 'const' keyword in the insertion operator's parameter list?**
   Answer: The 'const' keyword in the insertion operator's parameter list specifies that the object being inserted should not be modified.
10. **What is the purpose of the '&' symbol in the insertion and extraction operator's parameter list?**
    Answer: The '&' symbol in the insertion and extraction operator's parameter list specifies that the parameters should be passed by reference.