1. Describe in general the types of polymorphism in C++.

Answer: There are four types of polymorphism in C++ :

The first type is subtype polymorphism (runtime polymorphism)- The polymorphism which allows the user to use derived classes through base class pointers and references.

The second type of polymorphism is Parametric polymorphism (compile-time polymorphism)- The polymorphism which allows the user a means to execute the same code for any type.

The third type of polymorphism is Ad-hoc polymorphism (overloading)- The polymorphism which allows the user to use functions with the same name act differently for each type.

The fourth type of polymorphism is Coercion (implicit or explicit casting) - The polymorphism which allows the user to use Coercion which happens when an object or a primitive is cast into another object type or primitive type.

1. Explain the differences and similarities between parametric polymorphism and adhoc polymorphism.

Answer: Similarity : The similarity is that they both use alogorithm for different types of arguments.

Difference: The difference is that parameteric polymorphic functions use one algorithm to operate on arguments of many different types, whereas overloaded functions may use a different algorithm for each type of argument.

1. Why should templated code be implemented in a header file?

Answer : Because header files are the only portable solution for a convenient use