# Lambda functions

## What are lambda functions and their benefits over normal functions?

<https://chat.openai.com/share/79a04d8a-f3b7-43b4-a510-2be3c2b42883>

## How are lambda functions used with STLs?

<https://chat.openai.com/share/134b2ba8-a5f0-4438-9751-c2ea19f4ad10>

# Try Throw and catch

# Abstraction:

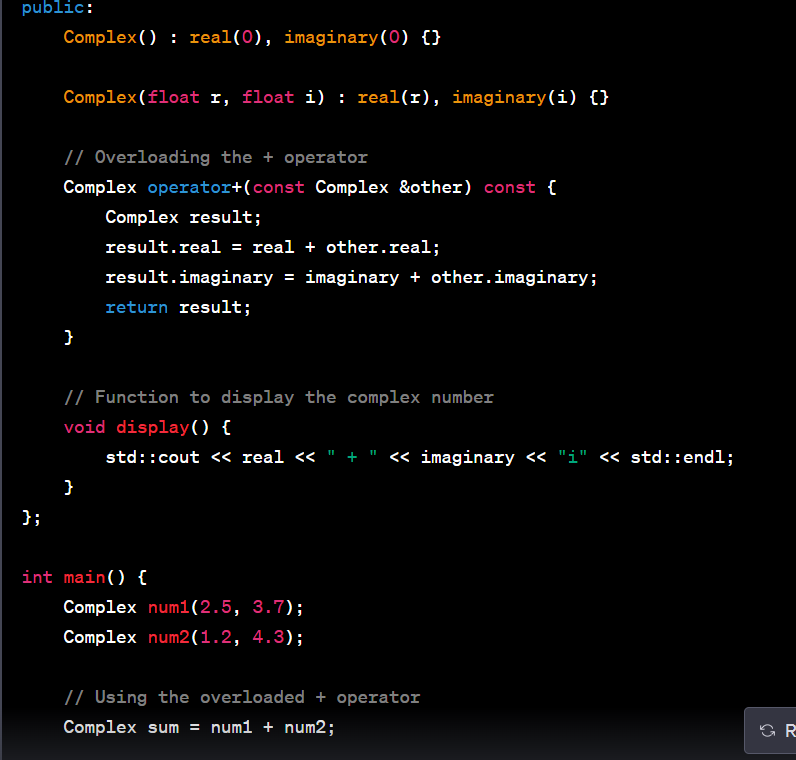
Abstraction is hiding unimportant data. For example keeping only the add() function in a class. Example: TV. While watching TV we only need to know functions such as volume. We do not need to know cathode ray tube

# Encapsulation:

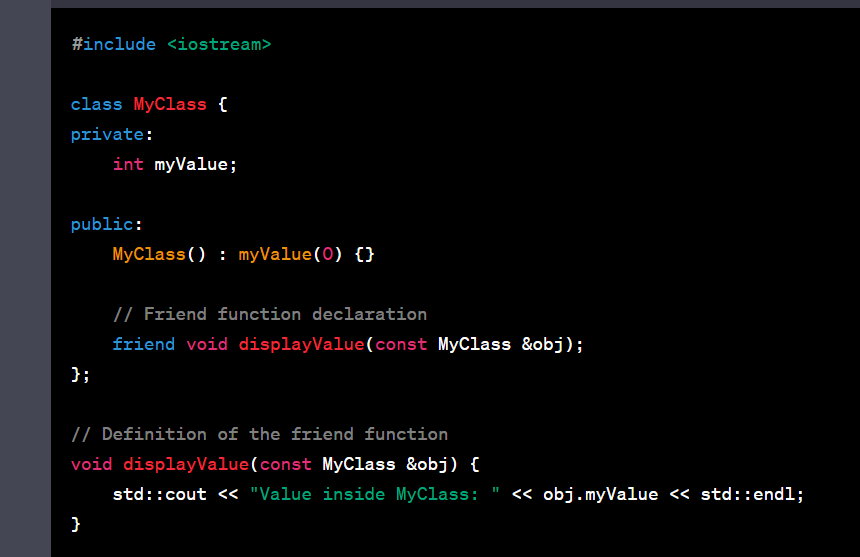
Encapsulation is the bundling of data (attributes) and methods (functions) that operate on the data into a single unit known as a class. It restricts access to some of an object's components and prevents the accidental modification of data. Example: Encryption

# Polymorphism:

# Operator Overloading:



# Friend Function:



# Smart Pointers:

<https://www.geeksforgeeks.org/auto_ptr-unique_ptr-shared_ptr-weak_ptr-in-cpp/>

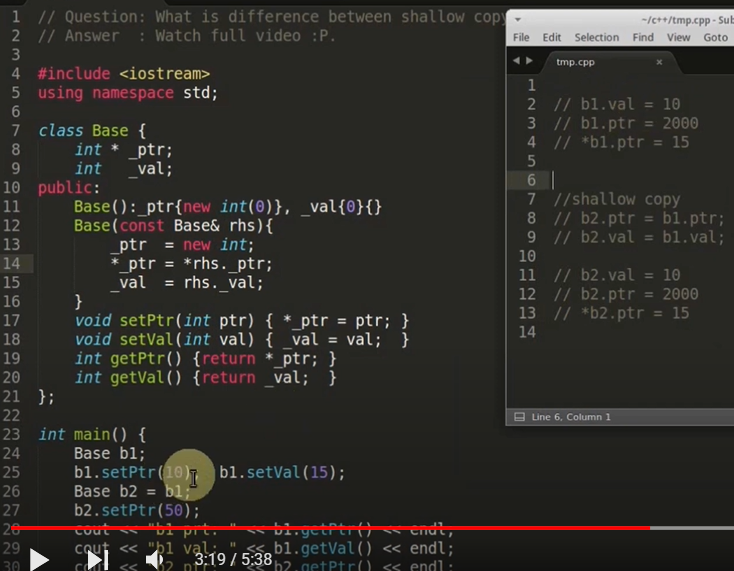
# Threads

# Shallow and Deep Copy:

Shallow copy is the default copy done by the compiler. In shallow copy the pointer is copied straightway and both the objects point to the same address. The issue happens when the latter object is modified.

In deep copy we first dynamically create the new space and copy all the data to this new space.

<https://www.youtube.com/watch?v=TjN_5rGMpfg>



# Diamond problem, lazy initialization, initializer list

<https://chat.openai.com/share/62d4548e-8e64-4696-8006-569e38eaa8ae>

# Move constructors, rvalue reference

<https://www.internalpointers.com/post/c-rvalue-references-and-move-semantics-beginners>

# How is C++ code compiled?

# Some advanced questions in C++

<https://www.mytectra.com/interview-question/top-advanced-c-programming-interview-questions-2017>

<https://www.interviewkickstart.com/interview-questions/cpp-interview-questions-for-experienced-developers>

WHats left

Threading

Design Patterns

Python revision

Basic Data Structures and Algorithms

New C++ features

Markov property question

FX specific things