ForEachLevel:
The ForEachLevel function allows iterating over each price level in the order book and executing a provided function f on each level. The function f takes arguments representing the bid and ask prices, quantities, and counts for each level. The function ForEachLevel iterates through both the buy and sell sides of the order book simultaneously, comparing the prices of the current levels on each side. It calls the provided function f with the relevant level information until either one of the sides reaches its end or the provided function returns false, indicating to stop the iteration.

ForEachOrder:
The ForEachOrder function allows iterating over the orders in the order book either from the innermost levels to the outermost levels or vice versa, depending on the values of the is_buy and inner_to_outer parameters. The function f is a provided function that takes arguments representing the order's buy/sell status, price, quantities, and order ID. The function ForEachOrder iterates through the levels on the corresponding side (buy or sell) of the order book and iterates over the orders within each level. It calls the provided function f with the relevant order information for each order.

UncrossBookSide:
The UncrossBookSide function is used to "uncross" the order book by removing any offensive orders from a particular side of the book. The offensive orders are determined based on the provided is_buy parameter. If is_buy is true, it means that the sell side is newer, and any offensive orders on the buy side should be removed. Conversely, if is_buy is false, it means that the buy side is newer, and any offensive orders on the sell side should be removed. The function iterates through the relevant side of the order book (buy or sell) and compares the prices of the levels on each side. If a level's price on the other side is less than or equal (in the case of buy side) or greater than or equal (in the case of sell side), the function collects the order IDs of the orders in that level and stores them in a list called stale_orders. Finally, the function processes delete operations for each order in stale_orders by calling the ProcessDelete function.

ProcessExec:
The ProcessExec function is used to process order executions. It takes an order ID and the executed quantity as parameters. The function finds the corresponding order in the orders_ map using the order ID. If the order is found, it compares the executed quantity with the order quantity. If the executed quantity is less than the order quantity, the function updates the order's quantity to reflect the remaining quantity and calls the ProcessReplace function to update the order in the order book. If the executed quantity is equal to the order quantity, the function calls the ProcessDelete function to remove the order from the order book. If the executed quantity exceeds the order quantity, an exception is thrown, indicating an error in the execution process.