

BofA Assignment

This is a test the Bank of America gave to an applicant for a programming job. It has been modified a bit. The applicant was given 3 hours to complete the test and return the code. You shall treat this as if it was a real-time interview assignment and try to finish it within the given time frame.

1. Make an inheritance hierarchy with at least two classes representing mathematical functions (e.g. $y = f(x)$) that take 1 double as input and return 1 double as output. Each class should provide a method that evaluates $f(x)$, and any other methods you find appropriate or necessary to solve the rest of the test or to make the classes useful in practice.
 - (a) One class should implement the function $y = a \cdot x$, where a is constant for each instance of the class.
 - (b) One class should implement the function $y = x \cdot x$.
 - (c) You may add any other classes or methods you think will make it easier for other people to use or extend your inheritance hierarchy.
2. Now, make a new class called *SumOfFunctions* that holds a vector of pointers to objects from the Function classes you wrote for problem 1. The pointers stored in the SumOfFunctions class will include both pointers to your $x \cdot x$ class, pointers to $a \cdot x$ objects with different values of a , and pointers to new derived classes added to your hierarchy by other programmers. Your SumOfFunctions class will be used frequently in an STL container. Give it the following methods:
 - a method that takes x as input and computes the sum of the $f(x)$ values for each Function in the array.
 - a method that adds a copy of a Function object to the array.
 - any other methods you find appropriate or necessary to make your class reliable and convenient, or to use in an STL container.
3. Your classes above are only one possible implementation of a mathematical function as a C++ class. Make a new templated C++ function that computes the difference $f(x) - g(x)$ between two functions $f(x)$ and $g(x)$ represented as classes, e.g. these classes could be your Function class from problem 1, your SumOfFunctions class from problem 2, or a completely different class that also has a method to compute $y = f(x)$ but which does not inherit from any other class. Your function should take the two objects f and g and the point x as input.