Introduction to C++: COMP 322 (Winter 2014) - Practice questions

1. What is the difference between a *has-a* relationship and an *is-a* relationship? What does that have to do with inheritance in C++?
2. What are some of the benefits of defining objects?
3. What are some benefits of inheritance?
4. Assume you have a `vector<int>`, write code to calculate the number of values that are positive.
5. How do you invoke a base member function from a derived class in which you have not overridden that function?
6. How do you invoke a base member function from a derived class in which you've overridden that function?

F. What is an inclusion guard?

7. What is a friend function?
8. What is the use of the default constructor?

O. Explain how the garbage collector works in C++.

9. Explain what the purpose of throwing an exception would be.
10. What is the difference between the compiler and the preprocessor?
11. What is a try block?

O. Explain the difference between a `HashMap` and `HashTable`.

12. What are the differences between the function prototype and the function definition?
13. What is the `protected` keyword used for?
14. When is the destructor called?
15. What is the `this` pointer?

L. Explain the `with` statement and its usage in C++.

16. (Long, but useful practice) Define a templated type `MyVector`. Your templated type should maintain a collection of any amount. It should do this by maintaining a regular array with empty spots in it. Only certain spots will be considered part of the actual array. When the array is full, and you try to `push_back` an element, your class should double the size of the array and copy elements from the old array to the new one. If someone tries to get an element that is outside the range of the vector you are representing but inside the array you are storing, you should throw an exception.