Introduction to C++ - Quiz 2, 2 April 2012

Name: ____ ID: ______________

This is a *closed book* quiz. There are six questions on two pages, for a total of 20 possible points.

In the code fragments, assume that the context (e.g. appropriate header files and `using namespace` statements) has been specified correctly.

1. What is the difference between the following three method headers that are part of a class `A`? Your answer should include a brief description (2 or 3 sentences) of *why* someone would choose one approach over another (5 points)

```
int foo();
virtual int foo();
virtual int foo() = 0;
```

2. If class Fruit is a base class for class Apple, what will happen if the following code is entered and run? Note that the code may or may not compile. Give a short (2 or 3 sentences) explanation for your answer. (4 points)

(A)
```
Apple b;
Fruit a = b;
```

(B)
```
Fruit* a = new Apple();
```

3. What is the difference between making an attribute of a class `private` `public` or `protected`? Why is it considered bad practice to make an attribute `public`? (2 points)

4. Suppose you did not know about the standard library types for iterators and wanted to define an iterator type `IntVectorIterator` that would work on `vector<int>`. What properties would your `IntVectorIterator` need to maintain at any given time to successfully iterate over a vector? Note that you should be able to iterate forward and backwards by arbitrary amounts. Keep in mind you can not use the type `vector<int>::iterator` or any related type. List the names of the variables along with the types. (3 points)

5. Assuming that you defined adding something to your vector as moving forward across the vector, write a method to overload the += operator. This operator should adjust the current iterator by x. (It should be like calling ++ several times) This code will depend on your choices in the prior question and you should update the appropriate member variables/properties. The header and return statements are given to you. (3 points)

```
IntVectorIterator& operator+=(int x)
{
        //insert code here to update the position of the iterator by x













         return (*this);
}
```

6. Suppose I have defined an object FavoriteNumberRecord. An FavoriteNumberRecord object contains two properties: a string representing the name of a person and an int representing the person's favorite number, which can be any integer. I want to write a function getFavoriteNumberOf which takes as input a string name and a vector<FavoriteNumberRecord> and returns the favorite number of the person with name name. It is possible, however, that there will be no FavoriteNumberRecord with corresponding to the name name. How can I use the concept of exceptions to help transfer this information to the calling function? (3 points)