## 2012.2 Object-Oriented Programming and Design Midterm Exam (Oct. 25th 7pm-8:20pm)

supervisor	
signature	

StudentID#: ( ), Name: ( )

* You may answer in either Korean or English.	
<ol> <li>(16 points) Fill out the blanks (a)~(d) with the most appropriate words.</li> <li>One of the main benefits of data encapsulation is implementation independence.         Implementation independence means (a.         ) in implementation should not affect (b. by clients of a class.     </li> </ol>	) seen
• One of the formal definitions of software engineering is "the establishment and use of sound engineering princ (c. ) obtain software that is reliable and works efficiently".	ciples in order to
• Abstract class is a class that contains at least one (d. ).	
2. (12 points) (1) It is often desirable to avoid passing large objects by value. Why? Explain.	)
(2) What is dependability? Explain in detail.	)
3. (15 points) (1) Explain the main difference between 'class' and 'object' in C++.	)
(2) Explain the main difference between const member function and a general member function in a class? contain the meaning of const member function.	Your answer should
(3) What is the main benefits of using const member function. Explain.	)
4. (12 points) (1) If you need to write a copy constructor, you also need to write a destructor. Why?	)
(2) If shallow copies are OK, you don't need to write a copy constructor. Why? Explain in detail.	)
<ul><li>5. (15points) This problem is about a function assert(x).</li><li>(1) Explain what should be put into parameter x in function assert(x).</li><li>(</li></ul>	)
(2) What does function assert(x) do? Explain	)
(3) Explain why using function assert(x) is useful.	)

and the limit it can represent is maximized. In (1), (2), and (3), Do not call other member functions. Instead, You may use standard library in your code. Your code should manage the memory correctly and efficiently. Your code should be grammatically and logically correct. class inf int { private: // points to a string of digits. Perform dynamic allocation when necessary. char\* digits; unsigned int length; // stores the number of actual digits bool thesign; // we assume that thesign is false if negative integer, true otherwise. // ex) 1531111111111111 : digits -> "1111111111111111351", length=17, thesign=true; // ex) -12345555555555 : digits -> "5555555554321" , length=14, thesign=false public : inf int(); // constructor. assign 0 as a default value. int\_int(int n); // constructor. the input integer n is converted to inf int format. inf int(const inf int& x); // copy constructor // other member functions should be here but they are just not shown. }; (1) Write C++ code for following constructor. (3) Write C++ code for following constructor. inf int::inf int() // assign 0 as a default value inf int::inf int(int n) (2) Write C++ code for following constructor. inf\_int::inf\_int(const inf\_int& x) // copy constructor

}

6. (30points) In our Project #2, class inf int is defined as follows. It can represent an arbitrarily big integer number in practice