2009.2 Object-Oriented Programming and Design
Final Exam (Dec. 21th 7pm-8:30pm)

StudentID# : ( ) , Name : ( )

* You may answer in either Korean or English

1. (3points) Complete following sentences by correctly filling out the blanks with English words.
(1) In STL, ( ) is a holder that stores a collection of other objects.
They are implemented as ( ), which allows a great flexibility in the types supported as elements.

(2) In STL, a range is any sequence of objects that can be accessed through ( ).

(3) UML describes a software system at a higher level of ( ).

(4) In a package diagram, packages are usually organized to maximize ( ) within each package and to minimize ( ) among packages.

2. (3points)
(1) What is the most important difference between an array and an STL vector? In your answer, you have to describe the main benefit of using the STL vector compared to using an array.
( )

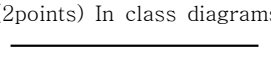
(2) What is the meaning of "system"? Explain in detail with <u>one sentence.</u>
( )

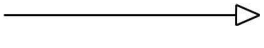
(3) What is the main purpose of using "use-case diagram"?
( )

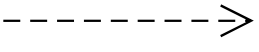
3. (3points)
(1) In STL, what are main similarities between a vector and a deque? List two.
(a.)

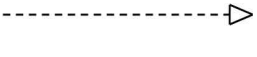
(b.)

(2) In STL, what are main difference between a vector and a deque? List at least one.
( )

4. (2points) In class diagrams, what are the meanings of following lines or arrows? Fill in each blank with <u>just one word</u>.
(1)  : ( )

(2)  : ( )

(3)  : ( )

(4)  : ( )

5. (3points) (1) What is "Copy-On-Write" for class implementation? <u>Explain in detail</u>.
( )

(2) Why "Copy-On-Write" is useful? <u>Explain in detail</u>.
( )

6. (6points) Insert C++ codes for generic implementation of the **display function** below. Assume all necessary header files are already included.

```
// Put your code here for generic implementation of the "display" function. Only one display function should be defined.
```

```
void main()
{
    list<int> li;
    vector<double> vs;
    li.push_back(3);  li.push_back(7);   li.push_back(10);
    vs.push_back(5.21); vs.push_back(2.1); vs.push_back(5.3);
    cout<< "li: "; display(li.begin(), li.end()); cout<< endl;
    cout<< "vs: "; display(vs.begin(), vs.end()); cout<< endl;
}
```

Execution Output result:

```
li: 3 7 10
vs: 5.21 2.1 5.3
```

7. (10points) Consider following C++ code and its execution output result.

```
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;

class Student
{
public :
    Student(int id , char* name)
    {
        stu_id = id;
        stu_name = name;
    }
    (a)

    char* getName() { return stu_name; }
    int getID() { return stu_id; }

private :
    int stu_id;         // student id
    char* stu_name;     // student name
};
```

```
int main()
{
        vector<Student> stu_vec;

        stu_vec.push_back(Student(4,"Nancy"));
        stu_vec.push_back(Student(1,"Tom"));
        stu_vec.push_back(Student(3,"Mike"));
        stu_vec.push_back(Student(2,"Lisa"));

        sort(stu_vec.begin() , stu_vec.end()); // sort by student id

        (b)
        // code for printing out each element (student id , name) of "stu_vec"

        return 0;
}
```

Execution Output result:

```
1 : Tom
2 : Lisa
3 : Mike
4 : Nancy
```

(1) (5points) What code should be inserted in (a) ?

```
//Put your code here.
```

(2) (5points) What code should be inserted in (b) ?

```
//Put your code here for printing out each element [student id : name] of "stu_vec".
```