BLG252E – Object Oriented Programming
FINAL EXAM
02.01.2014
Duration: 2 hours , Books and notes closed.

QUESTION 1) [25 points] Suppose you are given the definitions below:

```
string      s[] = {"Apple", "Orange", "Grape", "Cherry", "Mango"};
int         a[] = {10,20,30,40};
class Complex { public: int re, im; };
Complex  c[] = { {-2,4}, {5,-6}, {0,3} };
```

a) [15 points] Write C++ codes to implement the class **MyIterator**. It should be a **template class** which will operate on different types of data arrays such as the given examples.

Member data:

- An array of generic data type. (Maximum array size 20).
- Number of elements in the array.
- Index of current element.

Member functions:

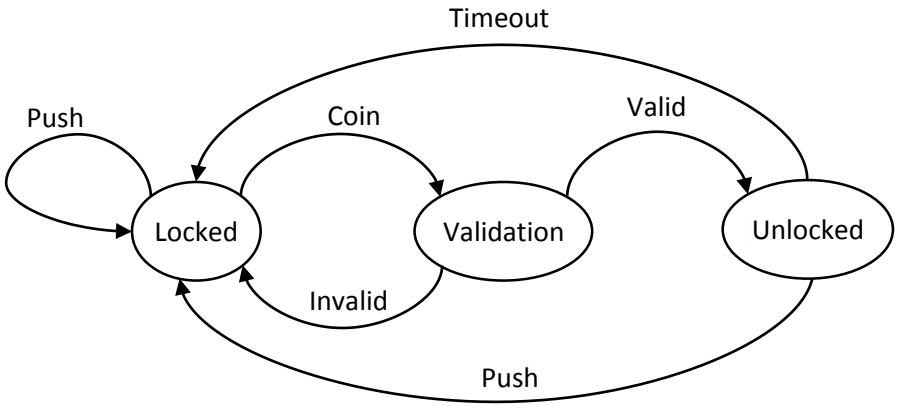
- Constructor : It takes two input arguments: An array of data (generic type) , and the number of elements (integer) in the array. The function initializes the current index to zero.
- hasNext() function: It does not take any input arguments. By checking the current index, the function returns true if still there are elements to access, or returns false if the index reaches beyond the end.
- getNext() function: It does not take any input arguments. The function returns the element of array which is at the current index.

b) [10 points] Write a main() program to do the followings:

i. Declare three MyIterator objects, one for each data types given above (string, int, and Complex).
ii. Loop thru each object and display the elements on the screen.

QUESTION 2) [25 points] A turnstile is a finite state automaton used to control entrance to places such as a metro gate. The following is a state transition diagram for a turnstile. Each state is represented by a circle. Arrows show the transitions from one state to another. Each arrow is labeled with the event that triggers that transition.

- Initially the turnstile is at the Locked state, preventing persons from passing through.
- Depositing a coin (metal money) in a slot on the turnstile changes the state to the Validation state.
- If the coin is valid (having enough money), the state changes to the Unlocked state, allowing a person to push through the turnstile.
- After the person passes through or a timeout duration happens, the turnstile is locked again until another coin is inserted.



Assume the following definitions are given:

```
enum { LOCKED, VALIDATION, UNLOCKED, MAXSTATES };
enum { COIN, PUSH, VALID, INVALID, TIMEOUT, MAXEVENTS };
string states[MAXSTATES] = {"LOCKED", "VALIDATION", "UNLOCKED"};
string events[MAXEVENTS] = {"COIN", "PUSH", "VALID", "INVALID", "TIMEOUT"};
```

a) [20 points] Write C++ codes to implement the class **Turnstile** that contains the current_state as integer member data and the following member functions:

- Default constructor : Initializes the current_state to LOCKED.
- process() function: It takes one input argument: An event code. First, the function displays the current state and the given event. Then, it changes the current_state to a new state based on the given event code. The function should check if the given event is applicable to the current_state. The whole function should be written with a **try-catch** block. If the given event code is not applicable to the current state, then the function must **throw** an exception. An error message should be displayed for an unapplicable event.

b) [5 points] Write a main() program to perform simulation of a turnstile : Declare a Turnstile object T, then in an infinite loop, generate a random event and process it.

QUESTION 3) [50 points] Consider the entity specifications given below.

- Publication : It represents a general entity. A publication has a Title and a Price.
- Book : It is a kind of Publication. A book has an ISBN number.
- Magazine : It is a kind of Publication. A magazine has a Period code (W=weekly, M=monthly).
- Journal : It is a kind of both Book and Magazine. A journal has a Subject name.
- Collection : A collection has an Owner name, and contains several publications (books, magazines, journals).

a) [10 points] Draw the UML class diagram for these specifications. Use inheritance and association (aggregation) where necessary.

b) [30 points] Write C++ implementation codes for the classes.

The Publication, Book, Magazine, and Journal classes should contain their own member functions as described below:

- Default constructor : It should ask the user to enter all of the related member data from keyboard.
- print() function : It should display all of the related member data.

The Collection class should be implemented as described below:

- In addition to owner name, the class should also contain a **STL vector** of pointers to Publications.
- Parameterized constructor: It takes one input argument : The owner name.
- add() function : It takes one input argument : A pointer to a Publication object. The given pointer is added to the vector.
- print() function: It displays the owner name. The function also loops through the elements of vector, and invokes the print() function of related objects. If the vector is empty, the function displays a warning message.

c) [10 points] Write a main() program to do the followings:

i. Declare a Collection object K , with its name as “İTÜ Library”.
ii. Repeat the loop below, until user chooses to exit:
    - Display operation choices to the user for data inputting, and read the user response. (B=book, M=magazine, J=journal, E=exit)
    - If the response is not exit, dynamically allocate a specified object and add it to K.
iii. Print K.