National Exams May 2016

04-Soft-A6, Software Quality Assurance

3 hours duration

NOTES:

1. If doubt exists as to the interpretation of any question, the candidate is urged to submit with the answer paper, a clear statement of any assumptions made.

2. This is an OPEN BOOK EXAM. Candidates may use any non-communicating calculator.

3. FIVE (5) questions constitute a complete exam paper. The first five questions as they appear in the answer book will be marked.

4. Each question is of equal value.

5. Most questions require short written answers. Clarity and organization of the answer are important, but full sentences are NOT required. Be sure to bullet lists and ideas wherever possible.

Marking Scheme

1.	a) 4 marks b) 3 marks c) 3 marks		5.	a) 3 marks b) 4 marks c) 3 marks
2.	a) 5 marks b) 5 marks		6.	a) 3 marks b) 3 marks c) 4 marks
3.	a) 3 marks b) 3 marks c) 4 marks		7.	10 marks
4.	a) 3 marks b) 4 marks c) 3 marks		8.	10 marks

1.
   a) Why is the software quality assurance (SQA) plan important in the software project plan?
   b) What is software maintainability?
   c) What is software usability?

2.
   a) Consider two cases: (1) a medium-size software company, for instance 100 employees, uses the Linear (Waterfall) model of software development process; (2) a small software team, for instance 5 developers, uses agile development with short (1-2 week) sprints. Who is responsible to conduct the Formal Technical Review (FTR)?
   b) If you are responsible for organizing SQA procedures in an agile development team, when and how the SQA review is likely to be scheduled.

3.
   a) Why it is likely that different people conduct development and testing?
   b) Explain briefly how the requirements and design specifications are effective to create test cases.
   c) Give a brief example of one specification (e.g. Use Case diagram, Class diagram, Sequence diagram, Deployment diagram, Access table, etc.) to support the answer (b).

4.
   a) What is the purpose of the unit test and integration tests?
   b) Describe briefly the "sandwich" integration testing approach. What is the advantage of this approach in comparison with the top-down and bottom-up strategies?
   c) What is the main reason to use regression testing?

5.
   a) In which cases the black-box unit testing technique is preferable?
   b) Describe briefly the equivalence partitioning testing technique. Is this a white-box or black-box technique? Provide one case in which this technique is important.
   c) Describe briefly at least two white-box testing techniques. Which technique is the most useful during the past decade and why?

6.
   a) What are the equivalents of the unit and integration testing in the Object-Oriented context?
   b) How software quality is related to software usability?
   c) How human resource management in a software company/team can affect software quality?

7. Assume you test a simple unit – authentication subsystem. The length of login must be from six to eight characters. The password must be six to eight characters long, contain lower and upper case letters and numbers. Create a **black-box** test for the unit. Mention the testing technique you use. Show all your work!!!

8. Assume you test a simple unit in Appendix A. Create a **white-box** test for the unit. Mention the testing technique you use. Show all your work!!!

```c
#include <stdio.h>
int main(){
    int year;
    printf("Enter a year: ");
    scanf("%d",&year);
    if(year%4 == 0)
    {
        if( year%100 == 0) /* Checking for a century year */
        {
            if ( year%400 == 0)
                printf("%d is a leap year.", year);
            else
                printf("%d is not a leap year.", year);
        }
        else
            printf("%d is a leap year.", year );
    }
    else
        printf("%d is not a leap year.", year);
    return 0;
}
```