

National Exams May 2015

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.

1.
 - a) What is usually included in the software quality assurance (SQA) plan?
 - b) What is software reliability?
 - c) What is software safety?

2.
 - a) What is the Formal Technical Review (FTR)? What are the constraints of the review meeting?
 - b) If you create review guidelines, identify and explain briefly five points of the guidelines.

3.
 - a) What are the main testing objectives?
 - b) List and explain briefly three testing principles to design effective test cases.
 - c) What are characteristics of testability?

4.
 - a) What is the testing strategy and what is the testing technique?
 - b) Describe briefly integration testing approaches you know.
 - c) Would it be reasonable to use regression testing in agile development?

5.
 - a) What is the difference between white-box and black-box testing?
 - b) What is the cyclomatic complexity of a program? Describe three ways to compute cyclomatic complexity.
 - c) Describe briefly at least two black-box testing techniques.

6.
 - a) Describe briefly five software quality metrics.
 - b) How software quality is related to software maintainability?
 - c) How software configuration management can affect software quality?

7. Assume you test a simple fragment of a database in Appendix A. Using the “insert” statements create a black-box test for the table “vehicle”.

8. Assume you test a simple unit in Appendix B. Create a test for this unit using the basis path testing approach. Show all your work!!!

Appendix A

```
drop database carDealership;  
create database carDealership;  
use carDealership
```

```
create table dealership (  
    dealer_name char (20) NOT NULL PRIMARY KEY,  
    dealer_address char (30) NOT NULL,  
    dealer_phone varchar (15) NOT NULL  
);
```

```
create table shipper (  
    ship_name char (20) NOT NULL PRIMARY KEY,  
    ship_address char (30) NOT NULL,  
    ship_telephone varchar (15) NOT NULL,  
    acc_number int NOT NULL  
);
```

```
create table vehicle (  
    vehicle_name char (20) NOT NULL,  
    vehicle_type char (20) NOT NULL,  
    vehicle_manufacturer char (20) NOT NULL,  
    vehicle_price int NOT NULL,  
    PRIMARY KEY (vehicle_name, vehicle_type)  
);
```

```
create table manufacturer (  
    manufacturer_name char (20) NOT NULL PRIMARY KEY,  
    manufacturer_address char (30) NOT NULL,  
    manufacturer_phone varchar (15) NOT NULL  
);
```

Appendix B

```
1. /*
2.   Generate Pyramid For a Given Number Example
3.   This Java example shows how to generate a pyramid of numbers for given
4.   number using for loop example.
5. */
6.
7. import java.io.BufferedReader;
8. import java.io.InputStreamReader;
9.
10. public class GeneratePyramidExample {
11.
12.     public static void main (String[] args) throws Exception{
13.
14.         BufferedReader keyboard = new BufferedReader (new
15.         InputStreamReader (System.in));
16.
17.         System.out.println("Enter Number:");
18.         int as= Integer.parseInt (keyboard.readLine());
19.         System.out.println("Enter X:");
20.         int x= Integer.parseInt (keyboard.readLine());
21.
22.         int y = 0;
23.
24.         for(int i=0; i<= as ;i++){
25.
26.             for(int j=1; j <= i ; j++){
27.                 System.out.print(y + "\t");
28.                 y = y + x;
29.             }
30.
31.             System.out.println("");
32.         }
33.     }
```

04-Soft-A6 Software Quality Assurance

Marking Scheme

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