Fundamentals of Software Testing
(A.Y. 2024/2025) – Duration: 1h30m

June 23^{rd}, 2025

Exercise 1.

Our company asked our team to develop a software component that is able to provide details on the quality checks to which a product should be submitted to before being approved for selling. In particular the software component takes in input the following parameters with

- the identifier of the product as a 4 digits where the first digit on the left has the following meaning:
  - 0 for products of type X
  - 1, 2, 3 for products of type Y
  - 4, 5 for products of type W
  - 6 for product of type Z
- a char specifying the color of the product as:
  - "c" for clear
  - "y" for yellow
  - "r" for red
  - "b" for black
- a number representing the length of the product. In the implementation it is necessary to consider that for products of lenght greater than 100cm and smaller than 10cm the company cannot perform the quality check, and the product has to be sent to an external company. Misuring systems owned by the company have a precision of ±1mm and return the length in centimeters using two decimal digits. The above specified limits have to be considered as uninfringeable, on the other hand quality checks performed externally are rather expensive and should be avoided, if possible.

In implementing the system it has to be considered that products of type Y and Z cannot be shorter than 20cm and longer than 90cm. Moreover they can only be white. The specific quality checks to perform depending on the product characteristics have not bedefined yet, in any case the system to implement should be able to discriminate all the possible relevant dimensions and combination of them.

You have been appointed test manager, and you have been asked to derive a test suite that could toughtfully test the implemented system. Select the strategy and discuss the motivations behind your selection. Provide value for each parameter possibly discussing how they should be combined and then specify the cardinality of the test suite you are going to deliver.

16 points

Exercise 2.

Consider the following program (warning: the program is only slightly inspired to the previous specification, no inferences from the previous exercise should be done)

```
 1 package qualitycontrol;
 2 
 3 public class QualityCheck {
 4 
 5   public static qualityProcedure qualityCheck (Product product) {
 6     QualityProcedure qp;
 7     String string_id = Integer.toString(product.id);
 8     char pcode = string_id.charAt(0);
 9     if (product.length <= 149.9 && product.length >=2.1) {
10       switch (pcode) {
11         case ’1’:
12           if (product.type == ’w’ || product.type == ’d’) {
13             qp = QualityProcedure.Type1;
14           }
15           qp = QualityProcedure.Type2;
16           break;
17         case ’2’:
18           if (product.type == ’b’) {
19             qp = QualityProcedure.Type3;
20             if (product.length >= 100) {
21               qp = QualityProcedure.Type2;
22             }
23           }
24           qp = QualityProcedure.Type2;
25           break;
26         case ’3’:
27     if (product.type == ’b’ || product.type == ’c’) {
28       qp = QualityProcedure.Type3;
29     }
30     qp = QualityProcedure.Type2;
31     break;
32         default:
33           qp = QualityProcedure.Type1;
34       }
35       if (qp == QualityProcedure.Type3 && product.length >= 100) {
36   qp = QualityProcedure.Type4;
37       }
38     } else {
39       qp = QualityProcedure.Type2;
40     }
41     return qp;
42   }
43 }
```

- Derive a test suite that satisfies the condition/decision coverage
- Derive a data-flow graph for the program. Some suggestions on how to perform the task:
  - Use line numbers to define the blocks in the data flow.
  - consider the variable in the class `Product` as three different variables.
- Provide an assessment for the all-uses coverage criteria, in relation to the test suite you previously derived.

16 points

Supporting Java classes

```java

//Product.java
package qualitycontrol;

public class Product {
  int id;
  char type;
  double length;
}

//QualityProcedure.java
package qualitycontrol;

public enum QualityProcedure {
  Type1,
  Type2,
  Type3,
  Type4
}
```