Prof. Dr.-Ing. Peter Liggesmeyer
M. Sc. Nishanth Laxman
M. Sc. Rasha Abu Qasem

TU Kaiserslautern
Dept. of Computer Science
AG Software Engineering: Dependability

Software Quality Assurance (WS 20/21)

Problem Set 4

Problem 1: Measurement Theory

a. We have A a set of software modules. We introduce a new binary empirical relation ¤ between two software modules on the set A. Write the formal definitions of the measurement axioms that should be satisfied by ¤ to yield an ordinal scale?

b. In Measurement Chapter, slide number 17, the following relations has been defined:

1) •≥ *more complex or equally complex*
2) •> *more complex*
3) •≈ *equally complex*

Clarify which axioms does each of the previous relations satisfy and subsequently the binary relation they form?
Note: you may find this link helpful.

Problem 2: Measurement Theory

On which scale types are the following measurement value groups valid and why? Please give the corresponding mapping and range for each group.

a) House number
b) Sea level of different sites
c) The amount of ducks on a lake
d) Weights of Martians on their planet

Problem 3: Single Measurement

What is McCabe's cyclomatic number? Determine the cyclomatic number for the following code snippets:

```
01 boolean ALL_POSITIVE(int[] array) {
02   boolean result;
03   int i,len,tmp;
04   len = array.length;
05   i=0;
06   result=true;
07   while (i<len&&result) {
08     tmp=array[i];
09     if (tmp<=0)
10       result=false;
11     i++;
12   }
13   return result;
14 }
```

```
01 public static int sum(int n) {
02   int sum = 0;
03   int i;
04   for (i = 1; i <= n; i++) {
05     sum = sum + i;
06   }
07   return sum;
08 }
```

```
01 public string printlnMCS() {
02   if (Type == MCSType.security)
03     return "MCS " + Number + " " + SecurityValue + "\n";
04   else if (Type == MCSType.safety)
05     return "MCS " + Number + " " + SafetyValue + "\n";
06     else
07       return "MCS "+Number+" ("+SafetyValue+", "+SecurityValue+")"+"\n";
08 }
```

Problem 4: Single Measurement

Given is a measure P, which equals the number of the atomic predicates in a software module. Atomic predicates in the sense of the measure P are only present in the decisions of a module. They have a Boolean value range and are not combined (Example: (x>5) is an atomic predicate; ((x=6) OR (y<z)) is not an atomic predicate, but is combined of two atomic predicates together)

a) What is the measure type of P?
b) Can the values of P be used as ordinal scale?
c) Can the values of P be used as rational scale in terms of the textual chaining of two modules?

Problem 5: Single Measurement

A data-flow oriented measure M_d should describe the number of different data accesses to different variables. Counted are defs, c-uses and p-uses; however, each variable will be counted only once.
Example: If *defs(x)* occurs more than once, only one time will be counted. If *c-uses(x)* occurs once or more times, the measure value will just be increased by 1. The same rule holds for *p-uses(x)*. Accesses to a different variable (e.g., *y*) will also be counted again.

```
y := x + 1;
y := y²;
z := y - 1;
```

For the code section mentioned above, the value of M_d is 4.

a) How should the empirical relation be evaluated concerning the given modifications 1-3, in order to apply the measure M_d as an ordinal scale?
    1. Add data access to new variable?
    2. Add already available data access to available variable?
    3. Add new data access type to available variable?
b) Can the measure M_d be used as a rational scale concerning the textual chaining of two modules? If so, why?
c) Please give the monotony condition as a criterion for the rational scale. Explain the significance of the monotony condition in your own words.
d) Please prove that measures *M*, that are quotients $M(C) = \frac{M_a(C)}{M_b(C)}$, generally do not fulfill the monotony condition in general (see the example of the textual chaining).