

Vorlesung Softwaretest und -debugging

Dr. Carsten Gnörlich

cg@techfak.uni-bielefeld.de

Übungsblatt 4

25. Mai. 2012

Aufgabe 1 (entspricht Aufgabe 4 aus Blatt 3)

Es sei noch einmal das Programm *wall* aus Übungsblatt 3 betrachtet. Geben Sie eine Testmenge für Aufrufe von *wall x y* an, so daß $C_i(2)$ -Überdeckung (starke C_{GI} -Überdeckung, also alle Iterationen bis zu einer Tiefe von 2) erreicht wird.

Aufgabe 2 (entspricht Aufgabe 5 aus Blatt 3)

Geben Sie alle sensitivierenden Testfälle (nach der $C_2(mM)$ -Überdeckung) für den Entscheidungsausdruck an, mit dem Sie ermitteln ob ein Baustein vom Typ „Full“ oder „Arc“ in der untersten Reihe plaziert wird.

Ist es möglich diese Fälle direkt über einen Aufruf des *wall*-Programmes zu generieren? Wie läßt sich dieses Problem gegebenenfalls lösen (nur beschreiben, nicht machen :-)?

Aufgabe 3

Geben Sie eine Menge von Testdaten an, um den folgenden Ausdruck auf Bereichsfehler zu testen: $(X < 20 \text{ and } Y \geq 5) \text{ or } Y > 10$

Aufgabe 4

Geben Sie eine Wertebelegung für die Variablen *x* und *y* an, um die folgende Funktion auf Datenzugriffs-Fehler zu untersuchen:

```
int f(int x, int y)
{ int a = 0;
  int b = 3;
  int c,d;

  c = b+x;
  d = y/c;
  ...
}
```

Aufgabe 5

Gegeben sei der folgende Programmausschnitt:

```
a:= x+1
b:= 2a
c:= 7
if b>9 then z:= b+x
      else z:= b+c/2
return z
```

- a) Zeichnen Sie den Datenflußgraphen auf, so daß er den Einschränkungen für das datenflußbezogene Testen genügt.
- b) Geben Sie eine minimale Testmenge für den Eingabewert x an, um das „alle defs“-Kriterium zu erfüllen.
- c) Erweitern Sie die Testmenge aus b), um das „alle def/ref-Interaktionen“-Kriterium zu erfüllen.