

Übung 1

Übungsaufgaben für den 27. Oktober 2001

Die in Java implementierte Klasse `OrderedTriple` dient in den folgenden Aufgaben als Grundlage und Anschauungsobjekt. Die Klasse enthält drei Eigenschaften vom Typ `int` (ganzzahlig) und sortiert automatisch, mit dem Erzeugen eines Objekts, die im Konstruktor übergebenen Parameterwerte aufsteigend.

```
class OrderedTriple {

    private int a;
    private int b;
    private int c;

    public OrderedTriple(int a,int b,int c) {
        init(a,b,c);
        order();
    }

    private void init(int a,int b,int c) {
        this.a=a;
        this.b=b;
        this.c=c;
    }

    private void order() {

        if ( (a>b) && (a>c) && (b>c) ||
            (a>b) && (a>c) && (b==c) ||
            (a==b) && (a>c) && (b>c) ) {
            flip_ac();
        }

        if ( (a>b) && (a<c) && (b<c) ||
            (a>b) && (a==c) && (b<c) ) {
            flip_ab();
        }

        if ( (a<b) && (a<c) && (b>c) ||
            (a<b) && (a==c) && (b>c) ) {
            flip_bc();
        }

        if ( (a>b) && (a>c) && (b<c) ) {
            flip_ab();
            flip_bc();
        }

        if ( (a<b) && (a>c) && (b>c) ) {
            flip_ac();
            flip_bc();
        }

    }

}
```

Übung 1

```
private void flip_ab() {
    int tmp;
    tmp=b;
    b=a;
    a=tmp;
}

private void flip_ac() {
    int tmp;
    tmp=c;
    c=a;
    a=tmp;
}
private void flip_bc() {
    int tmp;
    tmp=c;
    c=b;
    b=tmp;
}
}
```

Aufgabe 1 *Kontrollflußgraph*

Zeichnen Sie den Kontrollflußgraph zur Operation *order()*.

Versuchen Sie im folgenden stets eine minimale Anzahl an Testfällen zu erzeugen. Wählen Sie die Werte der Testdaten aus dem Intervall [1,3].

Aufgabe 2 *Anweisungsüberdeckung*

Geben Sie die notwendigen Testfälle der Operation *order()* für die Erfüllung der Anweisungsüberdeckung an.

Aufgabe 3 *Zweigüberdeckung*

Geben Sie die notwendigen Testfälle der Operation *order()* für die Erfüllung der Zweigüberdeckung an.

Aufgabe 4 *Bedingungsüberdeckung*

Die folgende Bedingungsanweisung ist Teil einer zu prüfenden Funktion.

```
...
if ( x==5 && y<0 || z>0 ) {
    ...
}
...
```

Übung 1

Die Variablen x, y und z sind als voneinander unabhängig zu betrachten. Bearbeiten Sie jede Teilaufgabe für links-unvollständige und vollständige Auswertung.

- Stellen Sie in einem geeigneten Schema die für einen einfachen Bedingungsüberdeckungstest notwendigen Testfälle dar.
- Stellen Sie in einem geeigneten Schema die für einen mehrfachen Bedingungsüberdeckungstest notwendigen Testfälle dar.
- Stellen Sie in einem geeigneten Schema die für einen minimalen mehrfachen Bedingungsüberdeckungstest notwendigen Testfälle dar.
- Stellen Sie in einem geeigneten Schema die für einen modifizierten Bedingungs-/Entscheidungsüberdeckungstest notwendigen Testfälle dar.

Aufgabe 5 Bedingungsüberdeckung

Benutzen Sie im folgenden das vereinfachte Schema der ersten Bedingung (blau dargestellt) in der Operation *order()*.

?	?	?	?	?	?	?	?	?
a>b	a>c	b>c	a=b	b=c	?????	?????	?????	?????

a	b	c
---	---	---

- Geben Sie die notwendigen Testfälle der Operation *order()* für die Erfüllung der einfachen Bedingungsüberdeckung an.

?	?	?	?	?	?	?	?	?
a>b	a>c	b>c	a=b	b=c	?????	?????	?????	?????

a	b	c

- Geben Sie die notwendigen Testfälle der Operation *order()* für die Erfüllung der minimalen mehrfachen Bedingungsüberdeckung an.

?	?	?	?	?	?	?	?	?
a>b	a>c	b>c	a=b	b=c	?????	?????	?????	?????

a	b	c

Übung 1

c) Geben Sie die notwendigen Testfälle der Operation *order()* für die Erfüllung der modifizierten Bedingungs-/Entscheidungsüberdeckung an.

?	?	?	?	?	?	?	?
a>b	a>c	b>c	a=b	b=c	?????	?????	?????

a	b	c

d) Welche Kopplung liegt zwischen den atomaren Bedingungen der betrachteten Bedingung I vor?

e) Bei dem in den vorangegangenen Aufgaben verwendeten Schema handelt es sich um eine starke Vereinfachung. Welcher Art ist diese Vereinfachung? Erstellen Sie ein Schema mit dem die Funktion *order()* mit dem mehrfachen Bedingungsüberdeckungstest geprüft werden kann.

f) Bestimmen Sie für die Operation *order()* die notwendige Anzahl von Testfällen für jedes in der Vorlesung vorgestellte kontrollflußorientierte Prüfverfahren.

Bei Fragen besteht jederzeit die Möglichkeit, sich an den zuständigen wissenschaftlichen Mitarbeiter, Christopher Robinson-Mallett, zu wenden.