

Arbeitsblatt (27.10.2015)

Black-Box-Tests

Hier üben Sie einen Black-Box-Test durchzuführen, dazu:

- ✓ Bestimmen Sie die Eingabeüberdeckung
- ✓ Bestimmen Sie die Ausgabeüberdeckung
- ✓ Bestimmen Sie Grenzwerte
- ✓ Bestimmen Sie die Funktionsüberdeckung

Spezifikation des Programms

Das Programm soll Einkäufe in einem Einkaufswagen verwalten.

Man kann Produkte einkaufen. Ein Produkt hat einen Typ, einen Namen (beliebige Zeichenkette) und einen Preis (ganze Zahl nicht größer als 100). Im Moment werden nur die Typen „Kekse“ und „Waffeln“ unterstützt.

Beim Einkauf werden die Produkte in einen Einkaufswagen gelegt. Ein Einkaufswagen hat einen Namen (beliebige Zeichenkette). Es passen max. 10 Produkte in einen Einkaufswagen.

Das Programm bietet die folgenden Funktionen zur Verwaltung von Produkten und Einkaufswagen

- Einkaufswagen anlegen bzw. anzeigen
- Produkt (identifiziert durch Typ, Name und Preis) einkaufen bzw. aus einem Einkaufswagen wieder entfernen
- Gesamtpreis des Einkaufswagen errechnen und anzeigen
- 2 Einkaufswagen miteinander vergleichen. Dabei werden die Produkte angezeigt, die in beiden Einkaufswagen die gleichen sind.

Daraus ergeben sich die folgenden Funktionen mit Ein- und Ausgabetypen. Der Systemzustand (also die Menge der vorhandenen Einkaufswagen) ist nicht als Ein/Ausgabe zu behandeln.

- EAnlegen(Name) : EWagen
- EAnzeigen(Name des EWagens)
- Einkaufen(Typ, Name des Produktes, Preis des Produkts, Name des EWagen)
- EinkaufZurücklegen (Typ, Name des Produktes, Preis des Produkts, Name des EWagen)
- EPreis(Name des EWagens)
- EVergleich(Name1, Name2) : ProduktMultimenge

Aufgabe 1: Ein- und Ausgabeüberdeckung und Grenzwerte

Identifizieren Sie für die verwendeten Eingaben (unabhängig von der konkreten Funktion) etwaige Einschränkungen aus der Spezifikation und leiten Sie daraus die gültigen und ungültigen Äquivalenzklassen ab. Berücksichtigen Sie dabei Grenzwerte. Beachten Sie dass der Füllzustand des Einkaufswagens in den Funktionen eine Art impliziter Parameter ist. Ungültig bedeutet hier nur, dass die Eingabe nicht den richtigen Datentyp hat (z.B. Preis hat falsche Größe).

Betrachten Sie als Eingaben auch die Kommandos zum Aufrufen der Funktionen.

Beispiel:

- Name für Einkaufswagen : muss eindeutig sein in Bezug auf den Systemzustand; außer beim Anlegen muss der Wagen bekannt sein
 - Gültig ENameK1: beliebige Zeichenkette
 - Man könnte hier noch typische Probleme abprüfen mit nicht so üblichen Zeichen: sind gültige Eingaben, die aber leicht zu Fehlern führen können
 - Ungültig : gibt es nicht

Diese Möglichkeiten müssen bei jeder Funktion überprüft werden.

Identifizieren Sie weiterhin pro Funktion

- Weitere ungültige Eingaben, die aufgrund des Systemzustands bei dieser Funktionen nicht erlaubt sind
 - Beispiel: Fehler FAnlegenK1: beim Anlegen: Name nicht neu
- Für die gültigen Eingaben weitere Unterscheidung, um wichtige Ausgaben abzudecken:
 - welche Funktion hat interessante unterschiedliche Ausgabe?

Aufgabe 2: Funktionsüberdeckung

Stellen Sie nun in einer Tabelle zusammen welche Testfälle Sie pro Funktion ausführen wollen. Stellen Sie dabei sicher, dass alle Ein- und Ausgabeäquivalenzklassen mindestens einmal vorkommen.