2.1 Die lineare Datenstruktur Liste – Erzeugung und Einfügen von Elementen

Einstieg:

- Folgende grundlegende Funktionalitäten sollten genannt werden: Hinzufügen eines neuen Eintrags, Löschen eines Eintrags, Abändern eines bestehenden Eintrags
- Vorteile:
  - einfache Umsetzbarkeit
  - unkomplizierter Zugriff auf einzelne Elemente
  - …

  Nachteile:
  - möglicherweise sehr viele Variablen
  - unübersichtlich und schwer verständlich
  - Bei Veränderungen an allen Elementen muss der Wert jeder Variable einzeln verändert werden.
  - …

A1

siehe Snap!-Datei `2-1_A1-A2_Einkaufsliste_Loesung.xml`

A2

siehe Snap!-Datei `2-1_A1-A2_Einkaufsliste_Loesung.xml`

A3

a) siehe Snap!-Datei `2-1_A3_Einkaufsliste_Loesung.xml`

b) siehe Snap!-Datei `2-1_A3_Einkaufsliste_Loesung.xml`

c) Das Element wird nicht an Position `100`, sondern an der ersten freien Position eingefügt. Da es sich bei der Liste um eine dynamische Datenstruktur handelt, passt sich deren Größe stets der Anzahl an enthaltenen Elementen an. Würde das Element tatsächlich an Position `100` eingefügt werden, so hätte die Liste danach Größe `100`, jedoch nicht zwangsläufig auch `100` Elemente. Aus diesem Grund „rutscht“ das Element so weit wie möglich nach vorne.

Aufgabe 1

a) Es wird `5` zurückgegeben

b) Der Eintrag „`Melanie`“ wird mit Index `6` in die Liste eingefügt.

c) Der Eintrag „`Vanessa`“ wird mit Index `4` in die Liste eingefügt. Die Einträge „`Tim`“, „`Tina`“ und „`Melanie`“ rutschen dadurch um eins nach hinten.

d) Die Liste hat aktuell Länge `7`. Also wird der Eintrag „`Simone`“ mit Index `8` in die Liste eingefügt.

© C.C.Buchner Verlag, Bamberg