Programmieren 1

ptlwedel

PRIVATE BERUFSFACHSCHULE

Wintersemester 2016/2017

Marcus Riemer, B.Sc.

Basierend auf den Unterlagen „Programmstrukturen 1“
von Prof. Dr. Andreas Häuslein

17 Aufgabe „Rezeptesammlung“

In dieser Aufgabe bauen Sie ein sehr einfaches Programm mit dem geprüft werden kann, ob für eine Auswahl von Rezepten alle nötigen Zutaten zur Verfügung stehen. Für die Lösung dieser Aufgabe ist auf der Webseite ein Rahmenprogramm aus Prozedur- und Funktionsdeklarationen vorgegeben.

- Im Gegensatz zur Aufgabe “Hangman” müssen Sie dieses Mal sowohl die Funktionsblöcke als auch das Hauptprogramm implementieren.
- Vor der Implementierung des Hauptprogramms nutzen Sie jedoch die bereitgestellten Testfälle. Erst wenn Sie alle Funktionen korrekt implementiert haben kommentieren Sie diese aus und beginnen mit der Arbeit am Hauptprogramm.

17 Aufgabe „Rezeptesammlung“

```
type
   TIngredient = (Flour, Milk, Egg, Sugar,
                  Butter, Apple, Yeast, Water, NoIngredient);
   TIngredientSet = set of TIngredient;
   TRecipeIndex = 1..6;
const
   ALL_RECIPES : array[TRecipeIndex] of TIngredientSet =
      (  [Flour, Milk, Egg, Sugar],
         [Flour, Milk, Egg, Sugar, Apple],
         [Flour, Sugar, Yeast, Water],
         [Flour, Egg, Sugar, Butter, Apple, Yeast],
         [Sugar, Apple, Water],
         [Apple]);
   RECIPE_NAMES : array[TRecipeIndex] of string =
      ('Pfannkuchen', 'Apfelpfannkuchen', 'Bier', 'Apfelkuchen',
       'Apfelsaft', 'Apfel');
   INGREDIENT_NAMES : array[TIngredient] of string =
      ('Mehl', 'Milch', 'Ei', 'Zucker', 'Butter', 'Apfel', 'Hefe',
       'Wasser', 'Fehler');
```

- Erweitern sie die verfügbaren Zutaten um den Wert `Love` (“Liebe”) und erfinden Sie ein beliebiges Rezept, welches diese Zutat verwendet.

17 Aufgabe „Rezeptesammlung“

```
// Gibt eine Menge von Zutaten aus. Die einzelnen Zutaten werden
// dabei durch ein Leerzeichen getrennt, die Ausgabe wird mit
// einem Zeilenumbruch abgeschlossen.
//
// @param toPrint Die auszugebende Menge mit Zutaten
//
// @bonus Geben sie die Zutaten in der mathematischen
//        Mengenschreibweise mit geschweiften Klammern aus. Achten
//        Sie dabei auf die korrekte Anzahl an Kommata, vor der
//        geschweiften Klammer darf kein Komma stehen.
procedure printIngredientSet(toPrint : TIngredientSet);
```

17 Aufgabe „Rezeptesammlung“

```
// Prüft ob mit den übergebenen Zutaten das gegebene Rezept gebacken
// werden kann.
//
// @param available Die verfügbaren Zutaten
// @param recipe Die im Rezepte benötigten Zutaten
// @return True, wenn jede geforderte Zutat des Rezept in den verfügbaren
//         Zutaten enthalten ist.
//
// @hint  Diese Funktion ist ein Einzeiler
function canBake(available, recipe : TIngredientSet) : Boolean;
```

17 Aufgabe „Rezeptesammlung“

```
// Gibt aus, ob das Rezepte gebacken werden kann oder nicht. Dem Benutzer
// soll dabei der Name des Rezeptes ausgegeben werden. Sofern es nicht
// gebacken werden kann, werden die fehlenden Zutaten ausgegeben.
//
// @param available Die verfügbaren Zutaten
// @param recipeId Die ID des zu backenden Rezepts.
//                 Bezieht sich auf einen Eintrag im Array ALL_RECIPES.
//
// @hint Nutzen sie die Funktionen / Prozeduren canBake und
//       printIngredientSet
procedure printRecipeAvailability(available : TIngredientSet;
                                  recipeId : TRecipeIndex);
```

17 Aufgabe „Rezeptesammlung“

```
// Wandelt die String-Darstellung einer Zutat in die entsprechende
// interne Darstellung für Zutaten.
//
// @param Ein String, der einem Wert aus
//        INGREDIENT_NAMES entsprechen sollte
// @return Die passende Zutat oder NoIngredient,
//         sofern der String auf keine Zutat passt
function ingredientFromString(strIngredient : string) : TIngredient;
```

17 Aufgabe „Rezeptesammlung“

```
// Nimmt einen String im Format '+Zutat' oder '–Zutat'.
// * Wenn das erste Zeichen ein + ist wird die Zutat hinzugefügt.
// * Wenn das erste Zeichen ein - ist wird die Zutat entfernt.
// * Sofern die angegebene Zutat nicht existiert oder das erste Zeichen
//   nicht passt ist der Parameter insgesamt ungültig.
//
// @param operation Der Anweisungsstring, soll mit '+' oder '-' beginnen,
//                  dann folgt der Name einer Zutat.
// @param current Die aktuell verfügbaren Zutaten
// @return Eine Menge an Zutaten mit einer weiteren oder einer fehlenden
//         Zutat. Wenn der Parameter "operation" ungültig ist soll
//         der Parameter "current" ohne Veränderungen zurückgegeben
//         werden.
function updateAvailableIngredients(operation : string;
                                    current : TIngredientSet)
            : TIngredientSet;
```

17 Aufgabe „Rezeptesammlung“

Implementieren Sie nun das Hauptprogramm wie folgt:

- Die aktuell dem Benutzer zur Verfügung stehenden Zutaten werden in einer Variablen gespeichert, diese ist zu Beginn leer.
- Dann soll der Benutzer so lange mit `+Zutat` oder `-Zutat` Veränderungen an der Menge der verfügbaren Zutaten vornehmen, bis er diese mit `X` oder `x` beendet.
- Nach jeder (gültigen oder ungültigen) Eingabe, werden die aktuell verfügbaren Zutaten angezeigt.
- Im Anschluss sollen zunächst alle Rezepte ausgegeben werden, die umgesetzt werden können. Danach erfolgt die Ausgabe aller Rezepte, für die Zutaten fehlen.