

7. Übung des Programmierpraktikums (Mathematik)

Abgabetermin: Fr, 29. Juni 2007, 23:59 Uhr

Die Übungen sind grundsätzlich allein zu machen. Gruppenarbeit ist nicht erlaubt. Abzugeben sind jeweils die sinnvoll dokumentierten Programmfiles (Files für Bsp. 5: `bsp_5.cpp`, `bsp_5_fkt.cpp`, `bsp_5_fkt.hpp`) indem Sie diese in Ihr Verzeichnis `math197:lv\haase\Abgabe\Name_Vorname` kopieren.

29. (8 P) Implementieren Sie eine Klasse Sparschwein, welche folgende Datenelemente besitzt:

- Zähler für vier Arten von Geldstücken (1-Cent, 10-Cent, 50-Cent, 1-Euro)
- Die maximale Anzahl von Geldstücken die ins Sparschwein passen.
- Ein Flag, welches anzeigt, ob das Sparschwein aufgebrochen wurde.

Die Klasse soll die folgenden Methoden besitzen:

- `init(...)` initialisiert ein leeres Sparschwein,
- `add1Cents(...)` wirft eine Anzahl 1-Cent-Münzen hinein,
- `add10Cents(...)`, `add50Cents(...)`, `add1Euro(...)` wie oben,
- `isEmpty(...)` überprüft, ob das Sparschwein leer ist,
- `isFull(...)` überprüft, ob das Sparschwein voll ist,
- `isBroken(...)` überprüft, ob das Sparschwein geknackt wurde,
- `breakInto(...)` bricht das Sparschwein auf, und liefert die gesparte Geldsumme zurück.

Benutzen Sie Header- und Sourcefile zur Implementierung der Klasse. Das Hauptprogramm (ein weiteres Sourcefile) soll Münzen solange in das Sparschwein einwerfen, bis dieses voll ist und es dann knacken. Zeigen Sie den angesammelten Betrag an. Werfen Sie danach noch 37 Cent in das Sparschwein und knacken Sie es erneut (Wieviel Geld ist drin?).

30. (12 P) Nutzen Sie für diese Aufgabe die Klassen `string` und `vector` aus der STL. Schauen Sie sich dazu §2.3.1 und §2.3.4 des Vorlesungsskriptes (bzw. `string`¹ und `vector`²) an. Zu `vector` können Sie sich auch ein Bsp. in §3.3.1 des Download-Buches von Breyman³ und zu `string` die Vorlesung von Schröder⁴ anschauen.

Deklariieren Sie eine Klasse zur Darstellung eines Handys, das durch einen Gerätenamen, den Namen des Besitzers, die Telefonnummer und die monatliche Grundgebühr charakterisiert ist. Diese Klasse soll außer den üblichen Konstruktoren (falls diese nötig sind) Zugriffsmethoden auf die Member enthalten und die gesamten Daten zum Handy ausgeben können.

Nutzen Sie diese Handy-Klasse, um sich ein Adressbuch Ihrer Freunde anzulegen (Sie sollten mehr als 5 haben). Wie wäre es mit einer Klasse Adressbuch?

Lesen Sie die unsortierte Datensätze von einem File ein und nutzen Sie evtl. die Funktion `getline`⁵ zum Einlesen ganzer Zeilen inklusive der Leerzeichen.

Geben Sie Ihr sortiertes Adressbuch aus (Tip: in den oben gegebenen `www`-Referenzen nach einem Algorithmus `sort` suchen).

Speichern Sie Klassendeklaration und -implementierung in den Files *bsp_30_fkt.hpp* und *bsp_30_fkt.cpp*. Geben Sie auch Ihr(e) Datenfile(s) *bsp_30_data.txt* ab.

¹<http://www.cplusplus.com/reference/string/>

²<http://www.cplusplus.com/reference/stl/vector/>

³<http://www.informatik.hs-bremen.de/~brey/stlb.html>

⁴http://www.cpp-tutor.de/cpp/1e06/1e06_02.htm

⁵http://www.cpp-tutor.de/cpp/1e06/1e06_02.htm