Prof. Dr. Harald Brandenburg
Hochschule für Technik und Wirtschaft (HTW)
Fachbereich 4 (Wirtschaftswissenschaften II)
Wilhelminenhofstraße 75 A
12459 Berlin (Oberschöneweide)
Raum WH C 605

Fon: (030) 50 19 - 23 17
Fax: (030) 50 19 - 26 71
harald.brandenburg@htw-berlin.de

Dienstag, 17. Dezember 2013

Programmierung 1

WS 2013 / 2014

Aufgabe 4:	Gruppe 1 07.01.2013	Gruppe 2 14.01.2014
-------------------	----------------------------	----------------------------

Der Zugriff auf ein Konto über einen Geldautomaten setzt den Besitz einer Chipkarte und die Kenntnis der zugehörigen 4-stelligen Geheimnummer voraus. Wird dreimal hintereinander eine falsche Nummer eingegeben, wird die Chipkarte vom Geldautomaten eingezogen.

Schreiben und dokumentieren Sie ein Programm, mit dem der Sicherheitsmechanismus von Geldautomaten getestet werden kann.

- Es soll folgendes Auswahlmenü zur Verfügung gestellt werden:

Test der Sicherheit eines Geldautomaten

- (1) **Einfacher Test**
- (2) **Mehrfacher Test**
- (3) **Programm beenden**

- Bei Wahl von (1) soll Folgendes geschehen:
 - Per Zufallszahlengenerator soll eine Geheimnummer erzeugt werden, die aus vier Ziffern besteht, die alle von 0 verschieden sind.
 - Anschließend soll versucht werden, von dem zugehörigen Konto abzuheben. Dazu soll jeweils per Zufallszahlengenerator eine weitere Nummer erzeugt werden, die aus vier von 0 verschiedenen Ziffern besteht, und geprüft werden, ob sie mit der Geheimnummer übereinstimmt. Nach drei Fehlversuchen soll der Vorgang abgebrochen werden (weil die Chipkarte eingezogen und das Konto gesperrt wurde!).
 - Dieser Vorgang soll (jeweils für ein anderes Konto mit einer neuen Geheimnummer) solange wiederholt werden, bis es zu einer (unberechtigten) Abhebung kommt, d.h. bis die Geheimnummer durch Zufall erraten wurde.
 - Die erfolglosen Vorgänge sollen gezählt und das Ergebnis z.B. in folgender Form auf dem Bildschirm ausgegeben werden:

Es wurden 789 Chipkarten eingezogen.

- Bei Wahl von (2) soll die Anzahl n der auszuführenden Tests über die Tastatur eingegeben werden ($1 \leq n \leq 100000$). Anschließend soll n -mal der Test gemäß (1) ausgeführt werden und am Ende ausgegeben werden:

Bei n Tests wurden im Durchschnitt k Chipkarten eingezogen.

[**Hinweise:**

- Das Programm soll sinnvoll auf mehrere Dateien mit zugehörigen Header-Dateien verteilt werden.
- Wann immer es möglich ist, sollen Dateien aus früheren Programmen – gegebenenfalls erweitert – wiederverwendet werden.
- Zum Erzeugen ganzer Zufallszahlen können Sie z.B. folgende Funktionen benutzen:

```
#include <stdlib.h>
#include <time.h>

void initialisiere_zufallszahlengenerator(void)
{
    srand((unsigned) time(NULL));
}

int liefere_ganze_zufallszahl(int min, int max)
{
    return (rand() % (max - min + 1)) + min;
}
```

- **Jede** Funktion Ihres Programms soll mit einem sinnvollen Dokumentationskommentar versehen sein, der ausführlich den Zweck und gegebenenfalls den Input (**@param**) und den Output (**@return**) der Funktion beschreibt (siehe entsprechende Folien).
- Auf den Rechnern des Labors sind (in dieser Reihenfolge) zu präsentieren:
 - die mit Hilfe von **Doxygen** erzeugte (HTML-)Dokumentation,
 - die C-Dateien,
 - die Übersetzung des Programms mit Hilfe von **scons** und **SConstruct**,
 - die Ausführung des Programms.
- Selbstverständlich darf Ihr Programm auch mehr leisten als gefordert.

]