

Bachelor-Studiengang Informatik
Übungen zur Vorlesung "Grundlagen der Informatik" (GDI), WS 2011/2012

Übungsblatt 9:

Ausgabe am: 25.11.2011
Abgabe am: 9.12.2011

Aufgabe 1: **5 + 3 = 8 Punkte**

Sehen Sie sich folgendes Interface *BaconChiffre* (Sie finden es auch auf der Wiki-Seite zur Vorlesung zum Download):

```
public interface BaconChiffre {

    String reinigeUrsprungstext( String text );

    String kodiereUrsprungstext( String text );

    String dekodiereUrsprungstext( String binaerCode );

    String versteckeText( String ursprungstext, String mediumText );

    String zeigeText( String steganogramm );

}
```

- a) Erstellen Sie eine Klasse *StandardBaconChiffre*, die dieses Interface implementiert. Benutzen Sie für die Implementierung Ihrer Methoden die (bei Bedarf verbesserte!) Implementierung Ihrer Methoden von Übungsblatt 5.
- b) Erstellen Sie eine Klasse *Start*, die Ihre ursprüngliche *main*-Methode mit der Menüsteuerung von Übungsblatt 5 übernimmt. Passen Sie Ihren ursprünglichen Code so an, dass er mit der neuen Klasse *StandardBaconChiffre* arbeitet. Erstellen Sie dazu in *main* eine lokale Variable, die ein Objekt dieser Klasse aufnimmt, bei dem Sie die Kodier- und Dekodier-Methoden aufrufen können.

Aufgabe 2

7 + 3 = 10 Punkte

- a) Erstellen Sie eine Kopie Ihrer Klasse *StandardBaconChiffre* aus Aufgabe 1 und nennen Sie die Kopie *ExtendedBaconImplementation*. Modifizieren Sie alle betroffenen Methoden dieser Klasse so, dass auch die Zeichen in der folgenden Tabelle kodiert bzw. dekodiert werden:

Buchstabe	Code	Buchstabe	Code
Leerzeichen	ggkkk	Ä	gggkk
. oder !	ggkkg	Ö	gggkg
,	ggkgk	Ü	ggggk
?	ggkgg	ß	ggggg

Achtung: Die Signatur der Methoden aus dem Interface darf *nicht* verändert werden!

- b) Ergänzen Sie Ihre *main*-Methode in der *Start*-Klasse mit dem Hauptmenü um zwei weitere Punkte, mit denen Sie entweder die bisherige Standard-Implementierung oder die neue erweiterte Implementierung "aktiv" schalten. Das soll geschehen, indem Sie ein Objekt der jeweiligen Klasse erzeugen, das dann für die folgenden Kodier- und Dekodier-Operationen verwendet wird.
Beachten Sie, dass die Demo mit der Standard-Kodierung durchgeführt werden soll, danach aber wieder die zuvor eingestellte Kodierung verwendet werden muss.

Aufgabe 3

10 Punkte

Erstellen Sie ein Interface für ein *Hotel* mit einheitlichen Einzelzimmern, bei dem man Zimmer reservieren, stornieren, bezahlen kann etc.

- Überlegen Sie sich zumindest 5 sinnvolle Methoden mit ihren Parametern und Rückgabewerten. Benennen Sie alle Methoden mit ihren Parametern auf deutsch.
- Dokumentieren Sie jede Methode mittels Javadoc.
- Wenn Sie andere Interfaces oder Klassen benötigen, erstellen Sie diese ebenfalls (leer), so dass Ihr Interface korrekt übersetzt werden kann.

Hinweise

- Vergeben Sie vernünftige Namen für Ihre Variablen und Parameter und vergessen Sie die Kommentare nicht!
- Die Aufgaben sind in Eclipse zu bearbeiten, legen Sie dafür ein Package *uebung09* an.
- Von allen Aufgaben sind Programmausdrucke (Listings) abzugeben, *keine* Ausdrucke von Testläufen. Die Aufgaben sind im Labor mit Eclipse vorzuführen.
- Erlaubt sind *MakeItSimple*-Funktionen (keine nicht besprochene Funktionalität aus der Java Standard Bibliothek) und das bisher erworbene Wissen aus den GDI-Vorlesungen. Fragen Sie, bevor Sie Java-Konstrukte verwenden, die noch nicht behandelt wurden! Zusätzliche eigene Hilfsfunktionen (keine fremden oder externen) sind ausdrücklich erlaubt.