GK 11 IF 1 / Ziemke	1. Informatikklausur im 2. Halbjahr	14. März 2007

Name:

Thema: Verschlüsselungsverfahren, Modellierung (OOA mit UML)
Erl. Mittel: grafikfähiger Taschenrechner (GTR), z. B. TI-84 plus, Vigenère-Quadrat
Arbeitszeit: 2 Unterrichtsstunden

1. Aufgabe: Im Unterricht wurden beispielhaft drei Verschlüsselungsverfahren besprochen. Stellen Sie (kurz – aber ausführlich genug) deren Vor- und Nachteile dar. Sie sollen die Verfahren hier nicht beschreiben!

2. Aufgabe: Eine mit dem Vigenère-Verfahren verschlüsselte Nachricht sei gegeben:

P	N	V	V	K	I	Y	O	S	W	K	H	K

a) Entschlüsseln Sie diese unter Verwendung des Codewortes LANDRTUCS. Hinweis: Im Anhang finden Sie als Hilfsmittel ein Vigenère-Quadrat.

b) Beschreiben Sie, weshalb die Kenntnis der Codewortlänge zum Knacken des Cypher hilfreich ist.

3. Aufgabe: Im RSA-Verfahren seien für das öffentliche Schloss N die Zahlen p=43 und q=131 gewählt.

a) Ermitteln Sie den kleinstmöglichen öffentlichen Schlüssel e. Benennen Sie für jede kleinere Primzahl, weshalb sie ungeeignet ist. Wozu dient der Schlüssel e?

b) Berechnen Sie den zugehörigen privaten Schlüssel d. Erläutern Sie zuvor die Gleichung $d \cdot e \bmod (p-1)\cdot(q-1) = 1$ und formen Sie diese ausführlich um in $d = \frac{k \cdot (p-1)\cdot(q-1)+1}{e}$. Wozu dient der Schlüssel d?

Hinweis: Nutzen Sie hier den GTR.

c) Geben Sie (z. B. in Form zweier Gleichungen) an, durch welche Umformungen eine Nachricht M zu C verschlüsselt wird, und wodurch aus C die Nachricht M zurück erhalten wird.

4. Aufgabe: Die behandelten Verschlüsselungsverfahren werden nun objektorientiert modelliert.

a) Stellen Sie zur Modellierung ein möglichst vollständiges UML-Klassendiagramm auf, das mindestens die Klassen *Crypto* (als Oberklasse), *Caesar*, *Vigenere* und *Rsa*, u. a. die Attribute *message* und *cypher* und u. a. die Methoden *encrypt()* und *decrypt()* enthält.

b) Ergänzen Sie im obigen Diagramm die Methoden *char2code()* und *char2ascii()*. Sie sollen einen Buchstaben in den Zahlencode (01-26) bzw. den ASCII (65-90) wandeln. Auch *code2char()* und *ascii2char()* sind zu ergänzen. Markieren Sie auch den geeigneten Schutz zu diesen Methoden.

c) Notieren Sie (in Pseudo-Code, in Java oder als Grobalgorithmus) den Algorithmus zu den Methoden *char2code()* und *code2char()*. Die Methode soll jeweils den umgewandelten Wert zurückgeben.
Hinweis: Beachten Sie dazu die Type-Cast-Hinweise in der Anlage.

d) Implementieren Sie (in Pseudo-Code, in Java, als Grobalgorithmus oder Struktogramm) nun die Methode *encrypt()* der Caesar-Klasse.
Hinweis: this.message="TEST"; write (this.message.charAt(2)); // schreibt 'S'

Viel Erfolg bei der Bearbeitung!

GK 11 IF 2 / Ziemke	1. Informatikklausur im 2. Halbjahr	9. März 2007

Name:

Thema: Verschlüsselungsverfahren
Erl. Mittel: grafikfähiger Taschenrechner (GTR), z. B. TI-84 plus, Vigenère-Quadrat
Arbeitszeit: 2 Unterrichtsstunden

1. Aufgabe: Im Unterricht wurden beispielhaft drei Verschlüsselungsverfahren besprochen. Stellen Sie (kurz – aber ausführlich genug) deren Vor- und Nachteile dar. Sie sollen die Verfahren hier nicht beschreiben!

2. Aufgabe: Eine mit dem Vigenère-Verfahren verschlüsselte Nachricht sei gegeben:

A	E	Z	D	C	D	O	J	A	D	T	R	Q

Entschlüsseln Sie diese unter Verwendung des Codewortes LANDRTUCS.
Hinweis: Im Anhang finden Sie als Hilfsmittel ein Vigenère-Quadrat.

3. Aufgabe: Im RSA-Verfahren seien für das öffentliche Schloss N die Zahlen p=67 und q=71 gewählt.

a) Ermitteln Sie den kleinstmöglichen öffentlichen Schlüssel e. Benennen Sie für jede kleinere Primzahl, weshalb sie ungeeignet ist. Wozu dient der Schlüssel e?

b) Berechnen Sie den zugehörigen privaten Schlüssel d. Erläutern Sie zuvor die Gleichung $d \cdot e \bmod (p-1)\cdot(q-1) = 1$ und formen Sie diese ausführlich um in $d = \frac{k\cdot(p-1)\cdot(q-1)+1}{e}$. Wozu dient der Schlüssel d?
Hinweis: Nutzen Sie hier den GTR.

c) Geben Sie (z. B. in Form zweier Gleichungen) an, durch welche Umformungen eine Nachricht M zu C verschlüsselt wird, und wodurch aus C die Nachricht M zurück erhalten wird.

4. Aufgabe: Zum Verschlüsseln mittels RSA wird eine Methode ***istPrimzahl(int zahl)*** benötigt, die eine übernommene Zahl auf die Primzahl-Eigenschaft prüft.
Notieren Sie (in Pseudo-Code, in Java, als Grobalgorithmus oder in Form eines Struktogrammes) den Algorithmus. Die Methode soll *true* zurückgeben, falls *zahl* eine Primzahl ist, sonst *false*.
Beschreiben Sie, nach welchem Teiler-Kandidaten die Untersuchung jeweils beendet werden kann.
Behandeln Sie auch die Übernahme ungeeigneter – z. B. negativer Zahlen.
Hinweis: Nutzen Sie hier den Operator MOD (in Java: % - z. B. 14%3 ergibt 2).

Viel Erfolg bei der Bearbeitung!

Hilfsmittel: Vigenère-Quadrat

Klartext

Code	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
2	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
3	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
4	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
5	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
6	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
7	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
8	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
9	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
10	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
11	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
12	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
13	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
14	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
15	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
16	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
17	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
18	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
19	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
20	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
21	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
22	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
23	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
24	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
25	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
26	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Hinweis: Type-Cast in Java (Umwandlung innerhalb primitiver Datentypen; hier: char – int (write() dient als Pseudobefehl für die Ausgabe.)

```
char myChar = 'C';          // char-Variable wird initialisiert
int myInt = 0;              // int-Variable wird mit Wert 0 initialisiert
myInt = (int) myChar;       // Type-Cast von char in int; myInt erhält neuen Inhalt
write (myInt);              // schreibt 67
myChar = (char) myInt-2;    // Type-Cast von int in char
write (myChar);             // schreibt 'A'
```