

## 2 Verschlüsselung nach Trithemius

Die Stärke der Trithemius-Verschlüsselung beruht darauf, dass nicht nur ein, sondern 26 verschiedene Geheimtextalphabete benutzt werden, um eine Botschaft zu verschlüsseln.

Im ersten Schritt wird ein so genanntes Vigenere-Quadrat dargestellt. Unter einem Klartextalphabet sind 26 Geheimtextalphabete aufgelistet, jedes um einen Buchstaben nach hinten verschoben.

Klar	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	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
2	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
3	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
4	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
5	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
6	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
7	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
8	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
9	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
10	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
11	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
12	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
13	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
14	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
15	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
16	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
17	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
18	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
19	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
20	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
21	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
22	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
23	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
24	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
25	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
26	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

**Abbildung 2.1:** Vigenere-Quadrat

Eine Botschaft wird nun verschlüsselt, in dem man den 1. Buchstaben mit dem 1. Geheimtextalphabet verschlüsselt, den 2. Buchstaben mit dem 2.

Geheimtextalphabet, usw. Ist der Text länger als 26 Buchstaben, fängt man wieder mit dem 1. Geheimtextalphabet an.

Übung 2.1:

Codiere und decodiere nach dem obigen Verfahren die Worte im Klartext

- a) informatikunterricht
- b) geheimwort

Übung 2.2:

Gib deinem Nachbarn einen verschlüsselten kompletten Satz. Dein Nachbar soll diesen wieder decodieren.

Übung 2.3:

Überlege dir, welche Probleme bei der Verschlüsselung eines Satzes mit Hilfe dieses Codierungsverfahrens auftreten können.

**Bemerkung:**

Bei der Erstellung eines Programms in Java, in dem das Codierungsverfahren nach Trithemius simuliert, werden wir zunächst alle Leerzeichen, Sonderzeichen und Satzzeichen nicht berücksichtigen. Dies erleichtert die Erstellung des Quellcodes.

## **2.1 Ein Wort wird verschlüsselt**

Dieses Codierungsverfahren ist eine Erweiterung der Verschlüsselung nach Caesar. Das Programm *Caesar01e* kann deshalb als Grundlage verwendet und erweitert werden.

Übung 2.4:

Studiere den Quellcode von *Caesar01e*. Formuliere schriftlich anhand der Übung 2.1 die Unterschiede zwischen den beiden Codierungsverfahren nach Caesar und nach Trithemius.

Übung 2.5:

Erstelle nun in **BlueJ** ein Programm *Trithemius01a*, das ein komplettes Wort verschlüsseln kann. Überlege dir, welche Teile von *Caesar01e* übernommen werden können und welche Teile ergänzt werden müssen.

Übung 2.6:

Entwerfe nun in **BlueJ** ein Programm *Trithemius01b*, das die grafische Oberfläche erstellt. Verwende dazu den Quellcode von *Caesar02* und gestalte diesen entsprechend um.

Übung 2.7:

Nun entwerfe eine stand-alone-Applikation *Trithemius01c* analog zu *Caesar03c*.

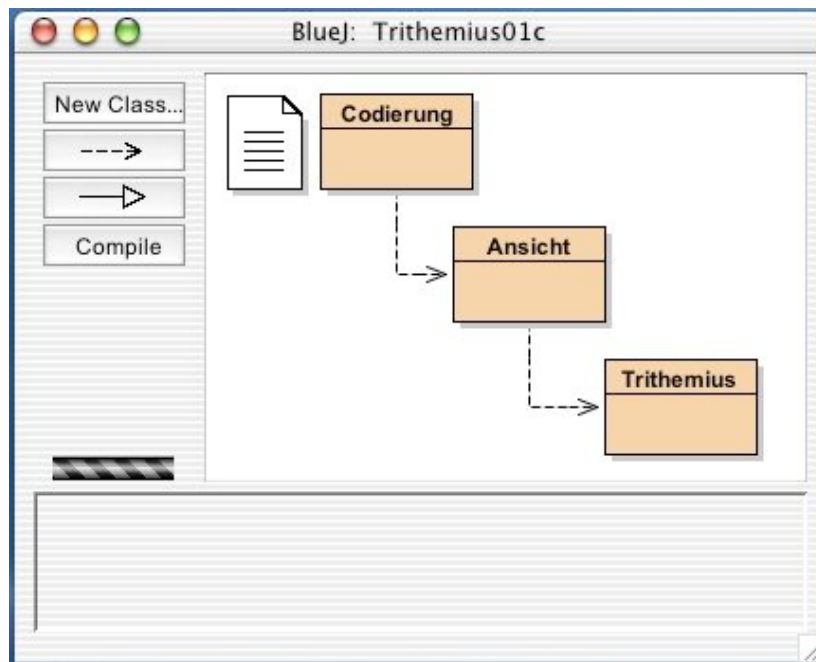


Abbildung 2.2: Klassendiagramm von *Trithemius01c*



Abbildung 2.3: Ein Beispiel für die GUI von *Trithemius01c*

## 2.2 Ein Wort wird entschlüsselt

Bei der Verschlüsselung wurde eine weitere Variable verwendet, die die entsprechende Verschiebung des Geheimtextalphabets steuerte. Diese erhielt den Startwert 1 und wurde in jedem Schleifendurchgang um 1 vergrößert. Beim Entschlüsseln muss dieser Vorgang nun wieder rückgängig gemacht werden

### Übung 2.8:

In den Übungen 2.1 und 2.2 hast du den Klartext *geheimwort* in den Geheimtext *HGKINSDWAD* verschlüsselt und wieder entschlüsselt. Formuliere schriftlich die einzelnen Decodierungsschritte.

### Übung 2.9:

Erstelle nun in **BlueJ** ein Programm *Trithemius01d*, das den Geheimtext an Hand deiner Überlegungen aus Übung 2.9 wieder entschlüsselt.

Überlege dir dazu in der Klasse *Trithemius*, welche Teile von *Trithemius01a* übernommen bzw. geändert werden können und welche Teile ergänzt werden müssen.

Ändere auch in der Klasse *Ansicht* die grafische Benutzeroberfläche entsprechend.

Erstelle nun mit Hilfe der Klasse *Codierung* eine stand-alone-Applikation.



Abbildung 2.4: Ein Beispiel für die GUI von *Trithemius01d*