

Prog1 WINF (WS 2020/21)

Prof. Dr. Im. Weber, Dr. T. Heindel, R. Hobeck, M. Grambow

Hinweise

- Die Abgabe dieser Übungsaufgaben muss bis spätestens Sonntag, den **22.11.20** 23.59h im ISIS-Kurs erfolgt sein. Es gelten die Ihnen bekannten Übungsbedingungen.
- Lösungen zu diesen Aufgaben sind als gezippter Projektordner abzugeben. Eine Anleitung zum Zippen von Projekten finden Sie auf der Seite des ISIS-Kurses.
- Bitte beachten Sie, dass Abgaben im Rahmen der Übungsleistung für die Zulassung zur Klausur relevant sind. Durch Plagiieren verwirken Sie sich die Möglichkeit zur Zulassung zur Klausur in diesem Semester.

Aufgabe 2, Block 1 (7 Punkte)

Aufgabe 2.1 – Arrays, Klassenmethoden und Schleifen (5 Punkte)

Erstellen Sie ein neues Projekt „Aufgabe2“ und darin eine neue Klasse mit Namen „VigenereCipher“. Binden Sie wie gewohnt die Prog1Tools ein. Schlussendlich soll diese Klasse mit Hilfe von Arrays, Klassenmethoden und Schleifen die sogenannte Vigenère-Chiffre implementieren (siehe <https://de.wikipedia.org/wiki/Vigen%C3%A8re-Chiffre>). Gehen Sie dazu in folgenden Schritten vor, wobei jede Klassenmethode mit entsprechenden Javadoc-Kommentaren zu versehen ist:

- Eine Methode theAlphabet ohne Parameter und als Rückgabewert ein char-Array der Länge 26, welches das lateinische Alphabet von `a` bis `z` enthält.
Tipp: (char) 97 ist der erste Buchstabe des Alphabets.
- Eine Methode shiftLeft mit einem char-Array und einem byte-Wert als formale Parameter und einem char-Array als Rückgabewert: Diese Methode soll das char-Array nach links rotieren, und zwar um so viele Stellen, wie vom byte-Wert angegeben. Die Rückgabe ist die rotierte Version des Arrays; das originale Array bleibt unberührt.
Tipp: Die Rotation von {`a`, `b`, `c`, `d`, `e`} um zwei Stellen nach links ergibt {`c`, `d`, `e`, `a`, `b`}; eine Rotation um null Stellen ergibt keine Veränderung.
- Eine Methode generateVigenereSquare ohne formale Parameter und einem zweidimensionalen char-Array als Rückgabewert, die der sogenannten *tabula recta* (https://en.wikipedia.org/wiki/Tabula_recta) entspricht.
Tipp: Benutzen Sie die Methode shiftLeft. Das Ergebnis ist sehr ähnlich zu der Tabelle in folgender Illustration:

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
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
B	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
C	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
D	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
E	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
F	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
G	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
H	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
I	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
J	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
K	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
L	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
M	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
N	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
O	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
P	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
Q	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
R	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
S	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
T	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
U	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
V	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
W	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
X	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
Y	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
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	Y

- Eine Methode encryptMsg mit einem char-Array und einem String-Wert als formale Parameter und einem String als Rückgabewert. Diese Methode soll die tatsächliche Verschlüsselung durchführen, wobei das char-Array als Schlüssel dient, der übergebene String der Klartext ist und der Rückgabewert der chiffrierte Text. Angenommen der Schlüssel ist `{'f', 'o', 'o', 'b', 'a', 'r'}`, dann wird ein Schlüsseltext wie folgt aus dem Klartext erzeugt:
 - Der erste Buchstabe des Klartextes wird entsprechend der Zeile »F« in obiger Tabelle verschlüsselt, sodass z.B. der Buchstabe »K« zu »P« wird (als erster Buchstabe des chiffrierten Textes).
 - Der zweite (und dritte) Buchstabe wird entsprechend der Zeile »O« verschlüsselt, sodass z.B. der Buchstabe »K« zu »Y« wird (als der zweite und dritte Buchstabe des chiffrierten Textes).
 - ...
 - Beim siebten Buchstaben wird dann wieder wie beim ersten Buchstaben verschlüsselt.

Tipp: Eine ausführlichere Beschreibung der Verschlüsselung befindet sich auf Wikipedia (<https://de.wikipedia.org/wiki/Vigen%C3%A8re-Chiffre>).

*Tipp: char-Werte können nach int **gecastet** und dann mit Addition und Subtraktion verändert werden; die Klassenmethode String.copyValueOf konvertiert ein char-Array in einen String.*

Hinweis: Wenn str eine Variable des Typs String ist, dann ist str.length() die Länge des Strings und str.charAt(5) das Zeichen an der Position 5, falls der String mindestens die Länge sechs hat.

- Eine main Methode: Hier soll zunächst die *tabular recta* auf der Konsole ausgegeben werden. Danach lesen Sie dann den Klartext ein, welchen Sie anschließend mit dem Schlüssel `{'f', 'o', 'o', 'b', 'a', 'r'}` verschlüsseln, um dann schließlich den chiffrierten Text auszugeben.
- Testen Sie Ihr Programm. Eine beispielhafte Ausgabe könnte wie folgt aussehen:

```
Das Quadrat von Vigenère sieht wie folgt aus:
{{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'},
{'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'}}

Bitte einen Klartext eingeben:einführung in die programmierung in java
Die verschlüsselte Nachricht ist: jwbguvmfogzsrwfpitufbmdnsfvnxnbxbvyr
```

Aufgabe 2.2 – Pixel-Schleifen (1 Punkte)

In dieser Aufgabe werden sie mit Hilfe von Java ein Bild zeichnen (siehe Abbildung 1).

Erstellen Sie eine neue Klasse „Zeichner“. Implementieren Sie nun eine main-Methode. Diese soll zuerst die Painter.show()-Methode aufrufen (enthalten in den Prog1Tools), diese sorgt dafür, dass Java ein Zeichenfenster anzeigen kann. Importieren Sie hierfür die richtige Klasse aus den Prog1Tools. Testen Sie Ihr Programm nun, um zu sehen ob Ihre Java-Installation ein Fenster anzeigt.

Die Painter-Bibliothek enthält ein Array mit den Bildinformationen für den in Abbildung 1 zu sehenden Pilz.

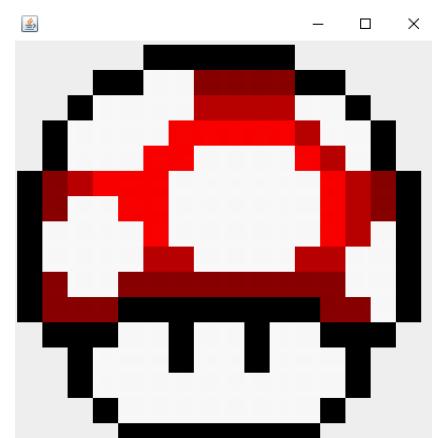


Abbildung 1

Nutzen Sie die richtige Kombination von Schleifen, um aus dem Painter.shroom-Array die Farbinformationen auszulesen und diese mit der Painter.paint-Methode auf den Bildschirm zu zeichnen.

Die Painter.paint-Methode erwartet eine x- und y-Koordinate auf der ein Pixel gezeichnet werden soll, und einen Farbwert. Das Painter.shroom-Array ist in zwei Dimension aufgeteilt, der Farbwert für die Position (x=10,y=20) befindet sich im Array an der Position Painter.shroom[20][10].

Nutzen Sie Javadoc-Kommentare und testen Sie Ihr Programm.

Aufgabe 2.3 – Überladene Methoden (1 Punkte)

Erstellen Sie eine neue Klasse „Concatenation“. Programmieren Sie drei verschiedene Klassenmethoden, die jeweils zwei formale Argumente vom Typ String oder char[] haben und die Konkatenation als String zurückgeben.

Ihre Klasse muss also Methoden mit den folgenden Methodensignaturen enthalten:

- void main(String[])
- String concatenate (char[] , String)
- String concatenate (String, char[])
- String concatenate (char[] , char[])

In der main-Methode

- definieren Sie zwei verschiedene char-Arrays und ein String,
- wenden Sie die Methoden in allen verschiedenen Kombinationen an
- und geben Sie die Resultate aus.

Nutzen Sie Javadoc-Kommentare und testen Sie Ihr Programm. Die Ausgabe könnte wie folgt aussehen.

```
42bar  
bar42  
foo42  
foobar  
42foo  
barfoo
```

Aufgabe 2.4 – Abgabe und im Lab

Exportieren Sie Ihren Projektordner mit allen Klassen in eine zip-Datei und laden sie diese auf der ISIS Kursseite hoch. Beantworten Sie die Fragen Ihres Tutors zu Ihrer Abgabe.