SVS Übung 1

Aufgabe 1.2

a),b)

Compilen mit g++ Main.cpp VigenereEncrypter.cpp VignereEncrypter.h –o VigenereEncrypter.exe

Erst Nachricht eingeben -> Enter -> Schlüssel eingeben -> Enter, dann d oder e für entschlüsseln bzw verschlüsseln -> Enter.

Lösung a)

`ouktitcbv_vadakahyhgduagmydnzhdnjbh_etsemywyeljsrtrhjtusrtsolucow`

Lösung b)

`bestaetige_auftragseingang`

c)

Kunde: ysbvakiskcowredkabosklkjlsykveqn_wezwsrpivunlr

Berater: escdavwwrp_rxtdbaxvstygrqu

Maximale Schlüssellänge: 6, beginne Tabelle aufzustellen für Schlüssellänge = 6, dann für 5 usw.

Kunde					
Y	S	B	V	A	K
I	S	K	C	O	W
R	E	D	K	A	B
O	S	K	L	K	J
L	S	Y	K	V	E
Q	N	_	W	E	Z
W	S	R	P	I	V
U	N	L	R		

Bank					
E	S	C	D	A	V
W	W	R	P	_	R
X	T	D	B	A	X
V	S	T	Y	G	R
Q	U				

Verteilungstabelle der Buchstaben nach Spalten, für Kundennachricht

	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	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	1	1	0	0	1	0	1	0	1	0	0
2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	**2**	0	0	0	**5**	0	0	0	0	0	0	0	0
3	0	1	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1
4	0	0	1	0	0	0	0	0	0	0	**2**	**1**	0	0	0	**1**	0	1	0	0	0	1	1	0	0	0	0
5	**2**	0	0	0	**1**	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
6	0	1	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0
N	H	M	M	M	H	M	M	H	H	M	M	M	M	H	H	M	L	H	H	H	M	L	L	L	L	L	H

In Spalte 2 tauchen sehr viele O und S auf. Der Abstand zwischen O und S ist 3, genau wie bei A und E. Vermutung: A -> O und E -> S, anhand der Verteilung der Buchstaben in der deutschen Schrift (vgl. dazu Spalte N mit dem normalen Auftreten im Deutschen).

Der Schlüssel ist also nun *O****.

In Spalte 4 tauchen „verhältnismäßig“ viele A und E auf. Vermutung: Der vierte Buchstabe des Schlüssels ist ein A.

Der Schlüssel ist nun *O**A*.

Verteilungstabelle für Nachricht der Bank:

	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	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	1	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	1	0	1	0	0	0	0
3	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
4	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
5	2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0	1	0	0	0
N	H	M	M	M	H	M	M	H	H	M	M	M	M	H	H	M	L	H	H	H	M	L	L	L	L	L	H

Entschlüsseln der Kunden-Nachricht mit bisherigem Schlüssel liefert:

`Y**e**bv**a**ui**e**kc**o**fr**r**dk**a**lo**e**kl**k**tl**e**yk**v**oq__w**e**iw**e**rp**i**eu_lr`

Die der Bank:

`e**e**cd**a**vw**i**rp_rx**f**db**a**xv**e**ty**g**rqg`

Verglichen mit der Standardantwort der Bank (→ Stark „formularhafter“ Charakter der Konversation)

```
bestaetige_auftragseingang
e**e**cd**a**vw**i**rp_rx**f**db**a**xv**e**ty**g**rqg
```

ist auffällig: gleiche Länge, Buchstaben stimmen überein.

- ⇨ B→E➔ Erster Buchstabe des Schlüssels ist evtl. D (Verschiebung um 4)
- ⇨ S→C➔ Zweiter Buchstabe des Schlüssels ist evtl. L (Verschiebung um 12)
- ⇨ T→D➔ Dritter Buchstabe des Schlüssels ist evtl. L (Verschiebung um 12)
- ⇨ E→V➔ Letzter Buchstabe des Schlüssels ist evtl. R (Verschiebung um 18)

Der Schlüssel ist demnach also DOLLAR.

Angewendet auf ergibt sich also:

Kunde: `verkaufe_sofort_alle_aktien_von_pleitegeier_ag`

Bank: `bestaetige_auftragseingang`