Facharbeit Informatik

Analyse, historische Einordnung & Implementation des Vigenère-Verfahrens

---

Inhaltsverzeichnis:

1 Einleitung

Die Kryptologie ist nach der Herleitung aus dem Griechischen (kryptos = geheim und Logos = Wort, Sinn) die Wissenschaft vom Verbergen. Die Verschlüsselung von Texten hat lange Tradition. Schon vor Tausenden von Jahren hatten die Menschen das Bedürfnis, Texte, die nicht in falsche Hände geraten sollten, sei es aus militärischen oder privaten Gründen, so zu verschlüsseln, dass nur Personen, die den Entschlüsselungsalgorithmus kennen, ihn auch entschlüsseln können.

In der Antike genügten noch einfache mechanische Vorrichtungen, doch da die gegnerische Seite immer wieder Mittel und Wege fand, die verschlüsselten Botschaften zu knacken, wurden die Geheimschriften im Laufe der Zeit immer komplizierter.

In der heutigen Zeit, wo der Kommunikationsaustausch über elektronische Medien zunehmend wächst, ist das Thema „Verschlüsselung von Nachrichten bzw. Daten“ aktueller und wichtiger als je zuvor. Wenn eine E- Mail über das Internet verschickt wird, könnte man dies mit dem Versand einer Postkarte vergleichen, die jeder, der sie in die Finger bekommt, lesen kann. Um zu verhindern, dass sicherheitsrelevante Daten (z.B. Konto- oder Kreditkartennummern) im weltumspannenden Datennetz auch zu anderen Leuten als zu den Empfängern gelangen, ist es unabdingbar diese Daten zu verschlüsseln.

Im folgenden wird zuerst auf die Kryptologie allgemein eingegangen um Fachbegriffe zu erklären und den Leser in dieses Thema einzuführen. Desweiteren liegt der Hauptteil dieser Facharbeit auf der Bearbeitung des Vigenère-Verfahrens.

1.1 Kryptologie: Kryptographie & Kryptoanalyse

Die Kryptologie gliedert sich in zwei Teilbereiche, in die Kryptographie und in die Kryptoanalyse.

Die Kryptographie (von griech. graphein: schreiben) beschäftigt sich mit der Wissenschaft, Inhalte von Nachrichten zu verheimlichen, bzw. vor der Kenntnisnahme Unbefugter zu schützen. Weiterhin beinhaltet sie die Aufgabe, sicherzustellen, dass Dokumente nicht unbemerkt verändert werden können. Eine weitere Anforderung an die Kryptographie ist, dass der Urheber eines Dokuments immer sicher feststellbar und nachweisbar sein sollte, so dass dieser seine Urheberschaft auch nicht

abstreiten kann.

Die Kryptoanalyse dagegen bezeichnet die Kunst, einen chiffrierten Text ohne Kenntnis des Schlüssels zu lesen, um somit die Sicherheit oder eben auch die Unsicherheit kryptographischer Systeme zu beweisen. Der Vorgang heißt auch Codebreaking oder Komoromittierung. Wenn ein Algorithmus der Kryptoanalyse nicht standhält, sagt man auch, er sei gebrochen oder kompromittiert. Kennt der Kryptoanalytiker ein relativ langes Stück des Geheimtextes, so kann er mit Hilfe von Häufigkeitsanalysen in Bezug auf Buchstaben, Bi- und Trigrammen der jeweiligen Sprache versuchen, die Chiffrierung zu kompromittieren.

1.2 Erklärung kryptologischer Grundbegriffe

Eine Nachricht, die nicht abgesichert wurde, also das Ausgangsmaterial, bezeichnet man als Klartext. Eine abgesicherte Nachricht ist der Geheimtext, der allerdings auch Chiffrat oder Chiffretext genannt wird. Um einen Klartext in einen Geheimtext zu überführen benutzt man einen Algorithmus. Diesen Vorgang nennt man Verschlüsselung oder Chiffrierung. Die Zeichen, in denen der Geheimtext abgefaßt ist heißen Codezeichen, ihre Gesamtheit Code oder Chiffre.

Grundsätzlich ist das Verschlüsseln von einem Text oder einer Nachricht über zwei verschiedene Methoden möglich, zum einen durch Substitution und zum anderen durch Transposition. Die Substitution ersetzt entweder jedes Klarzeichen einzeln (mono-graphisch) oder jeweils ganze Zeichenfolgen (polygraphisch). Bei der Transposition besteht der Chiffretext aus einer Neuanordnung der ursprünglichen Zeichen.

Die Dechiffrierung oder Entschlüsselung erzeugt wiederum den ursprünglichen Text.

2 Die Vigenère-Verschlüsselung

2.1 Historische Einordnung

Als eine monoalphabethische Verschlüsselung (Cäsar-Verschlüsselung) auf Grund unzureichender Sicherheit nicht mehr ausreichte, weil Kryptoanalytiker in Arabien und Europa Häufigkeitsanalysen entwickelt hatten, mit deren Hilfe jene geknackt werden konnten, standen die Kryptographen nun vor der Aufgabe eine „stärkere Verschlüsselung“ zu entwickeln.

Leon Battista Alberti, der 1404 geboren wurde, hatte 1460 bereits die Idee beim Substitutionsverfahren das Geheimtextalphabet durch zwei oder mehrere zu ersetzen. Er setzte seine Überlegungen allerdings nicht in ein ausgereiftes Verschlüsselungssytem um. Eine in sich

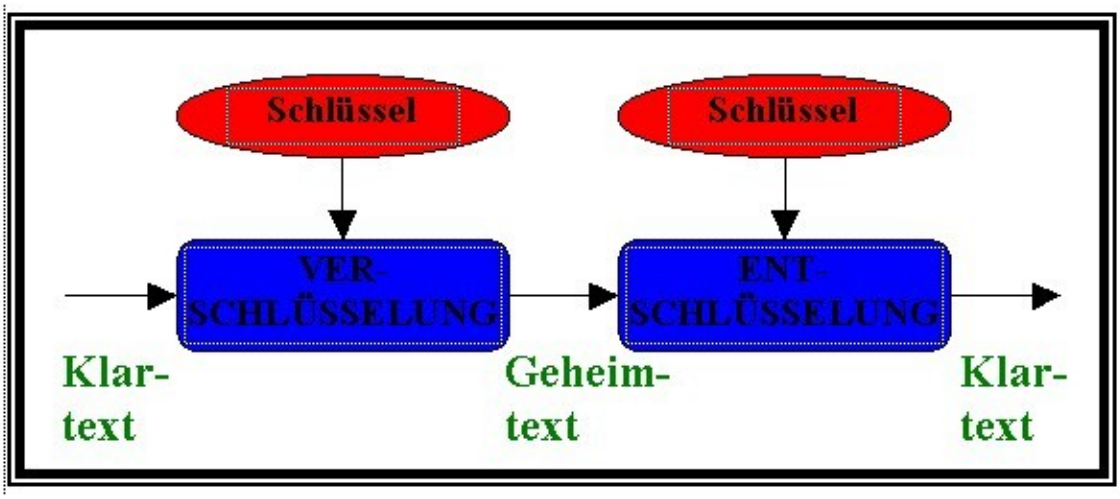
schlüssige und ausgereifte Gestalt nahm das Verfahren erst Ende des 16. Jahrhunderts an, nachdem zunächst Johannes Trithemius (*1492) und Giovanni Porta(*1535) wichtige Beiträge lieferten, welche der französische Diplomat Blaise de Vigenère (*1523) nach gründlichem studieren zu einem Verschlüsselungssystem verband. Dieses Verfahren machte er im Alter von 63 Jahren, also 1586 der Öffentlichkeit zugänglich. Es trägt den Namen Vigenère-Verschlüsselung (sprich: Wischenähr) und ist die heute wohl bekannteste polyalphabetische Chiffrierung. Obwohl sie früher keine große Rolle gespielt hat, wie man heute vermuten könnte. Es war zwar toll wieder eine sichere Chiffriermethode zu haben, doch vielen Leuten war die Anwendung der Vigenere-Verschlüsselung zu aufwendig, so dass sie die altbewährte monoalphabetische Verschlüsselung benutzten, da sie auch oft ausreichend war, weil schließlich nicht jeder dazu in der Lage war eine Kryptoanalyse durchzuführen, sondern nur Spezialisten auf dem Gebiet.

2.2 Polyalphabetische Verschlüsselung

Die Vigenère-Verschlüsselung ist im Gegensatz zur Cäsar-Verschlüsselung (monoalphabetisch [siehe oben]) eine polyalphabetische Substitution. Dies bedeutet, dass sie anstatt jedem Buchstaben nur 1 Zeichen zuzuordnen, jedem Buchstaben im Klartext verschiedene Zeichen im Geheimtext zuordnet. Somit erscheint das Knacken der Verschlüsselung, mit Hilfe von Häufigkeitsanalysen (fast) unmöglich, da diese ein relativ gleichmäßig verteiltes Ergebnis liefern würde.

2.3 Vigenère-Verfahren

Für die Vigenère-Verschlüsselung benötigt man ein sogenanntes Vigenère-Quadrat (siehe Anhang) und einen Schlüssel. Das Vigenère-Quadrat besteht aus 26 unterschiedlichen Geheimtextalphabeten, die immer um einen Buchstaben gegenüber dem vorigen versetzt sind. Der Schlüssel, den Sender und Empfänger vereinbaren müssen, besteht aus einem Wort.



Um den Vorgang der Vigenère-Verschlüsselung verständlich zu machen, zeigt man die Verwendung

von Quadrat und Schlüssel am besten an einem Beispiel. Verschlüsseln wir die Nachricht „Ich freue mich auf die Ferien“ mit Hilfe des Schlüsselwortes „Sonne“.

1) Zuerst wird der Klartext ohne Leerzeichen, also Buchstabe für Buchstabe aneinandergereiht, aufgeschrieben

2) Dann nehmen wir das Schlüsselwort und notieren dieses Zeichen für Zeichen über dem Klartext bis die Länge des Klartextes erreicht ist.

Schlüsselwort:	S	O	N	N	E	S	O	N	N	E	S	O	N	N	E	S	O	N	N
Klartext:	i	c	H	f	r	e	u	e	m	i	c	h	a	u	f	d	i	e	f

3) Jeder Buchstabe muss nun mit Hilfe des Vigenère-Quadrats einzeln verschlüsselt werden. Die Buchstaben des Schlüsselwortes bestimmen, welches Geheimtextalphabet verwendet wir (bestimmt die Zeile) und die Buchstaben des Klartext zeigen, welche Spalte benutzt wird. Der ergebene Schnittpunkt stellt den verschlüsselten Buchstaben dar.

4) In unserem Beispiel ist suchen wir das Geheimtextalphabet, also die Zeile, die mit „S“ beginnt und schauen nach, was in der Spalte „I“ steh; dies ist der Buchstabe „A“. Auf die gleiche Weise führt man dies mit allen anderen Buchstaben durch. Dann erlangt man folgenden Geheimtext:

Schlüsselwort:	S	O	N	N	E	S	O	N	N	E	S	O	N	N	E	S	O	N	N
Klartext:	I	c	h	f	r	e	u	e	m	i	c	h	a	u	f	d	i	e	F
Geheimtext:	A	Q	U	S	V	W	I	R	Z	M	U	V	N	H	J	V	W	R	S

In manchen Quellen steht dies auch vertauscht, also dass man mit dem Klartextbuchstaben die Zeile und mit dem Zeichen des Schlüsselwortes die Spalte ausfindig macht. Dies kommt einem im ersten Moment irritierend vor, doch man kann beides machen, da dasselbe herauskommt.

Bei der Entschlüsselung nimmt der Empfänger die Zeile, die mit dem Schlüsselwortbuchstaben beginnt. In dieser sucht er nach dem Geheimtextbuchstaben. Wenn er ihn gefunden hat, schaut er in jener Spalte nach dem ersten Buchstabe, da dieser den Klartextbuchstaben definiert.

Wenn man sich die Tabelle mit Schlüsselwort, Klartext und Geheimtext einmal genauer anschaut, sieht man wirklich, dass ein Buchstabe im Klartext, im Geheimtext durch verschiedene Zeichen dargestellt wird. Beispielsweise wird das „I“ einmal zu „A“, „M“ und zweimal zu „W“. Dadurch wird deutlich, dass man bei dieser gleichmäßigen Verteilung durch eine Häufigkeitsanalyse zu

keinem Ergebnis kommt.
Wenn man allerdings die Länge des Schlüsselwortes kennt, beispielsweise 3, dann muss man drei Häufigkeitsanalysen durchführen, die sich dann immer auf den jeweils dritten Buchstaben beziehen. Im Allgemeinen ist es so: je länger der Schlüssel, desto sicherer die Vigenère-Verschlüsselung. Es ist jedoch auch möglich, einen Text mit absoluter Sicherheit mit dem Vigenère-Chiffrat zu verschlüsseln, denn wenn man ein Schlüsselwort wählt, dass genauso lang ist, wie der eigentliche Klartext fehlen dem Kryptoanalytiker jegliche statistische Ansatzpunkte. Doch andererseits versucht man ein möglichst kurzes Schlüsselwort zu benutzen, da sie somit leichter und unauffälliger übermittelt werden können.

2.4 Umsetzung des Verfahrens in Delphi

Bei meiner Umsetzung des Vigenère-Verfahrens in Delphi kommt es mir nicht darauf an ein absolut durchdachtes und perfektes Programm zu entwickeln, sondern den Algorithmus in einem überschaubaren Rahmen umzusetzen.

Nachdem der Anwender den Schlüssel und den Klartext eingegeben auf den Button mit der Aufschrift „verschlüsseln“ geklickt hat passiert folgendes:

Zuerst wird der Schlüssel „aufgetragen“, das heißt der Schlüssel wird so oft aneinander geschrieben bis er die Länge des Klartextes erreicht hat. Der Vorgang des aneinander schreiben des Schlüssels wird durch die vordefinierte Funktion Concat gelöst, welche zwei oder mehr Strings zu einem einzigen String verknüpft.

Der eigentliche Algorithmus wird mit Hilfe der ASCII-Codetabelle umgesetzt.

Zunächst wird den Variablen *‚klarwert‘* (Wert des Klartextbuchstaben)und *‚schwert‘* (Wert des Schlüsselbuchstabens) der Wert der ASCII-Codetabelle des jeweiligen Großbuchstaben zugewiesen und dann 65 subtrahiert, weil das große ‚A‘ mit 65 beginnt und wir mit den Zeichenwerten von 0 bis 25 rechnen wollen.

Der *‚chiffwert‘*, also die zentrale Rechnung (gelb unterlegt) setzt sich aus der Summe des *‚klarwertes‘* und des *‚schwertes‘* zusammen, welche dann „durch mod 26 geteilt“ werden (d.h. es ergibt sich der ganzzahlige Rest der Division durch 26).

```
for i:=1 to klaenge do
  begin
  klarwert:=ord (upcase (ktext[i]))-65;
  schwert:=ord (upcase (schluessel[i]))-65;
  chiffwert:=((klarwert+schwert)mod 26);
  chiffre:=chr(chiffwert+65);
  memover.text:=memover.text+chiffre;
  end;
```

Daraufhin wird zum *chiffwert* wieder 65 addiert, um danach aus der Zahl mit Hilfe der vordefinierten Funktion CHR(i) und der ASCII-Codetabelle wieder einen Buchstaben zu machen, der nun den Geheimtextbuchstaben darstellt. Dies ist natürlich alles in eine FOR-Schleife eingebunden, damit jeder einzelne Buchstabe auf diese Weise verschlüsselt wird.

Das Entschlüsseln ist eigentlich ganz gleich aufgebaut nur mit dem Unterschied, dass hier die zentrale Rechnung wie folgt aussieht:

```
dechiffwert:=((klarwert-schwert+26)mod 26);
```

Um das Vigenère-Quadrat mit der richtigen Anordnung der Buchstaben zu füllen, also mit 26 unterschiedlichen Geheimtextalphabeten, habe ich zunächst ein ‚Array of char' deklariert, um auf der Datenstruktur arbeiten zu können. Und zwar wird das Quadrat Zeile für Zeile gefüllt, wofür die äußere FOR-Schleife verantwortlich ist. In jener gibt es zwei weitere FOR-Schleifen. Die Erste trägt die jeweiligen Buchstaben bis ‚Z' ein und die Zweite die Buchstaben nach dem ‚Z' (siehe Anhang: Struktogramm).

Der Anwender darf bei der Benutzung des Programms nur Klein- oder Großbuchstaben, also weder Zahlen noch Sonderzeichen, eingeben, um ein richtiges Ergebnis zu erlangen. Außerdem muss der Klartext beziehungsweise der verschlüsselte Text ohne Leerzeichen aneinander geschrieben werden. Es ist nämlich so üblich, dass das Chiffrat aus Großbuchstaben besteht, die alle aneinander gereiht werden, denn so bleibt auch die Länge der einzelnen Wörter geheim.

2.5 Quelltext

```
unit mvigenere3;

interface

uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms,
Dialogs,
 ComCtrls, Grids, StdCtrls, ExtCtrls, jpeg;

type
 TForm1 = class(TForm)
  PageControl1: TPageControl;
  TabSheet1: TTabSheet;
  TabSheet2: TTabSheet;
  edschluessel: TEdit;
  Label1: TLabel;
  Label2: TLabel;
  btclear: TButton;
  btclose: TButton;
  StringGrid1: TStringGrid;
  imvigenere: TImage;
  memoklar: TMemo;
  btverclear: TButton;
  btentclear: TButton;
  btklarclear: TButton;
  btver: TButton;
  memover: TMemo;
  memoent: TMemo;
  btent: TButton;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  TabSheet3: TTabSheet;
  Memo1: TMemo;
  Label6: TLabel;
  procedure PageControl1Change(Sender: TObject);
  procedure btcloseClick(Sender: TObject);
  procedure btclearClick(Sender: TObject);
  procedure btverClick(Sender: TObject);
  procedure btentClick(Sender: TObject);
  procedure btklarclearClick(Sender: TObject);
  procedure btverclearClick(Sender: TObject);
  procedure btentclearClick(Sender: TObject);

 private
 vigqua:ARRAY[1..26,1..26] of char;

 procedure verschluesseln;
 procedure entschluesseln;
```

```
 procedure ausgabe;
  { Private-Deklarationen }
 public
  { Public-Deklarationen }
 end;


var
  anzahl,rest,i,j, klaenge, slaenge:integer;
  ktext,schluessel,s:string;
 Form1: TForm1;

implementation

{$R *.DFM}
<<Verschlüsselung>>
procedure TForm1.btverClick(Sender: TObject);
begin
ktext:=(memoklar.text);
schluessel:=edschluessel.text;
klaenge:=length(ktext);
slaenge:=length(schluessel);
while slaenge<klaenge do
 begin
 schluessel:= concat(schluessel,schluessel);
 slaenge:=length(schluessel);
 end;
verschluesseln;
end;

procedure TForm1.verschluesseln;
var
klarwert,schwert,chiffwert:integer;
chiffre:string;
begin
for i:=1 to klaenge do
 begin
 klarwert:=ord (upcase (ktext[i]))-65;
 schwert:=ord (upcase (schluessel[i]))-65;
 chiffwert:=((klarwert+schwert)mod 26);
 chiffre:=chr(chiffwert+65);
 memover.text:=memover.text+chiffre;
 end;
end;

<<Entschlüsselung>>
procedure TForm1.btentClick(Sender: TObject);
begin
ktext:=(memover.text);
schluessel:=edschluessel.text;
klaenge:=length(ktext);
```

```
slaenge:=length(schluessel);
while slaenge<klaenge do
 begin
 schluessel:= concat(schluessel,schluessel);
 slaenge:=length(schluessel);
 end;
entschluesseln;
end;

procedure TForm1.entschluesseln;
var
klarwert,schwert,dechiffwert:integer;
dechiffre:string;
begin
for i:=1 to klaenge do
 begin
 klarwert:=ord (upcase(ktext[i]))-65;
 schwert:=ord (upcase(schluessel[i]))-65;
 dechiffwert:=((klarwert-schwert+26)mod 26);
 dechiffre:=chr(dechiffwert+65);
 memoent.text:=memoent.text+dechiffre;
 end;
end;

<<Vigenère-Quadrat>>
procedure TForm1.PageControl1Change(Sender: TObject);
var s, z:integer;
begin
i:=0;
for z:=1 to 26 do
 begin
 for s:=1 to 26-i do
  vigqua[s,z]:=chr(s+64+i);
 j:=0;
 for s:=26-i+1  to 26 do
  begin
  j:=j+1;
  vigqua[s,z]:=chr(64+j)
  end;
 i:=i+1;
 end;
ausgabe;
end;

procedure TForm1.ausgabe;
begin
for i:=1 to 26 do
for j:=1 to 26 do
stringgrid1.cells[i-1,j-1]:=vigqua[i,j];
end;
```

```
<<alles löschen>>
procedure TForm1.btclearClick(Sender: TObject);
begin
memoklar.text:='';
edschluessel.text:='';
memover.text:='';
memoent.lines[0]:='';
end;

<<einzeln löschen>>
procedure TForm1.btklarclearClick(Sender: TObject);
begin
memoklar.text:='';
end;

procedure TForm1.btverclearClick(Sender: TObject);
begin
memover.text:='';
end;

procedure TForm1.btentclearClick(Sender: TObject);
begin
memoent.text:='';
end;

end.

<<Ende>>
procedure TForm1.btcloseClick(Sender: TObject);
begin
close;
end;
```

3 Schlusswort

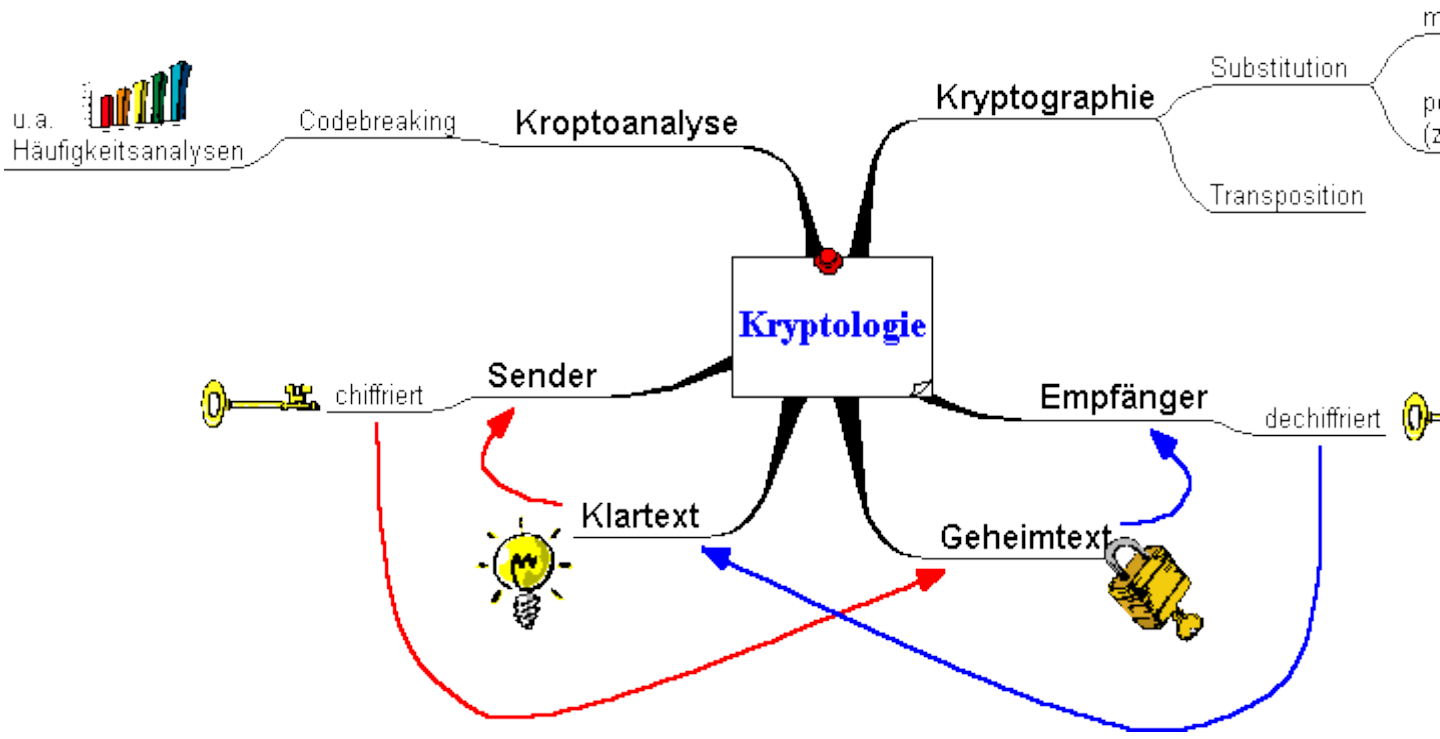
Die Möglichkeiten, Nachrichten zu verschlüsseln, sind enorm.

Das behandelte Vigenère-Verfahren ist zwar aus heutiger Sicht relativ simpel, doch hat man relativ lange gebraucht. 1854 wurde das Verfahren von Charles Babbage einem englischen Mathematiker geknackt. Doch erst der preußische Infanteriemajor Friedrich Wilhelm Kas(s)iski veröffentlichte den nach ihm benannte Kas(s)iski-Test. Den zweiten Angriff machte 1925 Colonel William Frederick Friedman. Die beiden Verfahren, der Kasiski-Test und der Friedmann-Test, gelten als besonders wichtig in der Kryptoanalyse. Obwohl die Unsicherheit des Vigenère-Algorithmus somit natürlich bekannt ist, wird er sogar in moderner Software noch eingesetzt, zum Beispiel bei der Textverarbeitung WordPerferct.

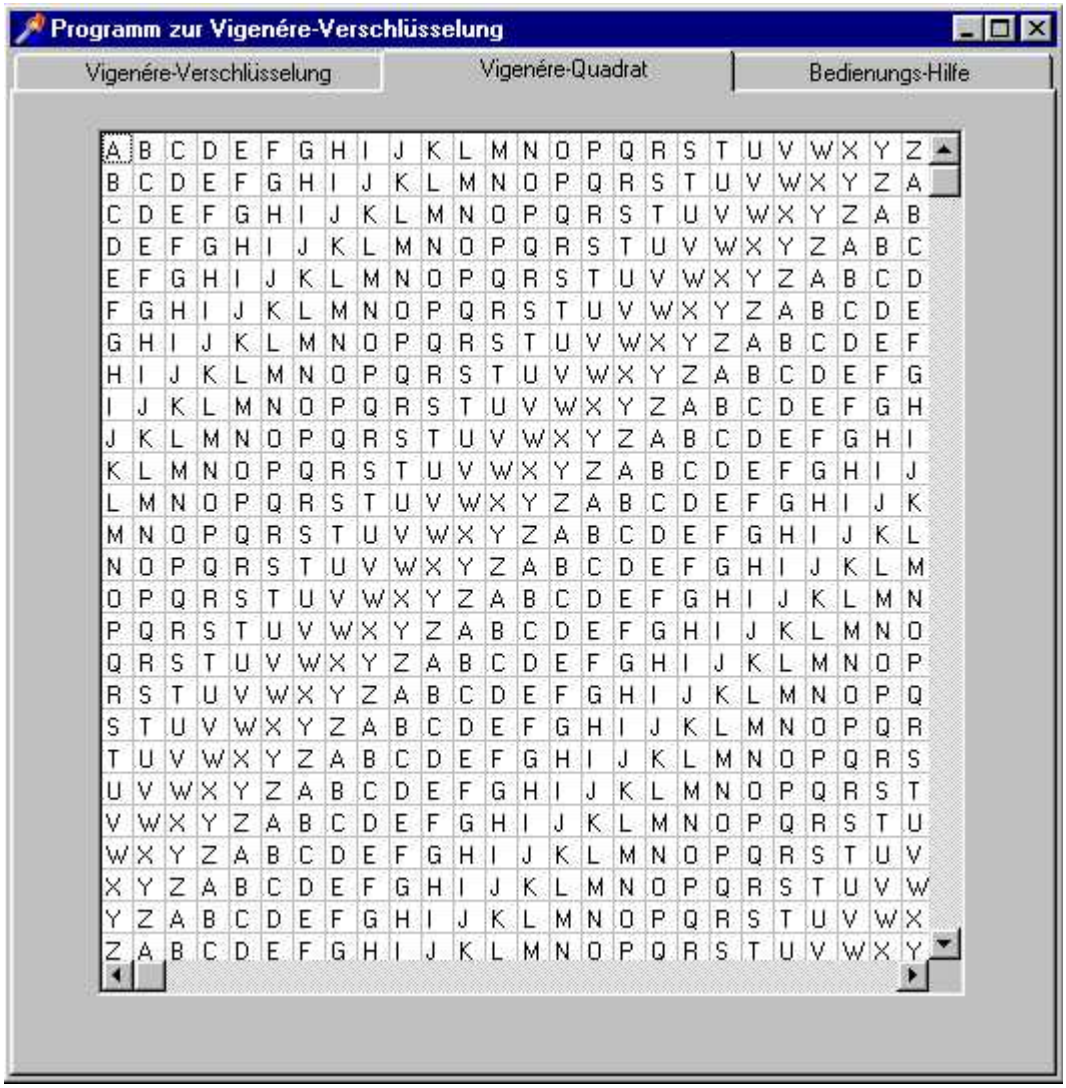
Die Umsetzung des Verfahrens in Delphi hat im Grund genommen eine leicht zu überschaubare Struktur, doch musste man sich erst einmal überlegen, wie man das ganze überhaupt angeht. Ich habe drei verschiedene Versionen begonnen und eine davon zu Ende geführt, denn man muss sich schließlich seine Gedanken darüber machen, wie man den Schlüssel aufträgt, die Ver- und Entschlüsselung angeht und wie man das Array so mit Buchstaben füllt, dass sich das Vigenère-Quadrat ergibt.

4 Anhang

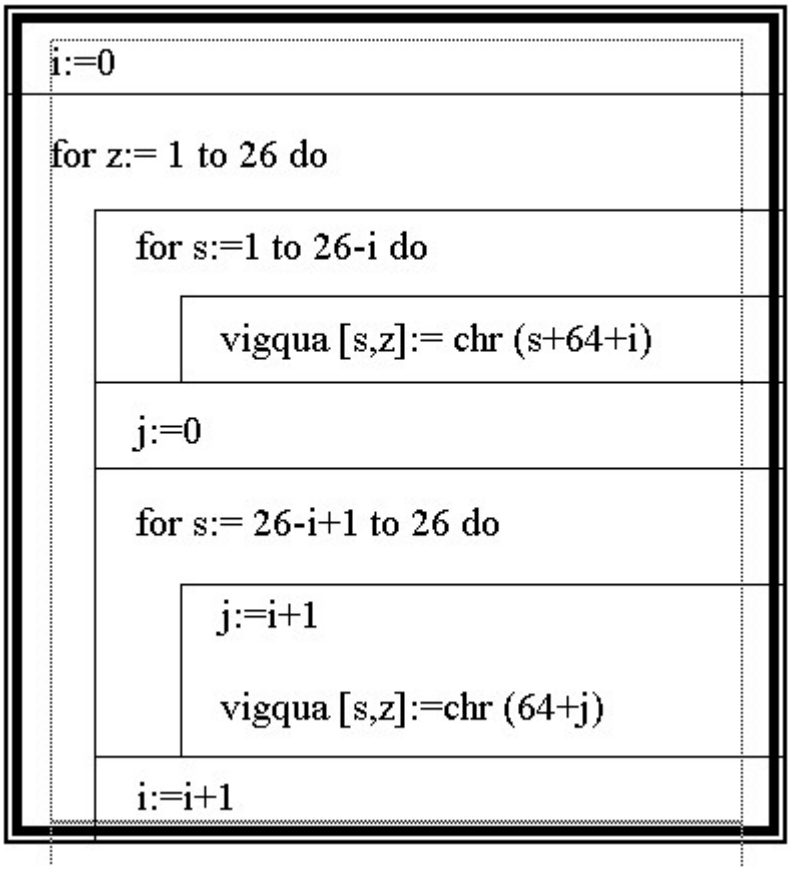
4.1 MindMap



4.2 Vigenère-Quadrat

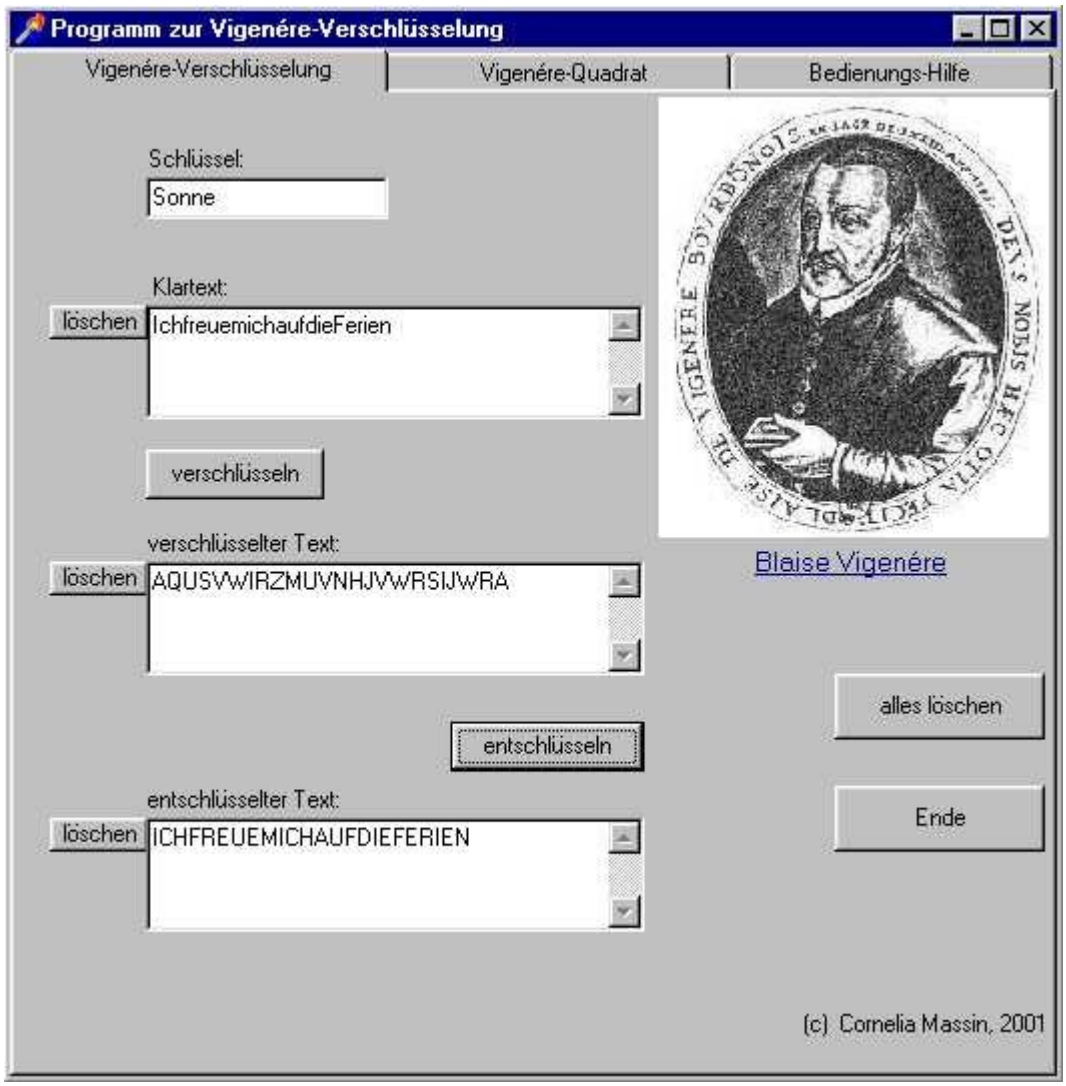


4.3 Struktogramm



4.4 Bedienungshilfe des Programms

4.5 Haupt-Formular-Seite des Programms



4.6 Tagebuch

- 22.01. Aushang der zugewiesenen Kurse
- 14.02. Materialsuche in der Bibliothek
- 16.02. Abgabe des Themas
- 18.02. Materialsuche im Internet
- 08.03. Beratungstermin in der Such- & Ordnungsphase
- 10.03. Ideensammlung zur Umsetzung der Implementation
- 11.03. Materialsuche im Internet & Ordnung des Materials

- 17.03. Programmierung in Delphi
- 18.03. Programmierung in Delphi
- 25.03. Gliederung & Einleitung schreiben
- 26.03. Programmierung in Delphi
- 29.03. Beratungstermin in der Schreibphase
- 31.03. Erstellung des Textes
- 01.04. Erstellung des Textes
- 02.04. Erstellung des Textes
- 06.04. Abgabe der Facharbeit

5 Quellenverzeichnis

- Singh, S., Geheime Botschaften, Die Kunst der Verschlüsselung von der Antike bis in die Zeiten des Internets, Carl Hanser Verlag München Wien 2000
- Beutelspacher, A., Was ist ein Verschlüsselungssystem?, in: Mathe-Welt 1995, Seite 8
- Beutelspacher, A., Die Vigenère-Verschlüsselung, in Mathe-Welt 1995, Seite 12 (aus AKG, Berlin)
- www.mjonet.de
- www.it.fht-esslingen.de/~schmidt/vorlesungen/inco/seminar/historie/Nav_Frame..
- http://eiche.theoinf.tu-ilmenau.de/~aaver/lehre/hs_ws97/vignere.html
- www.mathematik.uni-kassel.de/~project/1_2000/spwg4/vigenere.html
- www.iti.fh-flensburg.de/lang/algorithmen/code/krypto/klassisch.htm
- www.rg18.asn-wien.ac.at/rg18/inform/krypt/krypt.htm
- http://home.fhtw-berlin.de/~s0291172/Semesterarbeit/Mathematische%20Grundlagen.htm
- www.mjonet.de
- www.internetwahlen.de/schluessel.html

6 Erklärung

„Ich erkläre, dass ich die Facharbeit ohne fremde Hilfe angefertigt und nur die im Literaturverzeichnis angeführten Quellen und Hilfsmittel benutzt habe.“
