DiffieHellmann-Verfahren, Schlüsselvereinbarung

Kryptographie mit MuPAD,
Prof. Dr. Dörte Haftendorn, Mathematik mit MuPAD 4.02, (Vorlesungs-Version)
(ex. in 2.5 Okt 99 , vom Nov.02 und in 3.11 Sept. 05) Juni 07
http://haftendorn.uni-lueneburg.de **www.mathematik-verstehen.de**
##
----------eigene Zahlentheorie Ergänzungen----------
Im Dateimenu bei "Eigenschaften" sehen die beien Prozeduren zur Umwandlung von Text in Zahl und Zahl in Text, daher können sie hier ausgeführt werden.
----------eigene Zahlentheorie Ergänzungen----------

Anton will mit Berta einen gemeinsamen Schlüssel vereinbaren.

Er bereitet als Grundlage p und g vor.

```
saat:=random(10000..99999):
p:=numlib::prevprime(floor(sqrt(saat()*10^50)));
//Exponent auch 150
```

$$2215423210133901255819665797$$

```
g:=numlib::prevprime(floor(sqrt(saat()*10^50)));
```

$$25153528579505500464455222 39$$

```
//p:=29:g:=27:
```

Er teilt dieses Berta mit, jeder darf das wissen.
Anton wählt sich eine beliebige Zahl a und berechnet:

```
r:= random(2..p-2): a:=r():  anton:=powermod(g,a,p);
```

$$19693755871757693449863426 35$$

Berta wählt sich eine beliebige Zahl b und berechnet:

```
r:= random(2..p-2): b:=r():     berta:=powermod(g,b,p);
```

$$16072636568746807103952984 88$$

Anton und Berta senden sich gegenseitig öffentlich ihre Ergebnisse.
Anton berechnet:

```
ka:=powermod(berta,a,p);
```

$$12477538165586805321890542 30$$

Berta berechnet:

```
kb:=powermod(anton,b,p);
```

$$12477538165586805321890542 30$$

```
//kb:=kb+1://Störung einbauen
if (ka=kb) then k:=ka;
 print("Der gemeinsame geheime Schlüssel für Anton und Berta ist") ;
   print("k = ".k) ;
else print("Vorsicht, Protokoll ist gestört") end_if;
"Der gemeinsame geheime Schlüssel für Anton und Berta ist"
"k = 12477538165586805321890542 30"
```

Diesen Vergleich kann beim Ablauf niemand machen. Die Manipulation wird dadurch gemerkt, dass man
später nicht entschlüsseln kann.

Berta will Anton einen Text senden, den nur Anton lesen kann.[1]

```
mText:="Montag im Medley":
```

----------eigene Zahlentheorie Ergänzungen----------
Im Dateimenu bei "Eigenschaften" sehen die beien Prozeduren zur Umwandlung von Text in Zahl und Zahl in Text, daher können sie hier ausgeführt werden.

Im Dateimenu bei "Eigenschaften" sehen die beien Prozeduren zur Umwandlung von Text in Zahl und Zahl in Text, daher können sie hier ausgeführt werden.
----------eigene Zahlentheorie Ergänzungen----------

```
m:=txToZoo(mText);
```
[77, 111, 110, 116, 97, 103, 32, 105, 109, 32, 77, 101, 100, 108,

498382886975047781044973728073 93

```
delete f,kk,x:      f:=(kk,x)->(kk*x):      f(kk,x);
```
kk · x

```
c:=f(kb,m);
```
6218591493306493821103353948058626134019126512620062192239 0

Sie teilt Anton den Algorithmus von f(k,c) und c mit. f muss invertierbar sein.
f ist hier einfach, in der Praxis "verrührt" f die Nachricht heftig.
Den gemeinsamen Schlüssel k weiß Anton schon, k bleibt geheim.

```
delete f_inv,kk,x:      f_inv:=(kk,x)->(x/kk):  f_inv(kk,x);
m_vonBerta:=f_inv(k,c);
```
$\frac{x}{kk}$

498382886975047781044973728073 93

Der Ascii-Text muss noch in Klartext verwandelt werden.

```
klar:=zooToTx(m_vonBerta);
```
[49, 83, 82, 88, 69, 75, 4, 77, 81, 4, 49, 73, 72, 80, 73, 93]
[77, 111, 110, 116, 97, 103, 32, 105, 109, 32, 77, 101, 100, 108, 101, 121

"Montag im Medley"

Anton will Berta Antworten:

```
mAntwort:="ja, um 10":    m:=textToZahl(mAntwort);
```
textToZahl("ja, um 10")

```
c:=f(ka,m);
```
1247753816558680532189054230 · textToZahl("ja, um 10")

Berta liest

```
m_vonAnton:=f_inv(kb,c); klar:=zahlToText(m_vonAnton);
```
textToZahl("ja, um 10")

zahlToText(textToZahl("ja, um 10"))

Anton und Berta können nun beliebig oft mit dem einmal berechneten
Schlüssel k Nachrichten austauschen.