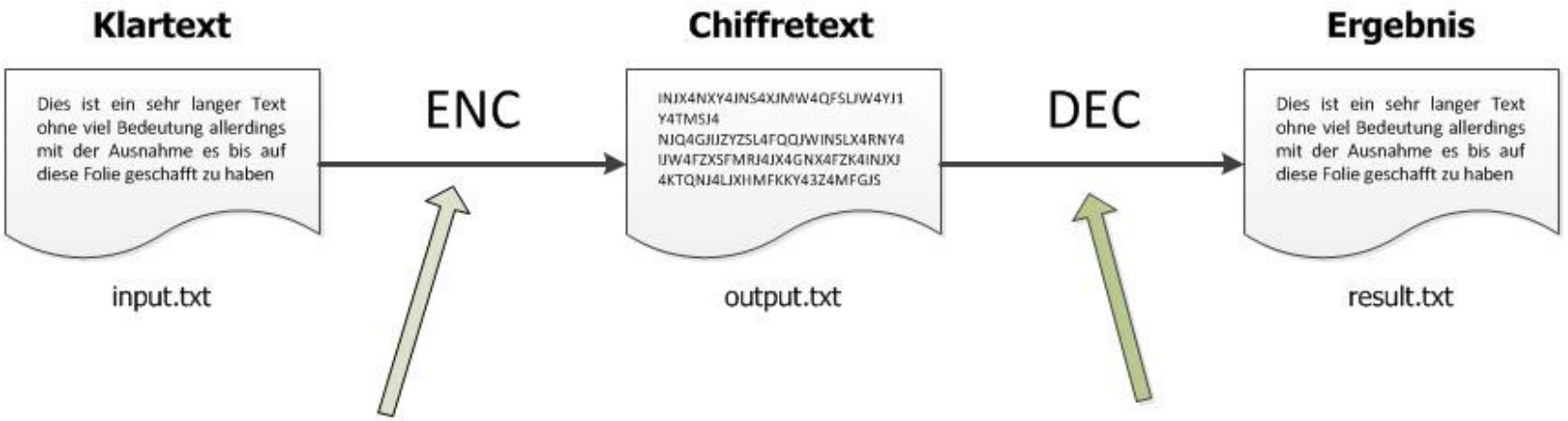
Beispielvorgang Ver- / Entschlüsselung mit Verschiebechiffre



Kommando: `Chiffrenarbeiter -e v 5`

`Chiffrenarbeiter -d v 5`

Der Inhalt der Datei input.txt wird verschlüsselt (**-e**)
Ein Verschiebechiffre (**v**) mit Schlüssel **5**
Das Chiffrat wird in die Datei output.txt geschrieben

Der Inhalt der Datei output.txt wird entschlüsselt (**-d**)
Verschiebechiffre (**v**) mit Schlüssel **5** angenommen
Das Ergebnis wird in die Datei result.txt geschrieben

Implementiert sind die Funktionalitäten:

- **Verschiebechiffre:** **z --> (z +k) mod n**
- **Multiplikat. Chiffre:** **z --> (z * t) mod n**
- **Tauschchiffre:** **z --> (z * t + k) mod n**
- **und Substitutionschiffre mittels Subst.-Tabelle (in Datei „subst.cfg“)**
- **Erzeugung eines MD5-Hashes**

```
usage: java -jar Chiffrenarbeiter.jar <option> <param1> <param2> [<param3>]

OPTION:
-e   encrypt text from input textfile

        PARAMETER 1:
         v   Verschiebechiffre
         m   Multiplikative Chiffre
         t   Tauschchiffre
         s   Substitutionschiffre nach Subst.-Tabelle

        PARAMETER 2:
         [1..36]   Schlüsselwert

        PARAMETER 3: (bei Tauschchiffre)
         [1..36]   Schlüsselwert2 (additiver)

OPTION:
-d   decrypt text from output textfile

        PARAMETER 1:
         v   Verschiebechiffre
         m   Multiplikative Chiffre
         t   Tauschchiffre
         s   Substitutionschiffre nach Subst.-Tabelle

        PARAMETER 2:
         0   alle Schlüsselwerte  (bei Subst.chiffre wird Häufigkeitsanalyse u. Substitutionstabelle erstellt)
         [1..36]   Schlüsselwert  (1: bei Subst.chiffre wird aufgrund Tabelle 'subst.cfg' ein Versuch ausgegeben)

        PARAMETER 3: (bei Tauschchiffre)
         [1..36]   Schlüsselwert2 (additiver)

OPTION:
-md5   generates MD5-Hash for textstring in <param1>

        PARAMETER 1:
         '...'   Textstring in einfachen Anführungszeichen
```

Demo 1:

Verschiebechiffre um k = 5

* `>java -jar './dist/Chiffrenarbeiter.jar' -e v 5`

Dechiffrierung erfolgt mit:

```
>java -jar './dist/Chiffrenarbeiter.jar' -d v 5
```

---

* Aufruf im Projektverzeichnis von Chiffrenarbeiter

Demo 2:

Multiplikative Chiffre mit t = 4, n = |Alphabet| = 37

```
>java -jar './dist/Chiffrenarbeiter.jar' -e m 4
```

Dechiffrieren – 1. Möglichkeit „brute force“ (2. Parameter = 0)

```
>java -jar './dist/Chiffrenarbeiter.jar' -d m 0
```

Ausgabe:

```
Multiplikative Chiffre anwenden, Schlüssel: 1
I1M4 148 M1L 4MY0 D6LUM0 8MO8 PYLM G1MD AMIMC8CLU 6DDM0I1LU4 H18 IM0 6C4L6YHM M4 A14 6CQ I1M4M QPD1M UM4EY6QQ8 WC Y6AML
  (Multiplikative Chiffre)
Multiplikative Chiffre anwenden, Schlüssel: 2
0389 39H 836 98W1 RD6O81 H8CH EW68 X38R L808PHP6O DRR81036O9 Z3H 081 DP96DWZ8 89 L39 DPG 03898 GER38 O89TWDGGH SP WDL86
  (Multiplikative Chiffre)
Multiplikative Chiffre anwenden, Schlüssel: 3
J5VE 5EQ V5S EVU2 4KSIV2 QV0Q 3USV D5V4 WVJV1Q1SI K44V2J5SIE G5Q JV2 K1ESKUGV VE W5E K16 J5VEV 6345V IVE7UK66Q O1 UKWVS
  (Multiplikative Chiffre)
Multiplikative Chiffre anwenden, Schlüssel: 4
17HJ 7JZ H7D JHS3 IRDCH3 ZHPZ TSDH U7HI 6H1HEZEDC RIIH317DCJ Y7Z 1H3 REJDRSYH HJ 67J REX 17HJH XTI7H CHJMSRXXZ KE SR6HD
  (Multiplikative Chiffre)
Multiplikative Chiffre anwenden, Schlüssel: 5
K93O 9O7 39Z O3Q4 WYZ634 73D7 IQZ3 A93W H3K3R7RZ6 YWW34K9Z6O F97 K34 YROZYQF3 3O H9O YRN K93O3 NIW93 63O0QYNN7 GR QYH3Z
  (Multiplikative Chiffre)
Multiplikative Chiffre anwenden, Schlüssel: 6
2BQT BTG QBK TQO5 94K0O5 GO1G 7QKO RBO9 SO2O3G3K0 499O52BK0T XBG 2O5 43TK4QXQ QT SBT 43D 2BQTO D79BQ 0QTFO4DDG C3 O4SQK
```

Dechiffrieren – 2. Möglichkeit „Kenntnis von t=4“

--> Schlüssel-Parameter kann berechnet werden

Suche das multiplikativ Inverse zu t=4, so dass gilt: $\mathbf{t * t^{-1} = 1 \bmod 37}$

$t^1 = 4$, $t^2 = 16$, $t^4 = 16*16 = 256 \bmod 37 = 34$,

$t^8 = 34*34 = 1156 \bmod 37 = 9$, $t^{16} = 9*9 = 81 \bmod 37 = 7$,

$t^{32} = 7*7 = 49 \bmod 37 = 12$

Nach Euler gilt $\varphi(n) = n - 1$, wenn n Primzahl, also ist $\varphi(37) = 35$,

weiterhin gilt $t^{\varphi(n)} = 1 \bmod n$ und $1 = t * t^{-1}$

$t^{\varphi(37)-1} = t^{-1} \leftrightarrow t^{35} = t^{-1}$

Bestimme $\mathbf{t^{-1}} = \mathbf{t^{35}} = t^{32} * t^2 * t = 12 * 16 * 4 = 768 \bmod 37 = \mathbf{28}$

Schlüssel zum erfolgreichen Dechiffrieren ist 28:

```
>java -jar './dist/Chiffrenarbeiter.jar' -d m 28
```

Demo 3:

Substitutionschiffre nach Tabelle, Inhalt der Datei „subst.cfg“:

```
0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789BHZTRUVWDSFCXYPLKMNIAQEOGJ
```

```
>java -jar './dist/Chiffrenarbeiter.jar' -e s
```

wendet die Substitutionstabelle auf den Inhalt der Datei input.txt an
und schreibt den Chiffretext in die Datei output.txt

Für die **Kryptoanalyse** wird der folgende Befehl verwendet,

```
>java -jar './dist/Chiffrenarbeiter.jar' -d s 0
```

Aktionen: Häufigkeitsanalyse und Erstellung anfängl. Substitutionstabelle

```
>java -jar './dist/Chiffrenarbeiter.jar' -d s 1
```

wendet „subst.cfg“ auf den Chiffretext in output.txt an