10.3　暗号化と復号化

暗号化・復号化の仕組みが分かったところで **Python3** を使ってみよう。鍵となる n の値は平文の長さによるが、相当大きな数になってしまう。それだと非力なコンピュータでは暗号体験ができないので、短い平文を使うことにする。ここでは平文 “TeX” を暗号化しよう。このサイト付近の散歩にはふさわしいかもしれないね。

まず、平文を ascii コードにしよう。それには前々節で書いたスクリプト `chr2ord()` の出番だ。

[py script]
```
>>> chr2ord('TeX')
8410188
```

これで平文の “TeX” が ascii コードになった。この数字列は、まだ暗号化されてはいないよ。暗号化に必要なのはこれより大きいふたつの素数の積である。この場合は適当な 3, 4 桁の素数があればよいので、以前書いたスクリプト `pmsg()` を使っておこう。

[py script]
```
>>> pmsg(3331)
素数
>>> pmsg(4567)
素数
>>> 3331*4567
15212677
```

一発で素数を見つけたような雰囲気が漂うけれど、実は 2、3 の試行錯誤をしているのだ。そして 2 数の積が、平文の数字列より大きいことと互いに素であることも、確認しておかなくてはいけない。どうやら大丈夫のようだ。

次は $\varphi(n)$ と互いに素になる数を探そう。これも以前書いたスクリプト `intfacto()` を使えば十分だろう。ああ、これまでに見た風景が蘇（よみがえ）るね。

[py script]
```
>>> intfacto((3331-1)*(4567-1))
2 * 2 * 3 * 3 * 3 * 5 * 37 * 761
```

素因数には 7 がないから $r = 7$ でいいか。さあ、暗号化の準備は整った。平文の 8410188、鍵となる 15212677 と 7 を使って暗号化だ。

[py script]
```
>>> 8410188**7 % 15212677
10589986
```

よし、暗号の出来上がり。r に小さい数を選んだので暗号作成は時間がかからない。実際の暗号化では大きな r を選ぶものだ。

そういうわけで、だれかが私が指定した公開鍵を用いて暗号

（暗号）10589986：　（鍵）$n = 15212677$、$r = 7$

を送ってきたとしよう。もし、何者かがこれを傍受して解読しようとしても、n が素因数分解できなければ復号はできない。しかし、私は鍵を作った本人だから $n = 3331 \times 4567$ であることを知っている。だから $\varphi(n) = (3331-1) \times (4567-1) = 15204780$ も直ちに分かる。よって

$$7t \equiv 1 \pmod{15204780}$$

が解ければよいのだ。さすがにこれはスクリプトを書いて解くべきだろう。

[py script]
```
>>> def solvemod(r, m):
...   t = 1
...   while (r*t % m) != 1:
...     t += 1
...   print(t)
...
>>> solvemod(7, 15204780)
4344223
```

愚直に t を 1 ずつ増やして合同式が成り立つかどうか調べているだけなので、まったく効率が悪い。コンピュータの性能によっては数秒要するかもしれない。でも t の値が分かったので復号できるぞ。復号には暗号文 10589986 と鍵 15212677、それといま求めた値 4344223 を使う。

[py script]
```
>>> 10589986**4344223 % 15212677
8410188
```

と言っても、こっちの計算はだいぶ時間がかかる。指数がバカでかいからだ。出力された数字列を見れば元どおりになったことが分かるが、念のため以前書いたコード `ord2chr()` で確認しておこう。

[py script]
```
>>> ord2chr('8410188')
TeX
```

うむ。なかなかよい具合である。ただし、このようにちまちまと細かい計算を繰り返すのは手間である。流れ作業ができるようにしたいものだね。この場合なら、たとえば `encrypt('TeX')` で `10589986` が出力され、`decrypt(10589986)` で `TeX` が出力されるのが理想だ。

具体的には、暗号スクリプトは `encrypt(平文)` のように文字列を引数にしたい。しかし、そのためには平文を数字列にしたとき、その数字列より大きくなるふたつの素数積が必要になる。あらかじめ用意したいが、平文が何桁の数字列になるか分からなければ、ふたつの素数は選びようがない。ここは工夫がいるだろう。

また、復号スクリプトは与えられた鍵をもとに解くのだから、単に `decrypt(10589986)` ではダメで、`decrypt(暗号, 鍵 n, r)` のような引数をとる必要がある。その上で、引数 `n` はスクリプト中で素因数分解しなくてはならない。

まあ、いまやった手作業をまとめてスクリプトにするだけの話なので、そんなに難しいことではない。