

Feuille d'exercices n° 4
Logarithme discret. Factorisation

Exercice 1. On suppose qu'il existe un algorithme **A** qui résout le problème du logarithme discret en temps polynomial pour tout entier premier p et racine primitive g modulo p pour un sous-ensemble $B_p \subset (\mathbb{Z}/p\mathbb{Z})^*$, avec $|B_p| \geq \epsilon |(\mathbb{Z}/p\mathbb{Z})^*|$.

Algorithme A : input (p, g, y) ; output t tel que $g^t = y \pmod p$ si $y \in B_p$; out of range sinon.
On considère l'algorithme probabiliste **B** décrit par le pseudo-code suivant :

Algorithme B : input (p, g, y)
begin repeat
 choose $c \in (\mathbb{Z}/p\mathbb{Z})^*$ at random
 $z \leftarrow g^c \pmod p$
 $w \leftarrow A(p, g, yz)$
 if $g^w = yz \pmod p$ **then output** $w - c$ **end if**
end repeat

1. Montrer que l'algorithme **B** résout le problème du logarithme discret en général avec un temps d'espérance polynomiale en la taille de p et en $\frac{1}{\epsilon}$.
2. Soit $\text{logd}(p, g, \cdot)$ la fonction logarithme discret modulo p par rapport à la racine primitive g . Justifier l'hypothèse suivante sur le problème du logarithme discret : pour tout polynôme à coefficients positifs et pour tout algorithme probabiliste **A**, on a pour k suffisamment grand :

$$\mathbf{P}[\mathbf{A}(p, g, y) = \text{logd}(p, g, y)] < \frac{1}{q(k)}.$$

Exercice 2. Montrer que, pour k suffisamment grand, la probabilité qu'un entier s'écrivant sur k bits soit premier est au moins égal à $\frac{1}{k}$. On utilisera le théorème des nombres premiers.

Exercice 3. Soit p un nombre premier impair, et soit N un entier non multiple de p . On s'intéresse aux nombre d'entiers $x \pmod p$ tels que l'équation

$$N = x^2 - y^2 \pmod p \tag{1}$$

a une solution $y \pmod p$.

1. Remarquer que si (x, y) est solution de (1), il en est de même pour $(x, -y)$.
2. Que dire de y_1 et y_2 si (x, y_1) et (x, y_2) sont solutions de (1) ?
3. Montrer que l'équation (1) est équivalente à une congruence $N = uv \pmod p$ qui détermine entièrement x et y . Dénombrer le nombre de solutions (u, v) à cette congruence.
4. En déduire que le nombre d'éléments $x \pmod p$ pour lesquels l'équation (1) a une solution y est égal à $\frac{p-1}{2}$ ou $\frac{p+1}{2}$.
5. En déduire une information quand à l'usage d'un crible pour trouver solutions dans \mathbb{Z} à $N = x^2 - y^2$ en vue de factoriser l'entier N .

Exercice 4 (algorithme $p - 1$ de Pollard). Soit n un entier impair, et soit $B > 0$ une borne fixée. Soit p un nombre premier divisant n . On suppose que toute puissance q^α d'un nombre premier q vérifiant $q^\alpha | p - 1$ est au plus égale à B .

1. Soit a l'entier $a = 2^{B!}$. Montrer qu'il existe un algorithme calculant $a' = a \pmod n$ en $2(B-1) \log_2 B$ multiplications modulaires.

2. Montrer que $p - 1 | B!$ puis que $a' = 1 \pmod p$. En déduire que le pgcd $(a' - 1) \wedge n$ est un facteur non trivial de n , sauf si $a' = 1$.
3. En déduire un algorithme de factorisation. Cet algorithme a été proposé en 1974 par Pollard, qui l'a baptisé *algorithme $p - 1$* .
4. Déduire de ce qui précède :
 - (a) une attaque contre RSA ;
 - (b) une précaution à prendre lorsqu'on choisit un module RSA pour ne pas être sujet à l'attaque précédente.