Number theory in cryptography

– Exercise set 7 –

The exercises T7.1, T7.3 have to be handed in on Tuesday, 16th April 2024, 8:30 at latest.

Theoretical questions

T 7.1 Let $\psi(x, y)$ be the number of y-smooth integers in the interval $[1, x]$ as introduced in the lecture notes on integer factorization. Let f be a real-valued function defined for $y \geqslant 2$ and satisfying $f(y) \geqslant 1$ for all y and $f(y) = y^{1+o(1)}$ as $y \to \infty$.

We let $y = L_x(1/2, v)$ for some parameter $v > 0$. Prove that as $x \to \infty$, we have

$$\frac{xf(y)}{\psi(x, y)} \sim L_x(1/2, g(v) + o(1))$$

for some function $g : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ satisfying $g(v) \geqslant \sqrt{2}$ for all $v > 0$.

T 7.2 Various algorithms for both integer factorization and discrete log (to be seen later in class) produce auxiliary numbers up to some bound x via random samples and then test these numbers for smoothness. Suppose that our goal is to find $y^{1+o(1)}$ numbers that are y-smooth for some $y < x$ (that depends on x).

How would you choose y (using the L-notation for the complexity parameter x) so that you minimize the number of samples needed to achieve the goal? Show that there is a choice of y for which $L_x(1/2, \sqrt{2})$ samples are sufficient for achieving the goal.

T 7.3 Define $\mathsf{L}_{\alpha,C}(X) := \exp\left(C \log(X)^\alpha \cdot \log(\log(X))^{1-\alpha}\right)$ for $C > 0, X > 1$ and $0 < \alpha < 1$. The goal is to show that $\mathsf{L}_{\alpha,C}(X)$ is subexponential in $\log(X)$, namely prove the following claims:

a) For all $\varepsilon > 0$, $\mathsf{L}_{\alpha,C}(X)$ is smaller than a constant times X^ε for X large enough.

b) For all $N > 0$, $\mathsf{L}_{\alpha,C}(X)$ is bigger than a constant times $(\log X)^N$ for X large enough.

T 7.4 Let $\alpha, \beta, r, s \in \mathbb{R}_{>0}$ be given with $s < r \leqslant 1$. Show that the probability that a random positive integer less than or equal to $L_x(r, \alpha)$ is $L_x(s, \beta)$-smooth is

$$L_x(r - s, -\alpha \cdot (r - s)/\beta)$$

as $x \to \infty$.

Programming exercises

P 7.1 **Bonus question: Easter Egg Hunt!** You intercept the Easter bunnies' communications. You know the bunnies are using RSA with $e = 11$ and

$$\begin{aligned} N = {} & 1785062011465543289425904041086005515261568965792206154594175352446844 8\\ & 526627298622559103756785243084279882691015984535250507088923794975161 76\\ & 845658733593322890112120474537973688547175234110015445380180370060705 02\\ & 00128340099072001 \end{aligned}$$

and you intercepted the ciphertext

$$
\begin{aligned}
c = {} & 29580696160778606524129453894167180894514908118024335930481312596390336 93\\
& 7851207490695949380127520594929942416499705760712639406573651432137036436 7\\
& 8443992122917744756916065170023433286838397457926726324853293317677163759 4\\
& 24201496
\end{aligned}
$$

You also know the message was encoded using a shift of ASCII: every block of 2 plaintext digits (which will be a number k between 10 and 99) encodes the character with ASCII code $k + 22$ if $k \leqslant 73$ and $k + 23$ otherwise. (e.g., the 12-character string `Hello World!` corresponds to the 24-digit integer 507885858810658891857711). Can you recover the plaintext?