# Mandatory Problem 3:
# Digital Signature Algorithm and Discrete Logarithms

The deadline is Sunday, November 1, midnight. You have to hand in a short description of the algorithms you implemented, the implementation code as a collection of subroutines (functions), and computational results.

1. Let the parameters of ElGamal Signature Algorithm be

$$\begin{aligned} p &= 593831971123607, \\ g &= 13, \end{aligned}$$

   where $p$ is prime and $g$ is a primitive root modulo $p$. Let $x \mod p-1$ be a system private key and $y \equiv 13^x \mod p$ be the system public key, where $y = 239505966643112$.

   Forge an ElGamal signature without knowledge of the private key, by constructing a triple $m, a, b$, where $m$ is an integer and $a, b$ is its signature.

2. Let the parameters of DSA(Digital Signature Algorithm) be

$$\begin{aligned} p &= 949772751547464211, \\ q &= 4748626326421, \\ g &= 314668439607541235, \end{aligned}$$

   where $p, q$ are primes, $q|p-1$ and $g$ is a residue modulo $p$ of order $q$. Let $x \mod q$ be a system private key and

$$y \equiv g^x \mod p \tag{1}$$

   be the system public key, where

$$y = 254337213994578435.$$

   (a) Let

$$m_1 = 23939231686113389855511149, m_2 = 93308042764066398743879\,38$$

be two messages(hash values) encoded by integers and their DSA signatures

$$
\begin{aligned}
&239392316861133898555 1149, 2381790971040, 3757634198511\\
&933080427640663987438 7938, 2381790971040, 4492765251707.
\end{aligned}
$$

Find the DSA private key $x$.

(b) Compute the DSA private key $x$ by solving the discrete logarithm problem (1) with $\rho$-method by Pollard.

3. The integer 2 is a primitive root modulo $p = 2602163$. Compute $x \mod p-1$ such that

$$2^x \equiv 1535637 \mod p$$

with the Index Calculus Algorithm, take smoothness bound $B = 30$.