1. (a) (2p) Describe how to set up an RSA cryptosystem.
   (b) (1p) What is the RSA problem? (What mathematical problem does one need to solve to break RSA?)
   Notes: e-th roots mod n intended, but factoring acceptable.
   (c) (2p) What are some of the best known methods for solving the RSA problem? What are their running times? Are they polynomial/subexponential/exponential?
   (d) (3p) Does Pollard's $p-1$ method have any bearing on the parameters one picks to set up an RSA cryptosystem, if each of the primes involved has around 1000 digits?
   More lead: do you think there are choices of primes p and q, each with around 1000 digits, such that one could solve the RSA-problem for the modulus pq on a laptop using Pollard's $p-1$ method?
   Notes: they do not need to be rigorous here, in justifying that there are primes p for which $p-1$ has only small prime factors. There are plenty of primes p with 1000 digits that are, for example, 11-smooth, and for which $N = pq$ can be factorised by Pollard $p-1$ very quickly.

2. (a) (2p) State the ElGamal problem (the problem one needs to solve to decrypt ElGamal-encrypted messages) and state the Diffie-Hellman problem.
   Notes: if they cannot state ElGamal, we can ask about DLP instead, reducing 1p.
   (b) (1p) Is the ElGamal problem easier to solve than the Diffie-Hellman problem? Is it harder? In what sense? (No proof is necessary.)
   (c) (3p) What are the algorithms involved in encryption and decryption in the ElGamal cryptosystem? What are their complexity? Are they polynomial/subexponential/exponential?
   (d) (2p) What is the fastest algorithm (that we know) that breaks the ElGamal cryptosystem? What is its complexity? Is it polynomial/subexponential/exponential?

3. For (a)–(d), the answers should be written down in the chat.
   (a) (1p) How many steps are needed to compute $a^b \pmod{n}$, where $1 < a < n-1$ and b is an integer?
   (b) (1p) How many steps are needed to compute $\gcd(a, b)$ for two integers a, b?
   (c) (1p) What is the average running time for the quadratic sieve to factorise a composite number N?
   (d) (1p) What is the running time of the best known algorithm for solving Elliptic Curve DLPs over $\mathbb{F}_p$?
   (e) (1p) Suppose an algorithm takes an input with k bits and requires $\mathcal{O}\left(e^{(\log k)^2}\right)$ steps to complete. Classify whether the algorithm runs in polynomial/subexponential/exponential time.
   (f) (3p) Classify whether the running times for the named algorithms above are polynomial/subexponential/exponential. How might one prove it?

4. (a) (1p) What kind of problem does the index calculus method solve?
   (b) (1p) What is a B-smooth number?
   (c) (4p) Give an overview of the index calculus method.
   (d) (2p) Let $g, h \in \mathbb{F}_p^*$ for some p. Suppose you know that $g^9 = 12$ and $g^7 = 6$, and further that $h \cdot g^{-10} = 9$. Can you use this to find an integer x such that $g^x = h$?

5. (a) (1p) Suppose you have computed that $3 \cdot 2533 = 100^2 - 49^2$. How can you use this to find some non-trivial factors of 2533?
   (b) (1p) If we have found distinct integers a, b such that $a^2 \equiv b^2 \pmod{N}$, will this definitely allow us to find a non-trivial factor of N?
   (c) (6p) In the course we saw a three-step factorisation procedure, with the steps *Relation building*, *Elimination* and *GCD Computation*. Give a brief description of each of these steps.

6. (a) (1p) The formulas used for adding points on an elliptic curve E work modulo any prime p and define a group structure on the points on this curve. However, modulo a composite number they do not always work. What can go wrong in this case?
   (b) (3p) Suppose the addition of two points on an elliptic curve modulo some composite number N fails. How can this be used to factor N? Give a basic description of Lenstra's factorisation algorithm.
   (c) (2p) Let N be a (large) composite number whose factorisation is unknown. To apply Lenstra's factorisation algorithm, one need to choose an elliptic curve (modulo N) together with a point on this curve modulo N. Why is it in general hard to find points on an elliptic curve modulo N?
   (d) (2p) How can one avoid the above problem and find an elliptic curve modulo N together with one point on this curve (so that one can use Lenstra's factorisation algorithm)?