**Permitted items:**

- The following items are permitted
  - Books (phyisical or pdf), printouts, digital documents on the computer or online including code, handwritten notes
  - Your homeworks and the corrections you received
  - Blank paper for taking notes (no upload of pictures)
  - Pens, pencils, etc
  - Calculators
  - You may run comuter algebra systems as well as your own code on the computer and in online calculators
  - You may use spell-checking tools and pepare text in other editors.
- You may **not** communicate with any other person regarding the exercises by any means during the exam. As an exeption you may contact Tanja Lange if you encounter any problems.
- Looking up existing webpages is permitted; posting the questions or answers counts as communication and is not permitted.
- You may visit the bathroom during the exam time and you may have food and drink on your desk.

**Instructions for answering questions:**
All answers should be entered into the answer fields in Ans; do not write on paper and upload photos of your answers.

The exam has numerical questions, i.e. questions you answer with a single number, and open questions, i.e. questions where you get a text field and can type arbitrary text. For the latter type of questions, make sure to justify your answers in detail and to give clear arguments. Document all steps, in particular of algorithms. It is not sufficient to state the correct result without explanation.

You may copy instructions and outputs from your computer algebra system into the answers but need to explain what they do and why you invoke them.If an exercise requires usage of a particular algorithm, other approaches will not be accepted even if they give the correct result.

**Video upload:**
After this first part finishes you should record a video of you explaining your solution. Choose 3 exercise parts which are not numerical questions and aim for 5 min of recording (no longer than 10 min). Show your student ID and state your name at the beginning of the video.
Please use https://surfdrive.surf.nl/files/index.php/s/LTo2ZYvDq33BTcv
for uploading your video. Name the file as
ID_{student ID}_[Last name].[file format]
filling in your TU/e student ID, your last name, and the file format (mp4, webm) instead of the brackets.
If your connection is too weak, store the video on your computer and compute the SHA-256 checksum of it and mail that to Tanja Lange at t.lange@tue.nl.

**Support:**
If you want to indicate that any unwanted disturbances occurred that might be registered as an irregularity, or if your exam does not go as expected due to technical problems that indered your exam (for example power or Internet failure in the region), you can report this within 24 hours to the Examination Committee via the Webform Online Exam at https://educationguide.tue.nl/studying

/corona/webform-online-exams/.

## Edwards and Montgomery

This exercise is about arithmetic on elliptic curves in Edwards and Montogomery form.
The point $P = (x, y)$ is on the twisted Edwards curve $E : ax^2 + y^2 = a + dx^2y^2$ modulo $p = 2^{31} - 1$ for $a = p - 1$ and
$x =$990692357
$y =$1951500418
$d =$342994443
Compute $2P$ on the Edwards curve and compute the birationally equivalent Montgomery curve $M : Bv^2 = u^3 + Au^2 + u$. Compute the images of $P$ and $2P$ on $M$ and double the image of $P$ on $M$ to see that it matches the image of $2P$.
Make sure to use positive integers in $[0, p - 1]$ to represent the results.
The first exercise parts are numerical exercises and ask for intermediate results from this computation; answer each with a number (you cannot enter text into the field). The last part is an open question where you should enter text and computation details

2.0p **1a** Answer this exercise with the $x$-coordinate of $2P$.

Answer

2.0p **1b** Answer this exercise with the $y$-coordinate of $2P$.

Answer

1.0p **1c** Answer this exercise with $A$.

Answer

1.0p **1d** Answer this exercise with $B$.

Answer

1.0p **1e** Answer this exercise with the $u$-coordinate of the image of $P$.

Answer

1.0p **1f** Answer this exercise with the $v$-coordinate of the image of $P$.

Answer

1.0p **1g** Answer this exercise with the $u$-coordinate of the image of $2P$.

Answer

1.0p **1h** Answer this exercise with the $v$-coordinate of the image of $2P$.

Answer

1.0p **1i** Answer this exercise with the slope $\lambda$ computed in doubling the image of $P$.

Answer

3.0p **1j** Let $Q = (u, v)$ be the image of $P$ on $M$.

Verify that $Q$ is on $M$.

Show the computation of $2Q$ on the Montgomery curve. Document the formulas you use and the results you get.
You should check that the result matches what you computed above

### Factorization

This exercise is about factoring integers. The integer $n$ is a product of three primes.
$n =$15168278781661

5.0p **2a** Let $n =$15168278781661
Use Pollard's rho method for factorization and Floyd's cycle-finding method with starting point

$s_0$ =9593556593250

and iteration function

$s_{i+1} = s_i^2 + d$ for

$d$ =23

to factor $n$,

Accumulate the product of 5 differences into $S$ before doing the gcd computation.

State your computation including copying the code you used.

Compute the other factor of $n$ as well.

1.0p **2b** This is a continuation of the previous exercise. Answer this question with the small factor you obtained from gcd$(S, n)$.

Answer

1.0p **2c** This is a continuation of the previous exercise.

Let $c$ be the factor found using Pollard's rho method and let $n' = n/c$.

Use the $p - 1$ method to factor the unfactored part of $n$ with base $a = 3139020$ and exponent $s$ the lcm of $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16\}$. Make sure to compute the value for $s$ and to compute the result $b$ of the exponentiation modulo $n'$.

For this part filll in the value for $s$.

Answer

1.0p **2d** This is a continuation of the previous exercise. For this answer fill in the value of $b$ (the result after exponentiation, but before subtracting 1).

Make sure to compute $b$ moduo $n'$, not modulo $n$.

Answer

1.0p **2e** This is a continuation of the previous exercise. Fill in the factor you obtained from the gcd in the $p - 1$ method.

Answer

1.0p **2f** This is a continuation of the previous exercise. Fill in the last factor of $n$.

Answer

13.0p **2g** This is a continuation of the previous exercise.

Explain why this was a useful approach to factoring $n$, i.e., explain why you would first use Pollard's rho method and then the $p - 1$ method to factor an integer $n$ and what size integer you could have hoped to reach with 5 steps of Pollard's rho method (a rough estimate suffices).

Let $c$ be the factor found using Pollard's rho method and let $n' = n/c$.

Explain why the $p - 1$ method was successful in finding the remaining factors of $n'$

Consider whether the exponent $s$ would have worked for any base $a$ for these factors and if not, give conditions for which $a$ it does work and how restrictve these are.

Call $p$ the factor you found from gcd$(b - 1, n')$ and $q$ the other factor of $n'$.

- For what fraction of bases $a$ does $p$ divide the gcd?
- For what fraction of bases $a$ does $q$ divide the gcd?
- For what fraction of bases $a$ does the algorithm succeed in factoring $n$?

**Hint:** To give a proper argument you will need to compute the factorizations of $p - 1$ and $q - 1$.

For the factorizations and other computations in this exercise you can use a computer algebra system (Sage, Pari-GP, ...). You do not need to run Pollard's rho method or such for obtaining factorizations. Make sure to state what computations you made,what the answers were, and how they help in solving this question.

## Elliptic-curve discrete logarithm

This exercise is about the elliptic-curve discrete-logarithm problem (ECDLP).

For this exercise we will be considering an elliptic curve given in Weierstrass form $M: y^2 = x^3 + Ax^2 + x$ with $A = 1068$ over the finite field $\mathbb{F}_p$ for $p = 424243$.
There are around $p + 1$ points on the curve but for some reason Alice has decided to work in a cyclic subgroup of order $n = 2^5 \cdot 5$

A generator for the group is $P = [321997, 412581]$. You are given $Q = [219150, 317043]$, another point on this curve which is a multiple of $P$. The task through this exercise is to compute the discrete logarithm of $Q$ with base $P$, i.e., compute $a$ with $Q = aP$.

14.0p **3a** [Scroll up to see the definitions of $M, P, Q$ etc. if you navigated here without seeing them.]

The following is - up to notation - a more detailed instruction of the Pohlig-Hellman computation for prime 2.

Compute $a \equiv a_{2,0} + a_{2,1}2 + a_{2,2}2^2 + a_{2,3}2^3 + a_{2,4}2^4 \bmod 2^5$ by first determining images of the base $P$ and target $Q$ in the subgroup of order $2$ that allow to compute $a_{2,0}$, and then updating the target to another element of in the subgroup of order $2$ to compute $a_{2,1}$ using the same table of multiples of $P$ as in the first step. Continue the same for $a_{2,2}, a_{2,3}$, and $a_{2,4}$.

Explain your steps.This includes providing any program code you used and state all intermediate points, at least those that you used to decide on the $a_{2,i}$ and the updates to $Q$ in the Pohlig-Hellman algorithm.

Verify your answer in the subgroup of order 32.

4.0p **3b** [Scroll up to see the definitions of $M, P, Q$ etc. if you navigated here without seeing them.]

Compute $a \bmod 5$.

3.0p **3c** [Scroll up to see the definitions of $M, P, Q$ etc. if you navigated here without seeing them.]

Combine the results from the previous two exercise parts to compute $a$ with $Q = aP$. Verify your answer.

## Coppersmith's method

University T has a data breach but notices it while data is being exfiltrated. They manage to stop the transfer but some bits have already gotten out.

An investigation after the incident shows that the attacker was targeting private RSA keys which were stored including the primes $p$ and $q$. The top 83 bits of $p$ were leaked and $p$ has only 123 bits.

University T seems unconcerned apart from some public-relations issues regarding the breach. Sadly they do not understand Coppersmith's method and thus do not realize that leaking more than 2/3 of $p$'s bits is as bad as leaking all of $p$.
That's where you come in. You need to show them how to reconstruct all of $p$ from the leaked information and $n$ and also that the same approach works for any of the partially known keys.

8.0p **4a** The information that you get for demonstrating the attack has public key $(n, e)$ with $e = 2^{16} + 1$ and

$n =$
$7102479121236520758442445209151010498795171722381203209451387498687826 44$

The top 83 bits of $p$ are

$p_t =$7975367974709495237422842361682067456

Compute the missing part of $p$ and verify that the resulting $p$ is prime and divides $n$. Compute the other factor of $n$.

**Hint:** In case things go wrong you might have defined x before as a value. Make sure to include
R.<x>=PolynomialRIng(ZZ)
before using x.

1.0p **4b** Aswer this question with the integer that is the bottom 40 bits of $p$, i.e. $p \bmod 2^{40}$. This is the value you obtain as root from Coppersmith's method.

Answer

9.0p **4c** Show that this method always works, i.e., explain in your own words and several math formulas that an attacker knowing the top 83 bits of the 123-bit prime $p$ and the full
value of $n = p \cdot q$ can recover all of $p$.

9.0p **4d** Show how you can adapt the method so that it also works if the bottom 2/3 bits of $p$ become known instead of the top 2/3. For this you need to change the polynomial compared to the setup under b) to ensure that the determinant does not get too large.
With that adjusted polynomial, show why the method succeeds and how to recover $p$ from the information.

## RSA-CRT signatures

This exercise is about RSA signatures and signing using the CRT method.

7.0p **5a** We have seen how to use the Chinese Remainder Theorem to speed up decryption by working with shorter operands and with smaller exponents.

The RSA signature system signs a message $m$ by first computing its hash $h(m))$ for some specified hash function that maps to $[0, n-1]$ and then computing $s \equiv (h(m))^d \bmod n$, where $(n, d)$ is the private key. The signature is $(m, s)$. Verification works by checking that $s^e \equiv h(m) \bmod n$.

Describe in your own words how to speed up signing using CRT, i.e., describe how you can compute $(h(m))^d \bmod n$ more efficiently given $(n, d, p, q, u)$, where $u \equiv p^{-1} \bmod q$.

Point out in which parts you gain efficiency compared to the above direct computation.

8.0p **5b** Dietrich the distracted is using the CRT method for signing. You find a signature $(m, s)$ from him and notice that $s^e \not\equiv h(m) \bmod n$.
You suspect that he lost concentration part way through the process and messed up the computation modulo the second prime. Show how you can compute his private key from this $(m, s)$.
For concreteness, here are the values of $n, e, h(m)$ and $s$:

$n =$
8802480753545133742961614830369231016786934739808948821413906980918012855

$e =$65537

$h(m) =$
843823104557966155504074813938865858062238201809062819132256491547287756

$s =$
76784234115719918382076304288567994613577773275266632795128611352189784