18.783 Elliptic Curves **Spring 2021**

Problem Set #11

Description: These problems are related to material covered in Lectures 17–21.

Instructions: Solve any combination of Problems that sum to 100 points. Problem 1 part (d) uses a result from Problem 3 part (f) of Problem Set 10 — e-mail me if you need this result. Your solutions are to be written up in latex and submitted as a pdf-file.

Collaboration is permitted/encouraged, but you must identify your collaborators or your group, as well any references you consulted that are not listed in the syllabus or lecture notes. If there are none write "**Sources consulted: none**" at the top of your solutions. Note that each student is expected to write their own solutions; it is fine to discuss the problems with others, but your writing must be your own.

The first person to spot each non-trivial typo/error in the problem sets or lecture notes will receive 1-5 points of extra credit.

In cases where your solution involves code, please either include your code in your write up, or (better) the name of a notebook in your 18.783 CoCalc project containing you code (use a separate notebook for each problem).

Problem 1. Mapping the CM torsor (49 points)

Let $\mathcal{O}$ be an imaginary quadratic order of discriminant D, and let $p > 3$ be a prime that splits completely in the ring class field of $\mathcal{O}$, equivalently, a prime of the form $4p = t^2 - v^2 D$. As explained Lecture 17, the set

$$\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p) = \{j(E/\mathbb{F}_p) : \mathrm{End}(E) \simeq \mathcal{O}\}$$

is a $\mathrm{cl}(\mathcal{O})$-torsor. This means that for any $j_1, j_2 \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$, there is a unique $\alpha \in \mathrm{cl}(\mathcal{O})$ for which $\alpha j_1 = j_2$. This has many implications, two of which we explore in this problem.

First and foremost, the $\mathrm{cl}(\mathcal{O})$-action can be used to enumerate the set $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$, all we need is a starting point $j_0 \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$. In this problem we will "cheat" and use the Hilbert class polynomial $H_D(X)$ to do this (in Problem 2 we will find a starting point ourselves). The polynomial $H_D(X)$ splits completely in $\mathbb{F}_p[X]$, and its roots are precisely the elements of $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$. We could enumerate $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ by factoring $H_D(X)$ completely, but that would not let us "map the torsor". We want to construct an explicit bijection from $\mathrm{cl}(\mathcal{O})$ to $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ that is compatible with the group action.

Let us start with a simple example, $D = -1091$. The class number $h(D) = 17$ is prime, so $\mathrm{cl}(D)$ is cyclic and every non-trivial element is a generator. For our generator, let α be the class of the prime form $(3, 1, 91)$, which acts on $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ via cyclic isogenies of degree 3: each $j \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ is 3-isogenous[1] to the j-invariant αj. This means that $\Phi_3(j, \alpha j) = 0$ for all $j \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$, where $\Phi_3(X, Y) = 0$ is the modular equation for $X_0(3)$.

To enumerate $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ as $j_0, j_1, j_2, \ldots$, with $j_k = \alpha^k j_0$, we start by identifying j_1 is a root of the univariate polynomial $\Phi_3(j_0, Y)$. Now $\left(\frac{D}{3}\right) = 1$ in this case, so by part (d) of problem 3 on Problem Set 10 there are two ideals of norm 3 in $\mathrm{cl}(D)$, both of which act via 3-isogenies; the other one corresponds to the form $(3, -1, 91)$, the inverse of α in

[1] When we say that j_1 and j_2 are 3-isogenous, we are referring to isomorphism classes of elliptic curves over $\overline{\mathbb{F}}_p$. There are 3-isogenous curves $E_1/\mathbb{F}_p$ and $E_2/\mathbb{F}_p$ with $j_1 = j(E_1)$ and $j_2(E_2)$, but one must be careful to choose the correct twists.

$\mathrm{cl}(\mathcal{O})$. Thus there are at least two roots of $\Phi_3(j_0, Y)$ in $\mathbb{F}_p$, but provided that we pick the prime p so that 3 does not divide v, there will be only two $\mathbb{F}_p$-rational roots.

There are methods to determine which of of these two roots "really" corresponds to the action of α, but for now we disregard the distinction between α and α^{-1}; this ultimately depends on how we embed $\mathbb{Q}(\sqrt{-1091})$ into $\mathbb{C}$ in any case. Let us arbitrarily designate one of the $\mathbb{F}_p$-rational roots of $\Phi_3(j_0, Y)$ as j_1. To determine j_2, we now consider the $\mathbb{F}_p$-rational roots of $\Phi_3(j_1, Y)$. Again there are exactly two, but we already know one of them: j_0 must be a root, since $\Phi_3(X, Y) = \Phi_3(Y, X)$. So we can unambiguously identify j_2 as the other $\mathbb{F}_p$-rational root of $\Phi_3(j_1, Y)$, equivalently, the unique $\mathbb{F}_p$-rational root of $\Phi_3(j_1, Y)/(Y - j_0)$.

(a) Let $D = -1091$, and let t be the least odd integer greater than $1000N$ for which $p = (t^2 - D)/4$ is prime, where N is the last three digits of you student ID. Use the Sage function `hilbert_class_polynomial` to compute $H_D(X)$, then pick a root j_0 of $H_D(X)$ in $\mathbb{F}_p$ (you will need to coerce H_D into the polynomial ring $\mathbb{F}_p[X]$ to do this). Using the function `isogeny_nbrs` implemented in this Sage notebook, enumerate the set $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ as $j_0, j_1, j_2, \ldots$ by walking a cycle of 3-isogenies starting from j_0, as described above, so that $j_k = \alpha^k j_0$ (assuming that your arbitrary choice of j_1 was in fact $j_1 = \alpha j_0$). You should find that the length of this cycle is 17, since α has order 17 in $\mathrm{cl}(D)$. Finally, verify that the you have actually enumerated all the roots of $H_D(X)$.

(b) Let D, p, and j_0 be as in part (a), and let $\beta \in \mathrm{cl}(D)$ be the class of the prime form $(7, 1, 39)$. Compute $k = \log_\alpha \beta$. Enumerate $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ again as $j'_0, j'_1, j'_2, \ldots$, starting from the same $j'_0 = j_0$ but this time use the action of β, by walking a cycle of 7-isogenies. Rather than choosing j'_1 arbitrarily, choose j'_1 in a way that is consistent with the assumption $j_1 = \alpha j_0$ in part (a): i.e., choose j'_1 so that $j'_1 = \beta j_0 = \alpha^k j_0 = j_k$. Then verify that for all $m = 1, 2, 3, \ldots, 16$ we have $j'_m = \beta^m j_0 = \alpha^{km} j_0 = j_{km}$, where the subscript km is reduced modulo $|\alpha| = 17$.

You should find the results of parts (a) and (b) remarkable (astonishing even). A priori, there is no reason to think that there should be a relationship between a cycle of 3-isogenies and a cycle of 7-isogenies.

The fact that we can use the modular polynomials Φ_ℓ to enumerate the roots of H_D is extremely useful. It allows us to enumerate the roots of polynomials with degrees in the millions, simply by finding roots of polynomials of very small degree (typically one can use Φ_ℓ with $\ell < 20$). We can also use the CM torsor to find zeros of Φ_ℓ, even when ℓ is ridiculously large.

(c) Let ℓ be the least prime greater than $10^{100}N$ for which $\left(\frac{D}{\ell}\right) = 1$, where N is the last three digits of your student ID. Determine the $\mathbb{F}_p$-rational roots of $\Phi_\ell(j_0, Y)$.

For reference, the total size of the polynomial $\Phi_\ell \in \mathbb{Z}[X, Y]$ is roughly $6\ell^3 \log \ell$ bits, which is more than 10^{300} bits in the problem you just solved. Even reduced modulo p, it would take more than 10^{200} bits to write down the coefficients of this polynomial (for comparison, there are fewer than 10^{100} atoms in the observable universe). This example might seem fanciful, but an isogeny of degree 10^{100} is well within the range of cryptographic interest.

Now for a slightly more complicated example, where the class group is not a cyclic group of prime order. Let $D = -5291$. In this case $h(D) = 36$ and the class group $\mathrm{cl}(D)$ is isomorphic to $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/18\mathbb{Z}$. In Problem 3 of Problem Set 10 you computed a polycyclic presentation $\vec{\alpha}$, $r(\vec{\alpha})$, $s(\vec{\alpha})$ for $\mathrm{cl}(D)$, which should involve generators $\vec{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$, of norms 3, 5, and 7. If you did not solve Problem 3 of Problem Set 10, you can email me for a solution.

(d) Let $D = -5291$, and let t be the least odd integer greater than $1000N$ for which $p = (t^2 - D)/4$ is prime, where N is the last three digits of you student ID. Using the polycyclic presentation for $\mathrm{cl}(D)$, enumerate $\mathrm{Ell}_{\mathcal{O}}(D)$ starting from a j-invariant j_0 obtained as a root of H_D. Your enumeration $j_0, j_1, j_2, \ldots, j_{35}$ should have the property that the element $\beta \in \mathrm{cl}(\mathcal{O})$ whose action sends j_0 to j_k satisfies $k = \log_{\vec{\alpha}} \beta$, subject to the assumption that $j_1 = \alpha_1 j_0$.

Here are a few tips on part (d). You will compute $j_0, \ldots, j_{r_1-1}$ using 3-isogenies, but to compute j_{r_1} you will need to compute a 5-isogeny from j_0. When choosing j_{r_1} as a root of $\Phi_5(j_0, Y)$, make this choice consistent with the assumption $j_1 = \alpha_1 j_0$ by using the fact that $s_2 = \log_{\vec{\alpha}} \alpha_2^{r_2}$ (assuming $s_2 \neq 0$, which is true in this case). When you go to compute j_{r_1+1}, you will need to choose a root of $\Phi_3(j_{r_1}, Y)$. Here you can make the choice consistent with the fact that $\mathrm{cl}(\mathcal{O})$ is abelian, so the action of $\alpha_1\alpha_2$ should be the same as the action of $\alpha_2\alpha_1$. Similar comments apply throughout; any time you start a new isogeny cycle, you must make a choice, but you can make all of your choices consistent with your initial choice of j_1.

I don't recommend writing code to make all these choices (it can be done but it is a bit involved), it will be easier and more instructive to work it out by hand, using Sage to enumerate paths of ℓ-isogenies as required (you can use the function `isogeny_path` in this Sage notebook).

Problem 2. Computing Hilbert class polynomials (49 points)

In this problem you will implement an algorithm to compute Hilbert class polynomials using an explicit CRT approach and then use it to construct an elliptic curve over a finite field $\mathbb{F}_q$ via the CM method. The plan is to compute H_D modulo primes p that split completely in the ring class field for the order $\mathcal{O}$ of discriminant D (primes of the form $4p = t^2 - v^2 D$). If we do this for a sufficiently large set of primes S, we can use the Chinese remainder theorem to explicitly determine the coefficients of H_D. For any prime (or prime power) q that satisfies the norm equation $4q = t^2 - v^2 D$ we can then use a root of H_D in $\mathbb{F}_q$ to construct an elliptic curve $E/\mathbb{F}_q$ with $\mathrm{End}(E) = \mathcal{O}$, and in particular, with trace of Frobenius $\pm t$ and $q + 1 \pm t$ rational points; by taking a quadratic twist we can adjust the sign of t.

We will use primes p that are small enough for us to readily find an element $j_0 \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ by trial and error. Note that this will typically not be true of our target prime q, particularly in cryptographic applications; we will use $q = 2^{66} + 9$ which is not of cryptographic size but still large enough to make trial and error an infeasible method for constructing an elliptic curve with $\mathrm{End}(E) = \mathcal{O}$.

Once we know one $j_0 \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$, we can enumerate $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ using a polycyclic presentation for $\mathrm{cl}(\mathcal{O})$, as described in Problem 3 of Problem Set 10. To make our lives simpler, in this problem we will choose $\mathcal{O}$ so that $\mathrm{cl}(\mathcal{O})$ is a cyclic group of prime order

generated by an ideal of small prime norm so that we don't have to compute a polycyclic presentation. This gives us a list of the roots of $H_D \bmod p$, and we can then compute

$$H_D(X) = \prod_{j \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)} (X - j) \bmod p. \tag{1}$$

Once we have computed the coefficients of $H_D \bmod p$ for sufficiently many primes p, we can use the CRT to compute the integer coefficients of $H_D \in \mathbb{Z}[X]$.

But our goal is to construct $E/\mathbb{F}_q$, which means we actually only need $H_D \bmod q$. Rather than computing $H_D \in \mathbb{Z}[X]$ and then reducing modulo q, we will instead apply an explicit form of the CRT that allows us to compute $H_D \bmod q$ directly from the coefficients of $H_D \bmod p$ for sufficiently many small primes p. This saves space (and a little bit of time), because for large $|D|$ the integer coefficients of H_D will typically be much larger than q (possibly by millions of bits).

(a) Write a program that, given a prime $p > 36$ and an integer t finds an elliptic curve $E/\mathbb{F}_p$ satisfying $\#E(\mathbb{F}_p) = p + 1 \pm t$. Do this by generating curves $E/\mathbb{F}_p$ with random coefficients A and B satisfying $4A^3 + 27B^2 \neq 0$. For each curve, pick a random point $P \in E(\mathbb{F}_p)$ (using the `random_point()` method), and test whether $(p+1)P = \pm tP$. If not, discard the curve and continue. Otherwise, compute the order m of P using the generic fast order algorithm provided by the Sage function `sage.groups.generic.order_from_multiple`. If $m > 4\sqrt{p}$ than $\#E(\mathbb{F}_p)$ must be $p + 1 \pm t$, and we have a curve we can use. Otherwise, try again.

Having found a curve $E/\mathbb{F}_p$ whose Frobenius endomorphism π has trace $\pm t$, where $4p = t^2 - v^2 D$, then $\mathbb{Z}[\pi]$ and $\mathrm{End}(E)$ must lie in the maximal order of $K = \mathbb{Q}(\sqrt{D})$. Assuming that D is fundamental, the order $\mathcal{O}$ we are interested in is the maximal order $\mathcal{O}_K$, but unless $\mathbb{Z}[\pi] = \mathcal{O}_K$ it is unlikely that $\mathrm{End}(E) = \mathcal{O}_K$. On the next problem set we will see how to find a curve isogenous to E with endomorphism ring $\mathcal{O}$, but for now we will simply choose primes p that have $v = 1$, in which case $\mathbb{Z}[\pi] = \mathrm{End}(E) = \mathcal{O}_K$.[2] With this provision, **(a)** gives us $j_0 \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$. We can then enumerate $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ as in Problem 1 and apply (1) to compute $H_D(X) \bmod p$.

Once we have computed $H_D \bmod p$ for all the primes in S, we can apply the Chinese remainder theorem to compute $H_D \in \mathbb{Z}[X]$. Let $p_1, \ldots, p_m$ be the primes in S, and let $M = \prod_{p \in S} p$. Let $M_i = M/p_i$, and let $a_i M_i \equiv 1 \bmod p_i$. Let c denote a coefficient of H_D, and let $c_i = c \bmod p_i$ be the corresponding coefficient of $H_D \bmod p_i$.

(b) Prove that

$$c \equiv \sum_{i=1}^{m} c_i a_i M_i \bmod M. \tag{2}$$

Provided that M is big enough, say $M \geq 2B$, where B is an upper bound on $|c|$, this congruence uniquely determines the integer c. Using complex analytic methods, one can establish very accurate bounds B on the absolute values of the coefficients of $H_D(X)$.

[2] With $v = 1$ fixed, we cannot actually prove that any such primes exist, not even under the generalized Riemann hypothesis (GRH), so this does not yield a true algorithm in the sense that we cannot prove it terminates on all inputs. Relaxing the constraint $v = 1$ yields an algorithm that is guaranteed to work, and under GRH, one can prove it is faster than any other method known.

(c) Prove that if $M > 4B$ and r is the nearest integer to $\sum c_i a_i/p_i$, then in fact

$$c = \sum_{i=1}^{m} c_i a_i M_i - rM, \tag{3}$$

and show that if we put $e := \lceil \log_2 m \rceil + 2$ and define $r_i := \lfloor 2^e c_i a_i / p_i \rfloor$, then we have $r = \lfloor 3/4 + 2^{-e} \sum r_i \rfloor$ (in other words, we only need to use $e = O(\log m)$ bits of precision when computing the sum $\sum c_i a_i/p_i$ in order to get the correct value of r).

The fact that (3) is an identity in $\mathbb{Z}$ means that it also holds modulo q; this means that as we compute the coefficients c_i of $H_D \bmod p_i$ it suffices to just accumulate the partial sums of $c_i a_i M_i$ modulo q and the partial sum of the r_i (we do want to compute the sums of the r_i in $\mathbb{Z}$, but they are tiny, typically much smaller than q). As each polynomial $H_D \bmod p_j$ is computed, we will update two running totals for each coefficient c as we go, one for $\sum_i c_i a_i M_i \bmod q$ and one for $\sum_i r_i$.

We are now ready to compute $H_D(X) \bmod q$, where $q = 2^{66} + 9$, and use it to construct an elliptic curve $E/\mathbb{F}_q$. We will use the discriminant $D = -2267$ with class number $h(D) = 11$; the class group is necessarily cyclic, generated by a primeform of norm 7. The coefficients of H_D can be analytically proven to have absolute values bounded by $B = 2^{520}$ via [6, Lemma 8]. As you can check using the `norm_equation` function in this Sage notebook, we have $4q = t^2 - v^2 D$, and for the positive choice of t, the integer $N = q + 1 + t$ is prime. Our goal is to construct $E/\mathbb{F}_q$ with $\#E(\mathbb{F}_q) = N$.

(d) Select a set S of primes $p_1, \ldots, p_m$ of the form $4p = (t^2 - D)$ such that $\prod_{p \in S} p > 4B$. Then compute the $a_i \bmod p_i$ as integers in $[0, p-1]$ and the products $a_i M_i$ modulo q as integers in $[0, q-1]$ for each $1 \le i \le m$. For each prime p_i in S do the following:

1. Find $j_0 \in \mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_{p_i})$ using **(a)**.
2. Enumerate $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_{p_i})$ by walking an 11-cycle of 7-isogenies (as in Problem 1, you can use the `isogeny_nbrs` function in this Sage notebook to do this).
3. Compute $H_D \bmod p_i$ via (1).
4. Update the sums $\sum_i c_i a_i M_i \bmod q$ and $\sum_i r_i$ for each coefficient of $H_D \bmod p_i$.

When all the primes $p_i \in S$ have been processed, for each coefficient c of $H_D \bmod q$, compute r and then c by applying (3) modulo q via **(c)**.

In your answer, list the primes $p_i \in S$ and give a summary of the computation for the first 3 primes in S, including the j-invariant j_0, the enumeration of $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ (in order), and the polynomial $H_D(X) \bmod p$, as well as the end result $H_D \bmod q$.

Here are A few tips for implementing **(d)**. You will need about 40 primes for the set S, the smallest of which should be 569. When debugging your code, you may find it helpful to use Sage to compute the Hilbert class polynomial H_D and compute its roots in $\mathbb{F}_{p_i}$, so that you know exactly the values of $\mathrm{Ell}_{\mathcal{O}}(\mathbb{F}_{p_i})$ that you should be getting. You may find that your algorithm in **(a)** struggles a bit with some of the larger $p_i \in S$, but it should never take more than 10 or 20 seconds or so to find a suitable E, and in most cases it should take less than a second. Once you get it working the entire computation for **(d)** should only take a few minutes. This can be reduced to a few seconds by modifying the algorithm to allow $4p_i = t_i^2 - v_i^2 D$ with v_i not necessarily equal to one and modifying the algorithm in **(a)** to use isogeny-volcano climbing to obtain E with $\mathrm{End}(E) \simeq \mathcal{O}$ in situations where this is not already forced by t_i, but you are not required to do this.

(e) Compute a root $j_0 \in \mathbb{F}_q$ of the polynomial $H_D \bmod q$ you computed in **(d)**, construct an elliptic curve $E/\mathbb{F}_q$ with $j(E) = j_0$ and test whether $\#E(\mathbb{F}_q) = N$ by checking that $NP = 0$ for a random nonzero point $P \in E(\mathbb{F}_q)$. If this is not the case, replace E with its quadratic twist (you can use the `quadratic_twist` method in Sage) and check again. Include a defining equation for your final E in your write-up.

Problem 3. Atkin-Morain ECPP (49 points)

The bottleneck in the Goldwasser-Kilian elliptic curve primality proving algorithm (Algorithm 11.15 in Lecture 11) is counting points on randomly generated elliptic curves in the hope of finding one with a suitable number of points (namely, the product of a large prime and a smooth cofactor). Atkin and Morain proposed an alternative approach that uses the CM method to construct an elliptic curve that is guaranteed to have a suitable number of points [1]. This yields a much faster algorithm, with a heuristic running time of $\widetilde{O}(n^4)$, where n is the size of the input (in bits) and the $\widetilde{O}$ notation ignores polylogarithmic factors of n. While its expected running time is not provably polynomial time, in practice it is substantially faster than even randomized versions of the AKS algorithm that also run in $\widetilde{O}(n^4)$ expected time [2], and is the current method of choice for proving the primality of large primes that are not of a special form. All the primality proving records listed on this top 20 list were proved using this algorithm.

Given a smoothness bound B and probable prime p, the algorithm proceeds as follows:

1. Select a fundamental discriminant $D < -4$ for which $4p = t^2 - v^2 D$ has a solution (t, v) such that $m = p + 1 \pm t$ can be factored as cq, where $c > 1$ is B-smooth and $q > (p^{1/4} + 1)^2$ is a probable prime.[3]

2. Find a root j of $H_D \bmod p$ and use it to construct an elliptic curve $E/\mathbb{F}_p$ in Weierstrass form $y^2 = x^3 + ax + b$, where $a = 3j(1728 - j)$ and $b = 2j(1728 - j)^2$. If unable to find a root of $H_D \bmod p$ within, say, twice the expected amount of time, perform a Miller-Rabin test on p. If it fails then report that p is not prime and otherwise repeat this step.

3. Generate a random $Q \in E(\mathbb{F}_p)$ with $P = cQ \neq 0$ and verify that $qP = 0$. If not, replace E with a quadratic twist $\tilde{E}\colon y^2 = x^3 + d^2Ax + d^3B$, for some nonresidue d, and repeat this step. If the verification $qP = 0$ fails for E and its twist, or if anything else goes wrong (e.g., a square-root computation or inversion fails), report that p is not prime.

4. Output the certificate (p, A, B, x, y, q), where $P = (x, y)$.

As with the Goldwasser-Kilian algorithm, if q is larger than a bound $T \approx (\log p)^4$ one then proceeds to construct a primality certificate for q using the same algorithm, producing a chain of primality certificates that terminates with a prime $q \leq T$ whose primality is verified by trial division (see Lecture 12 for details).

For a fixed fundamental discriminant $D < 0$, we know from the Chebotarev Density Theorem that the proportion of primes p that split completely in the ring class field L for the order of discriminant D is $1/\mathrm{Gal}(L/\mathbb{Q}) = 1/(2h(D))$, where $h(D)$ is the class

[3] In practice one also uses $D = -3, -4$ but for simplicity we will ignore these.

number. We also know that $h(D) \sim \sqrt{|D|}$ as $|D| \to \infty$, and that a constant proportion of all integers $D < 0$ are fundamental discriminants.[4]

(a) Assuming the integers $m = p + 1 \pm t$ in step 1 are as likely as random integers to of the form $2q$ with q prime, give a heuristic upper bound on the absolute value of the discriminant D chosen in step 1 of the form $\widetilde{O}(n^e)$ for some $e > 0$, where $n = \log p$.[5]

(b) Using your heuristic estimate in (a), compute upper bounds on the expected running times of each of steps $i = 1, 2, 3$ of the form $\widetilde{O}(n^{e_i})$; you can assume that the time to compute $H_D(X)$ is quasi-linear in $|D|$, and that the time to solve the norm equation is bounded by the expected time to compute a square root of D modulo p using a probabilistic algorithm (as required by Cornacchia's algorithm, see Problem Set 2). Use these bounds to heuristically bound the expected complexity of proving that p is prime (assuming it is), including the cost of recursively proving that q is prime.

You should find that your heuristic complexity bound is substantially better than the $\widetilde{O}(n^7)$ complexity of the Goldwasser-Kilian algorithm that you analyzed in Problem Set 6, but worse than $\widetilde{O}(n^4)$, and that the cost is dominated by step 1.

In order to obtain an $\widetilde{O}(n^4)$ bound we need to exploit an idea due to Jeffrey Shallit. The key idea is to avoid the need to compute square roots of so many D's modulo p by restricting to discriminants of the form $D = -\ell_1\ell_2$, where ℓ_1 and ℓ_2 are primes in the set $S := \{\ell \le \sqrt{M} : \ell \text{ is prime}\}$ with M chosen according to the heuristic bound on $|D|$ you computed in part (a). The strategy is to compute square roots of $\pm\ell$ modulo p for all the primes in S and use these to efficiently construct square roots of $D = -\ell_1\ell_2$ modulo p.

(c) Using the fact that if it is given the square root of D modulo p, Cornacchia's algorithm can solve the norm equation in quasi-linear time using a fast-GCD approach, derive a new heuristic estimate for the expected running time of step 1 that exploits Shallit's idea (include the cost of computing square roots of the primes $\ell \in S$). Use this to obtain a heuristic $\widetilde{O}(n^4)$ bound on the total expected time to prove that p is prime using the Atkin-Morain approach.

(d) Implement the Atkin-Morain ECPP algorithm described above in Sage and use it to construct a primality proof for the least probable prime p greater than $2^{500}N$, where N is the last 4 digits of your student ID, using the smoothness bound $B = 2^{16}$. You are not required to implement Shallit's optimization, as it won't make much of a difference for primes of this size.

You can use the `norm_equation` function in this Sage notebook to solve the norm equations in step 1. In your implementation, create the finite field $\mathbb{F}_p$ in Sage using `GF(p,proof=false)` to prevent Sage from trying to prove that p is prime. Use the `is_pseudoprime` function in Sage to test whether q is a probable prime after using trial-division to remove the B-smooth factor c. You needn't implement the Miller-Rabin test in step 2 (it is very unlikely to be necessary).

In your write-up, do not list all the primality certificates in full. Just give a table that lists the discriminant D, the j-invariant of the elliptic curve E, and the primes q for each certificate, as well as the time spent constructing each certificate.

[4] Any square free $D \equiv 1 \bmod 4$ certainly works, and this set already has density $3/(2\pi^2)$.

[5] Requiring $m = 2q$ might seem overly restrictive, since the algorithm only requires $m = cq$ with $c > 1$ B-smooth, but it makes no difference in the value of e (unless B is unrealistically large).

Problem 4. Surjectivity of Mod-ℓ Galois Representations (49 points)

This problem is a continuation of Problem 2 of Problem Set 6 and Problem 4 of Problem Set 9. You don't need to have solved those problems in order to do this one, but you will want to at least read through them. In particular, you will need the classification theorem proved in Problem 4 of Problem Set 9 (which you can assume).

Let ℓ be an odd prime and let V be a 2-dimensional $\mathbb{F}_\ell$-vector space, with automorphism group $\mathrm{GL}(V)$, as in the previous problem, and let $\varphi\colon \mathrm{GL}(V) \twoheadrightarrow \mathrm{PGL}(V)$ denote the quotient map.

(a) Let s be an element of $\mathrm{GL}(V)$ whose order is not divisible by ℓ, let $u = \mathrm{tr}(s)^2/\det(s)$, and let r be the order of $\varphi(s)$ in $\mathrm{PGL}(V)$. Prove that $u = \zeta_r + \zeta_r^{-1} + 2$, for some primitive rth root of unity $\zeta_r \in \mathbb{F}_{\ell^2}^\times$.

(b) Suppose that we are in case (iii) of the classification theorem, in which G is a subgroup of $\mathrm{GL}(V)$ whose image in $\mathrm{PGL}(V)$ is isomorphic to A_4, S_4, or A_5. Prove that for all elements $s \in G$, $u = \mathrm{tr}(s)^2/\det(s)$ is equal to $4, 0, 1, 2$ or satisfies $u^2 - 3u + 1 = 0$.

Now we are ready to use this classification to deduce some results about surjectivity of the mod-ℓ Galois representation

$$\rho_{E,\ell}\colon \mathrm{Gal}(\mathbb{Q}(E[\ell])/\mathbb{Q}) \to \mathrm{Aut}(E[\ell]) \simeq \mathrm{GL}(V),$$

of an elliptic curve $E/\mathbb{Q}$. As in Problem Set 6, for each prime $p \neq \ell$ of good reduction for E we pick a Frobenius element $\mathrm{Frob}_p \in \mathrm{Gal}(\mathbb{Q}(E[\ell])/\mathbb{Q})$ which is uniquely determined only up to conjugacy. As shown on Problem 2 of Problem Set 6, every element of $\mathbb{F}_\ell^\times$ arises as the determinant of $\rho_{E,\ell}(\mathrm{Frob}_p)$ for some prime p (infinitely many in fact).

(c) Let $G := \mathrm{im}\,\rho_{E,\ell} \subseteq \mathrm{GL}(V)$. Show that the image H of the G in $\mathrm{PGL}(V)$ contains a (normal) subgroup of index 2. Deduce that if $G \neq \mathrm{GL}(V)$ then one of the following is true:

1. G is contained in the normalizer of a Cartan subgroup;
2. G is contained in a Borel subgroup;
3. G is exceptional and $H = S_4$.

It is a longstanding conjecture that for all $\ell > 37$ we have $G = \mathrm{GL}_2(\mathbb{F}_\ell)$ for all elliptic curves $E/\mathbb{Q}$ without CM. This conjecture remains open, but we know that the only possible exceptions occur when G is contained in the normalizer of a non-split Cartan. Given a particular $E/\mathbb{Q}$ without CM and a particular prime ℓ it is not hard to verify that $G = \mathrm{GL}_2(\mathbb{F}_\ell)$, when this is in fact the case.[6]

(d) Let $G := \mathrm{im}\,\rho_{E,\ell} \subseteq \mathrm{GL}(V)$. Determine three types of elements (specified by their trace and determinant) such that if G contains these elements, then $G = \mathrm{GL}_2(\mathbb{F}_\ell)$.

(e) Let E be the elliptic curve

$$y^2 + y = x^3 - x^2,$$

which has good reduction outside 11. By considering the Frobenius elements $\pi_2 = \rho_{\ell,E}(\mathrm{Frob}_2)$ and $\pi_3 = \rho_{\ell,E}(\mathrm{Frob}_3)$, and using your criterion above, show that $\rho_{E,\ell}$ is surjective for all $\ell \geq 13$ satisfying $\left(\frac{11}{\ell}\right) = -1$.

[6] There is also an effective procedure to determine a finite set of ℓ that need to be checked.

Problem 5. The Gross-Zagier formula for singular moduli (98 points)

The j-invariants of elliptic curves $E/\mathbb{C}$ with complex multiplication are sometimes called singular moduli, since such j-invariants are quite special. As we now know, singular moduli are the roots of Hilbert class polynomials $H_D(X)$. A famous result of Gross and Zagier [5] gives a remarkable formula[7] for the prime factorization of the norm of the difference of two singular moduli arising as roots of two distinct distinct Hilbert class polynomials.

Let D_1 and D_2 be two relatively prime fundamental discriminants. To simplify matters, let us assume that $D_1, D_2 < -4$. Define

$$J(D_1, D_2) = \prod_{i=1}^{h_1} \prod_{k=1}^{h_2} (j_{1,i} - j_{2,k}),$$

where $h_1 = h(D_1)$ and $h_2 = h(D_2)$, and $j_{1,i}$ and $j_{2,k}$ range over the roots of the Hilbert class polynomials $H_{D_1}(X)$ and $H_{D_2}(X)$, respectively.

(a) Prove that $J(D_1, D_2)$ is an integer.

Gross and Zagier discovered an explicit formula for the prime factorization of $J(D_1, D_2)$. To state it we first define two auxiliary functions.

Let us call a prime p suitable if $\left(\frac{D_1 D_2}{p}\right) \neq -1$, and call a positive integer n suitable if all its prime factors are suitable. For all suitable primes p, let

$$\epsilon(p) = \begin{cases} \left(\frac{D_1}{p}\right) & \text{if } p \nmid D_1 \\ \left(\frac{D_2}{p}\right) & \text{if } p \nmid D_2. \end{cases}$$

where $\left(\frac{D}{p}\right)$ denotes the Kronecker symbol.

(b) Prove that $\epsilon(p)$ is well-defined for all suitable primes p.

We extend ϵ multiplicatively to suitable integers n. For suitable integers m, let

$$F(m) = \prod_{nn'=m} n^{\epsilon(n')},$$

where the product is over positive integers n and n' whose product is m.

Theorem (Gross–Zagier)**.** *With notation as above, we have*

$$J(D_1, D_2)^2 = \prod_{\substack{x^2 < D_1 D_2 \\ x^2 \equiv D_1 D_2 \bmod 4}} F\left(\frac{D_1 D_2 - x^2}{4}\right).$$

Note that the product on the RHS is taken over all integers x (positive and negative) that satisfy the constraints (so each nonzero value of x^2 occurs twice).

(c) Prove that for every x in the product of the theorem above, $(D_1 D_2 - x^2)/4$ is a suitable integer (so the formula is well-defined).

[7] This is not the Gross–Zagier formula, it is their second most famous formula. The Gross–Zagier formula concerns the heights of Heegner points and is related to the Birch and Swinnerton–Dyer conjecture.

It is not immediately obvious that the product on the right is actually an integer; in general $F(m)$ need not be. But in fact every $F(m)$ appearing in the product is a (possibly trivial) prime power.

(d) Let m be a positive integer of the form $(D_1D_2 - x^2)/4$. Prove that $F(m) = 1$ unless m can be written in the form:

$$m = p^{2a+1}p_1^{2a_1}\cdots p_r^{2a_r}q_1^{b_1}\cdots q_s^{b_s},$$

where $\epsilon(p) = \epsilon(p_1) = \cdots = \epsilon(p_r) = -1$ and $\epsilon(q_1) = \cdots = \epsilon(q_s) = 1$. Prove that in this case we have

$$F(m) = p^{(a+1)(b_1+1)\cdots(b_s+1)},$$

and thus if p divides $F(m)$ then p is the only prime dividing m with an odd exponent and $\epsilon(p) = -1$. (Hint: see exercises 13.15 and 13.16 in [3]).

(e) Prove that every prime p dividing $J(D_1, D_2)$ satisfies the following:

(i) $\left(\frac{D_1}{p}\right) \neq 1$ and $\left(\frac{D_2}{p}\right) \neq 1$;

(ii) p divides an integer of the form $(D_1D_2 - x^2)/4$;

(iii) $p \leq D_1D_2/4$.

(f) Implement an algorithm to compute the prime factorization of $|J(D_1, D_2)|$, using the Gross-Zagier theorem and parts (d) and (e) above. Then use your algorithm to compute the prime factorization of $|J(D_1, D_2)|$ for three pairs of distinct discriminants that have class number greater than 4. Note that you can compute the class number of D in Sage by creating the number field $\mathbb{Q}(\sqrt{D})$ using `K.<a>=NumberField(x**2-D)` and then calling `K.class_number()`.

(g) For each of the three pairs of discriminants D_1 and D_2 you selected in part (f):

(1) Construct a set S of primes p_i that split completely in the Hilbert class fields of both D_1 and D_2 such that $\prod p_i > 10^6 \cdot |J(D_1, D_2)|$. The `norm_equation` function in this Sage notebook may be helpful.

(2) For each prime $p_i \in S$, compute $J(D_1, D_2) \bmod p_i$ directly from its definition by using Sage to find the roots of $H_{D_1}(X)$ and $H_{D_2}(X)$ modulo p_i and computing the product of all the pairwise differences (in Sage, use the function `hilbert_class_polynomial` to compute $H_{D_1}, H_{D_2} \in \mathbb{Z}[X]$ then use the method `.change_ring(GF(p)).roots()` to find their roots in $\mathbb{F}_p$.

(3) Use the Chinese remainder theorem to compute $J(D_1, D_2) \in \mathbb{Z}$, as explained in Problem 2 above (be sure to get the sign right). Verify that your results agree with your computations in part (f).

Problem 6. Survey (2 points)

Complete the following survey by rating each problem you attempted on a scale of 1 to 10 according to how interesting you found it (1 = "mind-numbing," 10 = "mind-blowing"), and how difficult you found it (1 = "trivial," 10 = "brutal"). Also estimate the amount of time you spent on each problem to the nearest half hour.

	Interest	Difficulty	Time Spent
Problem 1			
Problem 2			
Problem 3			
Problem 4			
Problem 5			

Please rate each of the following lectures that you attended, according to the quality of the material (1="useless", 10="fascinating"), the quality of the presentation (1="epic fail", 10="perfection"), the pace (1="way too slow", 10="way too fast", 5="just right") and the novelty of the material (1="old hat", 10="all new").

Date	Lecture Topic	Material	Presentation	Pace	Novelty
5/5	Ring class fields, the CM method				
5/10	Isogeny volcanoes				

Please feel free to record any additional comments you have on the problem sets or lectures, in particular, ways in which they might be improved.

References

[1] A.O.L. Atkin and F. Morain, *Elliptic curves and primality proving*, Mathematics of Computation **61** (1993), 29–68.

[2] D.J. Bernstein, *Proving primality in essentially quartic random time*, Mathematics of Computation **76** (2007), 398–403.

[3] David A. Cox, *Primes of the form $x^2 + ny^2$: Fermat, class field theory, and complex multiplication*, second edition, Wiley, 2013.

[4] F. Morain, *Implementing the asymptotically fast version of the elliptic curve primality proving algorithm*, Mathematics of Computation **76** (2007), 493–505.

[5] B. Gross and D. Zagier, *On singular moduli*, J. Reine Angew. Math. **355** (1984), 191–220.

[6] A.V. Sutherland, *Computing Hilbert class polynomials with the Chinese remainder theorem.*, Math. Comp. **80** (2011), 501–538.

18.783 / 18.7831 Elliptic Curves

Spring 2021