18.783 Elliptic Curves **Spring 2015**
Problem Set #5 **Due: 03/13/2015**

Description

These problems are related to the material covered in Lectures 9-10. As usual, the first person to spot each non-trivial typo/error will receive one point of extra credit.

Instructions: Solve **two** of Problems 1-3 and then do Problem 4, which is a survey. Please be sure to include your name on your solutions and give your submission a filename of the form `LastNamePSet5.pdf`.

Problem 1. The image of Galois (50 points)

Let $E/\mathbb{Q}$ be an elliptic curve, let ℓ be a prime, and let $K = \mathbb{Q}(E[\ell])$ be the Galois extension of $\mathbb{Q}$ obtained by adjoining the coordinates of all the points in the ℓ-torsion subgroup $E[\ell]$ to $\mathbb{Q}$. The Galois group $\mathrm{Gal}(K/\mathbb{Q})$ acts linearly on the vector space

$$E[\ell] \simeq \mathbb{Z}/\ell\mathbb{Z} \oplus \mathbb{Z}/\ell\mathbb{Z} \simeq \mathbb{F}_\ell^2,$$

thus there is a group homomorphism

$$\rho_E : \mathrm{Gal}(K/\mathbb{Q}) \to \mathrm{GL}_2(\mathbb{F}_\ell)$$

that maps each field automorphism $\sigma \in \mathrm{Gal}(K/\mathbb{Q})$ to an element of the general linear group $\mathrm{GL}_2(\mathbb{F}_\ell)$, which we may view as an invertible 2×2 matrix with coefficients in $\mathbb{F}_\ell$ (after choosing a basis for $E[\ell]$).

As you may recall, a homomorphism from a group G to a group of linear transformations is called a (linear) *representation* of G. The map ρ_E is a representation of the group $\mathrm{Gal}(K/\mathbb{Q})$, known as the *mod-$\ell$ Galois representation* attached to E.[1]

For each prime $p \neq \ell$ where E has good reduction there is a *Frobenius element* Frob_p of $\mathrm{Gal}(K/\mathbb{Q})$, which reduces to the Frobenius map $x \mapsto x^p$ modulo a prime of K lying above p. Let E_p denote the reduction of E modulo such a prime p. The Frobenius element is mapped by ρ_E to an element of $\mathrm{GL}_2(\mathbb{F}_\ell)$ corresponding to π_ℓ, the restriction of the Frobenius endomorphism of E_p to the ℓ-torsion subgroup $E_p[\ell]$. The *Frobenius element* Frob_p is only determined up to conjugacy (and is usually identified with its conjugacy class), since it depends on a choice of basis, but we can unambiguously determine the characteristic polynomial of $\rho_E(\mathrm{Frob}_p) = \pi_\ell$. In particular, the trace of $\rho_E(\mathrm{Frob}_p)$ is the trace of Frobenius $t = p + 1 - \#E_p(\mathbb{F}_p)$ modulo ℓ, and the determinant of $\rho_E(\mathrm{Frob}_p)$ is simply $p \bmod \ell$ (note that $p \neq \ell$).

The Chebotarev density theorem tells us that for any conjugacy class C of $\mathrm{Gal}(K/\mathbb{Q})$, the proportion of primes p for which Frob_p lies in C is exactly the ratio $\#C/\#\mathrm{Gal}(K/\mathbb{Q})$. Asymptotically, we can think of each prime p as being assigned a uniformly random Frobenius element $\mathrm{Frob}_p \in \mathrm{Gal}(K/\mathbb{Q})$ which is mapped by ρ_E to a uniformly random element of the image of ρ_E in $\mathrm{GL}_2(\mathbb{F}_\ell)$. For a typical elliptic curve $E/\mathbb{Q}$, the representation ρ_E is surjective and its image is all of $\mathrm{GL}_2(\mathbb{F}_\ell)$, but this is not always the case. Number theorists (and others) are very interested in understanding these exceptional

[1] One can replace K with any algebraic extension (including an algebraic closure of $\mathbb{Q}$.

cases. The image of ρ_E has a direct impact on the statistical behavior of $E_p[\ell]$ as p varies. For instance, the proportion of primes p for which $E_p[\ell] = E_p(\mathbb{F}_p)[\ell]$ is precisely $1/\#\operatorname{im}\rho_E$, since this occurs if and only if $\rho_E(\mathrm{Frob}_p) = \pi_\ell$ is the identity.

The purpose of this exercise is for you to attempt to determine the image of ρ_E for various elliptic curves $E/\mathbb{Q}$ by analyzing the statistics of π_ℓ as $p = \ell$ varies over primes of good reduction, by comparing these statistics to the corresponding statistics for various candidate subgroups of $\mathrm{GL}_2(\mathbb{F}_\ell)$. Not every subgroup of $\mathrm{GL}_2(\mathbb{F}_\ell)$ can arise as the image of ρ_E, since, for example, $\operatorname{im}\rho_E$ must contain matrices with ever possible nonzero determinant (as p varies, $\det\pi_\ell$ will eventually hit every element of $\mathbb{F}_\ell^*$).

For $\ell = 3$ there are, up to conjugacy, 8 candidate subgroups G of $\mathrm{GL}_2(\mathbb{F}_\ell)$ for the image of ρ_E. These are listed in Table 1, and can also be found in the Sage worksheet 18.783 Problem Set 5 Problem 2.sagews.

group	order	description	generators
C_2	2	cyclic	$\begin{pmatrix}2&0\\0&1\end{pmatrix}$
D_2	4	dihedral	$\begin{pmatrix}2&0\\0&1\end{pmatrix}, \begin{pmatrix}2&0\\0&2\end{pmatrix}$
$D_3 = S_3$	6	dihedral	$\begin{pmatrix}2&1\\2&0\end{pmatrix}, \begin{pmatrix}1&0\\1&2\end{pmatrix}$
C_8	8	cyclic	$\begin{pmatrix}1&1\\1&0\end{pmatrix}$
D_4	8	dihedral	$\begin{pmatrix}2&0\\0&1\end{pmatrix}, \begin{pmatrix}0&2\\1&0\end{pmatrix}$
D_6	12	dihedral	$\begin{pmatrix}1&2\\1&0\end{pmatrix}, \begin{pmatrix}0&1\\1&0\end{pmatrix}$
Q_{16}	16	semi-dihedral	$\begin{pmatrix}1&1\\2&1\end{pmatrix}, \begin{pmatrix}0&1\\1&0\end{pmatrix}$
$\mathrm{GL}_2(\mathbb{F}_3)$	48	general linear	$\begin{pmatrix}2&0\\0&1\end{pmatrix}, \begin{pmatrix}2&1\\2&0\end{pmatrix}$

Table 1. Candidates for the image of ρ_E in $\mathrm{GL}_2(\mathbb{F}_3)$.

(a) The determinant $\det A$, trace $\operatorname{tr} A$, and the multiplicative order $|A|$ of a matrix in $\mathrm{GL}_2(\mathbb{F}_\ell)$ are all invariant under conjugation. Show that the pair $(\det A, \operatorname{tr} A)$ does not determine the conjugacy class of A in $\mathrm{GL}_2(\mathbb{F}_3)$, but then prove that the triple $(\det A, \operatorname{tr} A, |A|)$ does determine the conjugacy class of A in $\mathrm{GL}_2(\mathbb{F}_3)$.

Thus we can get more information about π_ℓ if, in addition to computing its trace, we also compute its multiplicative order in the ring $\mathrm{End}(E_p[\ell])$.

(b) Devise and prove a criterion for computing the order of π_2 in $\mathrm{GL}_2(\mathbb{F}_2)$ based on the number of roots the cubic $f(x)$ has in $\mathbb{F}_p$, where $y^2 = f(x)$ is the Weierstrass equation for E.

(c) Modify the function `trace_mod` that was used in our implementation of Schoof's algorithm in Lecture 9 (which can be found in the Sage worksheet 18.783 Lecture 9-Schoof's algorithm.sagews) so that it also computes the order of π_ℓ and returns both the trace t_ℓ and the order $|\pi_\ell|$ of π_ℓ.

Important: The order of π_ℓ must be computed modulo the full division polynomial ψ_ℓ, not modulo one of its factors. So compute $|\pi_\ell|$ before computing q_ℓ, which is the first place where a division-by-zero error could occur, causing h to be replaced by a proper factor. Also, be sure to compute $|\pi_\ell|$ only the first time through the loop when you know that $h = \psi_\ell$, don't accidentally recompute it if the loop repeats.

Now address the first part of (a) in a different way: pick an elliptic curve $E/\mathbb{Q}$ and find two primes p and p' for which $\pi_3 \in \mathrm{End}(E_p[3])$ and $\pi_3' \in \mathrm{End}(E_{p'}[3])$ have the same characteristic polynomial but different orders in $\mathrm{GL}_2(\mathbb{F}_3)$.

(d) Write a program that, given an elliptic curve E, a prime ℓ, and an upper bound N, enumerates the primes $p \le N$ distinct from ℓ and for which E has good reduction, and for each E_p, computes the triple $(\det \pi_\ell, \operatorname{tr} \pi_\ell, |\pi_\ell|)$. In Sage you can use `prime_range(N+1)` to enumerate primes $p \le N$. Keep a count of how often each distinct triple occurs (use a dictionary, as in the `group stats` function in the Sage worksheet 18.783 Problem Set 5 Problem 2.sagews). Then normalize the counts by dividing by the number of primes p used, yielding a ratio for each triple.

For $\ell = 3$, use your program to provisionally determine the image of ρ_E for each of the ten elliptic curves below, by comparing the statistics computed by your program with the corresponding statistics for each of the 8 candidate subgroups of $\mathrm{GL}_2(\mathbb{F}_3)$. With N around 5000 or 10000 you should be able to easily distinguish among the possibilities. The curves below are also listed in the Sage worksheet 18.783 Problem Set 5 Problem 2.sagews.

$y^2 = x^3 + x$
$y^2 = x^3 + 432$
$y^2 = x^3 + 21x + 26$
$y^2 = x^3 - 3915x + 113670$
$y^2 = x^3 + 5805x - 285714$
$y^2 = x^3 + 1$
$y^2 = x^3 + x + 1$
$y^2 = x^3 - 112x + 784$
$y^2 = x^3 + 4752x + 127872$
$y^2 = x^3 + 652509x - 621544482$

(e) Note that if a given triple $(\det \pi_\ell, \operatorname{tr} \pi_\ell, |pi_\ell|)$ occurs for some E_p but does not occur in a candidate subgroup $G \subset \mathrm{GL}(\mathbb{F}_\ell)$, you can immediately rule out G as a possibility for the image of ρ_E. Analyze the 8 candidate subgroups in Table 1 to find a pair of triples that arise in $\mathrm{GL}_2(\mathbb{F}_3)$ but do not both arise in any of its proper subgroups. If for a given curve $E/\mathbb{Q}$ you can find both of these triples for some E_{p_1} and E_{p_2}, then you have unconditionally *proved* that ρ_E is surjective for $\ell = 3$.

Use this to devise an algorithm that attempts to prove ρ_E is surjective for $\ell = 3$. Your algorithm should return `true` as soon as it can determine $\operatorname{im} \rho_E = \mathrm{GL}_2(\mathbb{F}_3)$ (this should happen quite quickly, if it is true). If this fails to happen after computing triples for E_p for every prime up to, say, 10000, then your algorithm should give up and return `false`. You can think of this as a Monte Carlo algorithm with one-sided error: the "randomness" comes from the assumption that each π_ℓ is uniformly and independently distributed over the image of ρ_E as p varies. If your program returns `true`, then ρ_E is definitely surjective; if it returns `false` it is almost certainly not surjective, but there is a small probability of error.

Using `ZZ.random_element(-100,100)`, generate random elliptic curves $E/\mathbb{Q}$ of the form $y^2 = x^3 + Ax + B$, with A and B uniformly distributed over the interval $[-100, 100]$. Excluding cases where $AB(A^3 + 27B^2) = 0$, use your program to test whether the mod-3 Galois representation ρ_E is surjective or not. List five curves for which your program returns `false`, and provisionally identify the image of ρ_E in each such case as in part 3 above (you may need to test a few thousand curves to achieve this).

Problem 2. Schoof's algorithm (50 points)

In this problem you will analyze the complexity of Schoof's algorithm, as described in Lecture 9 (Algorithms 9.1 and 9.3) and implemented in the Sage worksheet 18.783 Lecture 9- Schoof's algorithm.sagews. In your complexity bounds, use $\mathsf{M}(m)$ to denote the complexity of multiplying two m-bit integers. You may wish to recall that the complexity of multiplying polynomials in $\mathbb{F}_p[x]$ of degree d is $O(\mathsf{M}(d \log p))$, provided that $\log d = O(\log p)$ (in Schoof's algorithm, $\ell = O(\log p)$, so this certainly applies). Under the same assumption, the complexity of inverting a polynomial of degree $O(d)$ modulo a polynomial of degree d is $O(\mathsf{M}(d \log p) \log d)$.

(a) Analyze the time complexity of computing t_ℓ as described in Algorithm 9.3 of the lecture notes and implemented in the `trace_mod` function in the worksheet. Give separate bounds for each of the four non-trivial steps in Algorithm 9.3 as well as overall bounds for the entire algorithm. Express your bounds in terms of ℓ and $n = \log p$, using $\mathsf{M}(m)$ to denote the cost of multiplying two m-bit integers.

(b) Analyze the total time complexity of Schoof's algorithm, as described in Algorithm 9.1 of the lectures notes and implemented in the `Schoof` function of the Sage Worksheet 18.783 Lecture 9- Schoof's algorithm.sagews, as a function of $n = \log p$. Give your answer in three forms, first using $\mathsf{M}(m)$ to express the cost of multiplying m-bit integers, then after plugging in the naïve bound $\mathsf{M}(m) = O(m^2)$ or the Schönhage-Strassen bound for FFT-based multiplication $\mathsf{M}(m) = O(m \log m \log\log m)$.

(c) In your answer to part (a), you should have found that the time complexity bound for one particular step is strictly worse than any of the other steps of Algorithm 9.3. Explain how to modify Algorithm 9.3 so that this step no longer strictly dominates the asymptotic running time.

(d) Revise your time and space complexity estimates in part (b) to reflect part (c).

(e) Analyze the space complexity of Schoof's algorithm as a function of n, both before and after your optimization in part (c).

Problem 3. A Las Vegas algorithm to compute $E(\mathbb{F}_p)$. (50 points)

Let $E/\mathbb{F}_p$ be an elliptic curve over a finite field $\mathbb{F}_p$ of prime order p. In this problem you will use the extended discrete logarithm to design (but need not implement) a Las Vegas algorithm to determine the structure of $E(\mathbb{F}_p)$ as a sum of two cyclic groups

$$E(\mathbb{F}_p) \simeq \mathbb{Z}/N_1\mathbb{Z} \oplus \mathbb{Z}/N_2\mathbb{Z},$$

with $N_1 | N_2$. We will assume that the group order N has already been computed, either by Schoof's algorithm or by the Las Vegas algorithm from Problem Set 3.

Our strategy is to determine the structure of the ℓ-Sylow subgroups of $E(\mathbb{F}_p)$ for each prime ℓ dividing N. Recall that an ℓ-Sylow subgroup is a maximal ℓ-group (a group in which the order of every element is a power of ℓ), and in an abelian group, there is a just one ℓ-Sylow subgroup and it contains every element whose order is a power of ℓ. If ℓ divides N but ℓ^2 does not, then the ℓ-Sylow subgroup is obviously isomorphic to $\mathbb{Z}/\ell\mathbb{Z}$, so we only need to consider primes whose square divides N. Furthermore, even if ℓ^2 does divide N, unless ℓ divides $p-1$, the ℓ-Sylow subgroup will still be cyclic:

(a) Prove that if the ℓ-Sylow subgroup of $E(\mathbb{F}_p)$ is not cyclic then $p \equiv 1 \bmod \ell$.

This yields the following high-level algorithm to compute N_1 and N_2, given N.

1. Compute the prime factorization N.
2. Set $N_1 = 1$ and $N_2 = 1$, and for each maximal prime power ℓ^e dividing N:
   (a) If $e = 1$ or ℓ does not divide $p - 1$, then set $N_2 = \ell^e N_2$ and continue.
   (b) Otherwise, compute the structure $\mathbb{Z}/\ell^{e_1}\mathbb{Z} \oplus \mathbb{Z}/\ell^{e_2}\mathbb{Z}$ of the ℓ-Sylow subgroup of $E(\mathbb{F}_p)$ as described below, with $e_1 \le e_2$, and set $N_1 = \ell^{e_1} N_2$ and $N_2 = \ell^{e_2} N_2$.
3. Output N_1 and N_2

All we need now is an algorithm to compute the ℓ-Sylow subgroup G_ℓ of $E(\mathbb{F}_p)$, given the orders ℓ^e and N of G_ℓ and $E(\mathbb{F}_p)$, respectively. Our strategy is to first pick two random points $P_1, P_2 \in G_\ell$, by generating random points in $E(\mathbb{F}_p)$ and multiplying them by N/ℓ^e. We hope that these points generate G_ℓ. Next, we reduce them to what we hope is a *basis* for G_ℓ, that is, points Q_1 and Q_2 such that $G_\ell \simeq \langle Q_1 \rangle \oplus \langle Q_2 \rangle$. We then have $G_\ell \simeq \mathbb{Z}/\ell^{e_1}\mathbb{Z} \oplus \mathbb{Z}/\ell^{e_2}\mathbb{Z}$ where $\ell^{e_1} = |Q_1|$, $\ell^{e_2} = |Q_2|$. Note that we can quickly compute the order of any element of G_ℓ, since it must be a power of ℓ. Provided that we know the points Q_1 and Q_2 are *independent* (meaning that $\langle Q_1, Q_2 \rangle \simeq \langle Q_1 \rangle \oplus \langle Q_2 \rangle$), in order to verify that we actually have computed a basis for G_ℓ and not some proper subgroup, we just need to check that $e_1 + e_2 = e$. If this does not hold, we try again with two new random points P_1 and P_2; eventually we must succeed.

Your job is to flesh out this strategy and analyze the resulting algorithm. We first recall the definition of the extended discrete logarithm given in Lecture 9.

Definition. For elements α and β of a finite group G, the *extended discrete logarithm* of β with respect to α, denoted $\mathrm{DL}^*(\alpha, \beta)$, is the pair of positive integers (x, y) with $\alpha^x = \beta^y$, where y is minimal subject to $\beta^y \in \langle \alpha \rangle$, and $x = \log_\alpha \beta^y$; or in additive notation, $x\alpha = y\beta$ with y minimal subject to $y\beta \in \langle \alpha \rangle$.

(b) Prove each of the following statements for a finite abelian ℓ-group G containing elements α and β.

(i) If G has ℓ-rank at most 2 and α and β are random elements uniformly distributed over the elements of G, then the probability that $G = \langle \alpha, \beta \rangle$ is at least $3/8$.

(ii) If $(x, y) = \mathrm{DL}^*(\alpha, \beta)$ then y is a power of ℓ.

(iii) For $(x, y) = \mathrm{DL}^*(\alpha, \beta)$ the following are equivalent:
- $x = |\alpha|$ and $y = |\beta|$;
- $\langle \alpha, \beta \rangle$ has order $|\alpha| \cdot |\beta|$.
- α and β are independent;

(iv) If $|\alpha| \ge |\beta|$ and $(x, y) = \mathrm{DL}^*(\alpha, \beta)$ then $y | x$ and $\gamma = \beta - (x/y)\alpha$ and α are independent.

The key fact is (iv), which tells us that we should order the P_i so that $|P_1| \le |P_2|$ and then let $Q_1 = P_1 - (x/y)P_2$ and $Q_2 = P_2$, where $(x, y) = \mathrm{DL}^*(P_2, P_1)$. If we then compute $\ell^{e_1} = |Q_1|$ and $\ell^{e_2} = |Q_2|$, it follows from (iii) that $G_\ell = \langle Q_1, Q_2 \rangle$ if and only if $e_1 + e_2 = e$. Fact (i) tells that we expect this to occur within less than 3 iterations, on average. By (ii), we can compute $(x, y) = \mathrm{DL}^*(P_2, P_1)$, by attempting to compute $x = \log_{P_2} \ell^i P_1$ for $i = 0, 1, 2, \ldots$ until we succeed, at which point we have $y = \ell^i$.[2]

To compute $\log_{P_2} \ell^i P_1$, we use the prime-power case of the Pohlig-Hellman algorithm described in Lecture 10 to reduce the problem to a discrete logarithm computation in a group of prime order ℓ for which we use the baby-steps giant-steps method.

(c) Prove that in a cyclic group of prime-power order $N = \ell^e$ the complexity of the Pohlig-Hellman algorithm is

$$O(e \log \ell \log e + e\sqrt{\ell})$$

group operations. Use this to bound the bit-complexity of computing $\log_{P_2} \ell^i P_1$ in the ℓ-Sylow subgroup of $E(\mathbb{F}_p)$ with order ℓ^e.

(d) Write down a high-level description (not a program) of an algorithm to compute the structure of the ℓ-Sylow subgroup G_ℓ of $E(\mathbb{F}_p)$ in the form $\mathbb{Z}/\ell^{e_1}\mathbb{Z} \oplus \mathbb{Z}/\ell^{e_2}\mathbb{Z}$, given $N = \#E(\mathbb{F}_p)$ and $\ell^e = \#G_\ell$, and analyze its expected time complexity as a function of ℓ, $n = \log p$, and e.

(e) Analyze the total expected time complexity of the algorithm to compute the structure of $E(\mathbb{F}_p)$ in the form $\mathbb{Z}/N_1\mathbb{Z} \oplus \mathbb{Z}/N_2\mathbb{Z}$, given $N = \#E(\mathbb{F}_p)$ (hint: first figure out what the worst case is, then analyze that). You can assume that we have a Las Vegas algorithm that factors N in subexponential time, meaning it is faster than N^ϵ for any $\epsilon > 0$.

Problem 4. Survey

Complete the following survey by rating each of the problems you attempted on a scale of 1 to 10 according to how interesting you found the problem (1 = "mind-numbing," 10 = "mind-blowing"), and how difficult you found the problem (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			

Also, 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").

[2] There are much better ways to do this (a binary search, for example), but using them won't improve the worst-case complexity of the overall algorithm.

Date	Lecture Topic	Material	Presentation	Pace	Novelty
3/5	Schoof's Algorithm				
3/10	Discrete Logarithm Problem				

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.

18.783 Elliptic Curves
Spring 2015