**18.783 Elliptic Curves** **Spring 2015**
**Problem Set #4** **Due: 03/06/2015**

---

## Description

These problems are related to the material covered in Lectures 7-8. Only Problem 5 requires any significant coding (those who choose to do Problem 5 will probably also want to do Problem 3, since it is referenced in Problem 5).

I have made every effort to proof-read the problems, but there may well be errors that I have missed. The first person to spot each error will receive 1-5 points of extra credit on their problem set, depending on the severity of the error.

**Instructions**: Solve Problem 1, **one** of Problems 2 and 3, and **one** of Problems 4 and 5. Then complete Problem 6, which is a survey.

## Problem 1 (30 points). A noncommutative endomorphism ring

Let $p = 7$, and consider the finite field $\mathbb{F}_{p^2}$, which we may represent explicitly as

$$\mathbb{F}_{p^2} \simeq \mathbb{F}_p[i]/(i^2+1) = \{a + bi : a, b \in \mathbb{F}_p\}.$$

To create the field $\mathbb{F}_{p^2}$ in Sage using this particular representation, use

```
F7.<x>=PolynomialRing(GF(7))
F49.<i>=GF(49,modulus=x^2+1)
```

Now consider the elliptic curve $E/\mathbb{F}_{p^2}$ defined by

$$y^2 = x^3 + (1+i)x.$$

The group of $\mathbb{F}_{p^2}$-rational points on $E$ is isomorphic to $\mathbb{Z}/6\mathbb{Z} \oplus \mathbb{Z}/6\mathbb{Z}$ and is generated by the affine points

$$P_1 = (i, i), \quad P_2 = (i+2, 2i),$$

which you can construct in Sage using `P1=E(i,i)` and `P2=E(i+2,2*i)`. Let $\pi_E$ denote the Frobenius endomorphism of $E$.

**(a)** Prove that $\pi_E = 7$ holds in $\mathrm{End}(E)$ (hint: this is easy, don't make it hard).

Since $\pi_E$ corresponds to an integer in $\mathrm{End}(E)$, you might be tempted to conclude that $\mathrm{End}(E) \simeq \mathbb{Z}$. But this is far from true.

**(b)** Show that the $p$-power Frobenius map $\pi$ of degree $p = 7$ does not lie in $\mathrm{End}(E)$.

**(c)** Prove that nevertheless $\mathrm{End}(E)$ does contain an endomorphism $\alpha$ of degree 7 by exhibiting an explicit rational map $\alpha\colon E \to E$ that satisfies $\alpha^2 = -7$.

**(d)** Now find another endomorphism $\beta$ that satisfies $\beta^2 = -1$ (give $\beta$ explicitly).

**(e)** Prove that $\alpha$ and $\beta$ do not commute, but that $\alpha\beta = -\beta\alpha$ holds.

## Problem 2 (30 points). Isogeny invariants

Let $E_1$ and $E_2$ be isogenous elliptic curves over a finite field $\mathbb{F}_q$ of characteristic $p$, related by the isogeny $\alpha\colon E_1 \to E_2$. Using **only material covered in the lecture notes**, prove the following statements.

**(a)** Prove that $E_1[p] \simeq E_2[p]$ (so $E_1$ is supersingular if and only if $E_2$ is supersingular).

**(b)** Prove that $\alpha(E_1(\mathbb{F}_q)) \subseteq E_2(\mathbb{F}_q)$ but that equality need not hold.

**(c)** Prove that, nevertheless, $\#E_1(\mathbb{F}_q) = \#E_2(\mathbb{F}_q)$ always holds.

**(d)** Prove that $E_1(\mathbb{F}_q)$ is not necessarily isomorphic to $E_2(\mathbb{F}_q)$ (give an explicit example).

For isogenous elliptic curves $E_1$ and $E_2$ over fields $k$ that are not finite, even when $\#E_1(k)$ and $\#E_2(k)$ are finite it is not generally true that $\#E_1(k) = \#E_2(k)$. As an explicit example, let $E_1/\mathbb{Q}$ be the elliptic curve defined by $y^2 = x^3 - 27x + 55350$. The group $E_1(\mathbb{Q})$ is finite and consists of the five points listed below

$$E_1(\mathbb{Q}) = \{(51 : \pm 432 : 1), (-21 : \pm 216 : 1), (0 : 1 : 0)\}.$$

**(e)** Use Vélu's formulas to determine the isogenous curve $E_2/\mathbb{Q}$ that is related $E_1$ via a separable isogeny with kernel $E_1(\mathbb{Q})$ (you only need to determine the equation for $E_2$, you don't need to write down the isogeny). Prove that $\#E_2(\mathbb{Q})$ is finite and that $\#E_1(\mathbb{Q}) \neq \#E_2(\mathbb{Q})$.

## Problem 3 (30 points). Fast order algorithms

Let $\alpha$ be an element of a generic group $G$, written additively. Let $N$ be a positive integer for which $N\alpha = 0$, and let $p_1^{e_1} \cdots p_r^{e_r}$ be the prime factorization of $N$. An algorithm that computes the order of $\alpha$ given $N$ and its prime factorization is known as a *fast order algorithm*. It's fast because the knowledge of $N$ and its factorization allows the algorithm to run in polynomial time (polynomial in $n = \log N$); determining the order of $\alpha$ without being given $N$ provably takes exponential time.

The naïve fast order algorithm given in class is rather inefficient. This is irrelevant in the context of computing the order of a point in $\#E(\mathbb{F}_q)$ with the baby-steps giant-steps method; the complexity is dominated by the time to determine $N$. But this is not the case in every application. In this problem you will analyze two more efficient approaches.

When giving time complexity bounds for generic group algorithms, we simply count group operations, since these are assumed to dominate the computation (so integer arithmetic costs nothing). Space complexity is measured by counting the maximum number of group elements that the algorithm must store simultaneously, but for this problem we will just be concerned with time complexity. You may use the fact that the maximum number of distinct primes dividing an integer $N$ is bounded by $O(n/\log n)$, where $n = \log N$, which follows from the Prime Number Theorem. All your complexity bounds should be specified in terms of $n$.

**(a)** The fast order algorithm given in class begins by initializing $m = N$ and then for each prime $p_i | N$ it repeatedly replaces $m$ by $m/p_i$ so long as $p_i | m$ and $(m/p_i)\alpha = 0$. Analyze the time complexity of this algorithm in the worst case, and give separate asymptotic bounds for inputs of the form $2^k$ and $p_1 \cdots p_k$.

**(b)** Consider an alternative algorithm that first computes $\alpha_i = (N/p_i^{e_i})\alpha$ for $1 \le i \le r$, and then determines the least $d_i \ge 0$ for which $p_i^{d_i}\alpha_i = 0$ by computing the sequence

$$\alpha_i,\ p_i\alpha_i,\ p_i^2\alpha_i,\ \ldots,\ p_i^{d_i}\alpha_i = 0,$$

where each term is obtained from the previous via a scalar multiplication by $p_i$. Show that the order of $\alpha$ is $\prod_i p_i^{d_i}$. Analyze the time complexity of this algorithm in the worst case, and give separate asymptotic bounds for inputs of the form $2^k$ and $p_1 \cdots p_k$.

**(c)** Consider a third algorithm that uses a recursive divide-and-conquer strategy. In the base case $r = 1$, so $N = p^k$ is a prime power and it computes the sequence $\alpha, p\alpha, p^2\alpha, \ldots, p^d\alpha = 0$ as above and returns $p^d$. For $r > 1$ it sets $s = \lfloor r/2 \rfloor$ and puts $N = N_1N_2$ with $N_1 = p_1^{e_1} \cdots p_s^{e_s}$ and $N_2 = p_{s+1}^{e_{s+1}} \cdots p_r^{e_r}$. It then recursively computes $m_1 = |N_1\alpha|$ and $m_2 = |N_2\alpha|$ and outputs $m_1m_2$.

**(i)** Prove that this algorithm is correct (you may use the fact $|\alpha| = \gcd(m, |\alpha|)|m\alpha|$ that you proved in Problem Set 2).

**(ii)** Analyze the time complexity of this algorithm in the worst case, and give separate asymptotic bounds for inputs of the form $2^k$ and $p_1 \cdots p_k$.

## Problem 4 (40 points). The probability of $\ell$-torsion

Let $\ell$ be a prime. In this problem you will determine the probability that a random[1] elliptic curve $E/\mathbb{F}_p$ has an $\mathbb{F}_p$-point of order $\ell$, where $p$ is either a fixed prime much larger than $\ell$, or a prime varying over some large interval. Let $\pi = \pi_E$ be the Frobenius endomorphism of $E$, and let $\pi_\ell \in \mathrm{GL}_2(\mathbb{F}_\ell)$ denote the matrix corresponding to the action of the Frobenius endomorphism of $E$ on the $\ell$-torsion subgroup $E[\ell]$ with respect to some chosen basis (here we have identified $\mathbb{F}_\ell$ with $\mathbb{Z}/\ell\mathbb{Z}$). The matrix $\pi_\ell$ is only defined up to conjugacy, since it depends on the choice of basis, but its trace $\operatorname{tr}\pi_\ell = \operatorname{tr}\pi \bmod \ell$ and $\det\pi_\ell = \deg\pi = p \bmod \ell$ are uniquely determined. We will make the heuristic assumption that $\pi_\ell$ is uniformly distributed over $\mathrm{GL}_2(\mathbb{F}_\ell)$ as $E$ varies over elliptic curves defined over $\mathbb{F}_p$ and $p$ varies over integers in some large interval (one can prove that the distribution of $\pi_\ell$ converges to the uniform distribution on $\mathrm{GL}_2(\mathbb{F}_\ell)$ as $p \to \infty$).

**(a)** Determine the probability that $E(\mathbb{F}_p)[\ell] = E[\ell]$, both for a fixed $p$ (in which case the answer will depend on $p \bmod \ell$), and for $p$ varying over some large interval (assume every possible value of $p \bmod \ell$ occurs equally often).

Use your answer to derive a heuristic estimate for the probability that $E(\mathbb{F}_p)$ is cyclic, for large $p$, by estimating the probability that $E(\mathbb{F}_p)[\ell] \ne E[\ell]$ for all $\ell$, assuming that these probabilities are independent.[2] Use Sage to compute the product of these probabilities for primes $\ell$ bounded by 50, 100, 200, 500, and then given an estimate that you believe is accurate to at least 4 decimal places for all sufficiently large $p$.

Now test your heuristic estimate using the following Sage script

[1] There are several ways to vary the random elliptic curve $E/\mathbb{F}_p$. We will just pick curve coefficients $A$ and $B$ at random and ignore the negligible number of cases where the discriminant is 0.

[2] This assumption is false, but the extent to which it is false becomes negligible as $p \to \infty$.

```
cnt=0
for i in range(0,1000):
    p=random_prime(2^20,2^19); F=GF(p)
    A=F.random_element(); B=F.random_element()
   if EllipticCurve([A,B]).abelian_group().is_cyclic():
        cnt += 1
print cnt/1000.0
```

In the unlikely event that you stumble upon a singular curve, simply rerun the test. Run this script three times (be patient, it may take a few minutes), and compare the results to your estimate.

**(b)** Show that a necessary and sufficient condition for $E(\mathbb{F}_p)[\ell] \neq \{0\}$ is

$$\operatorname{tr} \pi_\ell \equiv \det \pi_\ell + 1 \pmod{\ell}.$$

**(c)** Under our heuristic assumption, to determine the probability that $E(\mathbb{F}_p)$ contains a point of order $\ell$, we just need to count the matrices $\pi_\ell$ in $\mathrm{GL}_2(\mathbb{F}_\ell)$ that satisfy this condition. Your task is to derive a combinatorial formula for this probability as a rational function in $\ell$. Do this by summing over the possible values of $\det \pi_\ell$, so that you can also compute the probability for any fixed value of $p$, which determines $\det \pi_\ell \equiv p \bmod \ell$. For each nonzero value of $d = \det \pi_\ell \in \mathbb{F}_\ell$, you want to count the number of matrices in $\mathrm{GL}_2(\mathbb{F}_\ell)$ that have determinant $d$ and trace $d+1$.

As a warm-up, for $\ell = 3$ use Sage to count the number of matrices $\pi_\ell \in \mathrm{GL}_2(\mathbb{F}_3)$ with trace $d+1$ for $d = 1$ and $d = 2$. You can then compute the probability of $\ell$-torsion for a fixed $p \equiv 1 \bmod 3$ or $p \equiv 2 \bmod 3$, and also the average probability for varying $p$ by averaging over the 2 possible values of $d = \det \pi_\ell \equiv p \bmod \ell$.

You can solve this problem with purely elementary methods, but if you know a little representation theory you may find it helpful to consult the character table for $\mathrm{GL}_2(\mathbb{F}_\ell)$ (be sure to list your sources). Assume initially that $\ell$ is odd, and after obtaining your formula, verify that it also works when $\ell = 2$.

**(d)** For $\ell = 3, 5, 7$ do the following: Pick two random primes $p_1, p_2 \in [2^{29}, 2^{30}]$, with $p_1 \equiv 1 \bmod \ell$ and $p_2 \not\equiv 1 \bmod \ell$, and for each prime generate 1000 random elliptic curves $E/\mathbb{F}_p$. Count how often $\#E(\mathbb{F}_p)$ is divisible by $\ell$, and compare this with the value predicted by the formulas you derived in part (b).

## Problem 5 (40 points). A Las Vegas algorithm to compute $\#E(\mathbb{F}_p)$

Implement a Las Vegas algorithm to compute $\#E(\mathbb{F}_p)$, as described in class. Use Sage's built-in functions for generating random points on an elliptic curve, for adding points on an elliptic curve, and for performing scalar multiplication, but write your own code for performing the baby-steps giant-steps search and the fast order computation. When implementing the search, you will want to use a python dictionary to store the baby steps — python will automatically create a hash table to facilitate fast lookups (alternatively you can do a sort and match yourself, just be sure to avoid a linear search).

In the code below, $\mathcal{H}(p) = [p+1-2\sqrt{p}, p+1+2\sqrt{p}]$ denotes the Hasse interval. The following algorithm to compute $\#E(\mathbb{F}_p)$ was given in class:

**Input**: An elliptic curve $E/\mathbb{F}_p$, where $p > 229$ is prime.
**Output**: The cardinality of $E(\mathbb{F}_p)$.

1. Find a random non-square element $d \in \mathbb{F}_p$ and use it to compute the equation of a quadratic twist $E_1$ of $E_0 = E$ over $\mathbb{F}_p$.
2. Set $N_0 = N_1 = 1$ and $i = 0$ (the index $i$ is used to alternate between $E_0$ and $E_1$).
3. While neither $N_0$ nor $N_1$ has a unique multiple in $\mathcal{H}(p)$:
   (a) Pick a random point $P$ on $E_i$.
   (b) Use a baby-steps giant-steps search to find a multiple $M$ of $|P|$ in $\mathcal{H}(p)$.
   (c) Compute the prime factorization of $M$ using Sage's `factor` function.
   (d) Compute $m = |P|$ using any of the fast order algorithms from Problem 3.
   (e) Set $N_i = \text{lcm}(m, N_i)$ and set $i = 1 - i$.
4. If $N_0$ has a unique multiple $M$ in $\mathcal{H}(p)$ then return $M$, otherwise return $2p+2-M$, where $M$ is the unique multiple of $N_1$ in $\mathcal{H}(p)$.

Let $E$ be the elliptic curve $y^2 = x^3 - 35x - 98$.

**(a)** Using $p = 4657$, run your algorithm on $E/\mathbb{F}_p$ and record the values of $N_i$, $M$, and $m$ that are obtained as the algorithm progresses.

**(b)** For $k = 20, 40, 60, 80$, pick a random prime $p$ in the interval $[2^{k-1}, 2^k]$ (using Sage's `random_prime` function with the `lbound` parameter). Record the time it takes for you program to compute $E/\mathbb{F}_p$ for the elliptic curve $y^2 = x^3 + 314159x + 271828$ for each of these primes and list these timings in a table.

The timings will vary depending on your exact implementation and the machine you are running on, but you should be able to see an $O(p^{1/4})$ growth rate for large $p$; the $k = 20$ and $k = 40$ cases will be too quick too see this, but you should see the times go up by a factor of roughly $2^{20/4} = 32$ as you move from a 40-bit to a 60-bit and then an 80-bit prime. As ball park figures to shoot for, the cases $k = 20, 40$ should both take less than a second, the $k = 60$ case should take a few tens of seconds (more like five seconds if your code is tight), and the $k = 80$ case will take several minutes but should not take more than half an hour (it can be done in well under five minutes). If you are not seeing $O(p^{1/4})$ growth it likely means that you are inadvertently doing a linear search of the baby steps rather than a table lookup; check your code.

Now consider the following alternative implementation of the algorithm above. Rather than having your baby-steps giant-steps search terminate as soon as it finds a multiple $M$ of $|P|$, have it search the *entire* Hasse interval so that it finds *every* multiple $M$ of $|P|$ in $\mathcal{H}(p)$. If it finds only one such $M$, then you can immediately proceed to step 4. If it finds more than one, then you can let $m$ be the difference of the least two $M$'s and eliminate steps 3c and 3d.

**(c)** Prove that this works, implement the modified algorithm, and repeat part (b).

## Problem 6. 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 it (1 = "mind-numbing," 10 = "mind-blowing"), and how difficult you found it (1 = "trivial," 10 = "brutal"). 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 | | | |

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").

| Date | Lecture Topic | Material | Presentation | Pace | Novelty |
|---|---|---|---|---|---|
| 2/26 | Endomorphism rings | | | | |
| 3/3 | Point counting | | | | |

Please 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