Mathematics 3159A Assignment 5 Fall 2020

Instructions

- This assignment is due on December 1, 2020 at 2:00 PM EST. Late submissions will **not** be accepted.
- This assignment consists of two problems. You should choose one for submission.
- Your solution needs to be formatted using the LaTeXtemplate available on OWL.
- All solutions must be written in full sentences.
- You are not allowed to work with others or use any online resources.
- This assignment is worth 5 points.

Problem 1.

Fix a prime p and a positive integer k. Recall that $\mathbb{F}_{p^k}$ is the field with p^k elements, given by polynomials of degree less than k over $\mathbb{F}_p$. Let $\varphi\colon \mathbb{F}_{p^k} \to \mathbb{F}_{p^k}$ be a map given by $\varphi(\mathbf{a}) = \mathbf{a}^p$.

1. Show that $\varphi(\mathbf{a}+\mathbf{b}) = \varphi(\mathbf{a}) + \varphi(\mathbf{b})$ and $\varphi(\mathbf{a}\cdot\mathbf{b}) = \varphi(\mathbf{a})\cdot\varphi(\mathbf{b})$. In other words, φ is a homomorphism.

2. Show that $\varphi(a) = a$ for every $a \in \mathbb{F}_p$, i.e., φ behaves like the identity function on constant polynomials.

3. Let $f(X,Y) \in \mathbb{F}_p[X,Y]$ and suppose that $(\mathbf{a},\mathbf{b}) \in \mathbb{F}_{p^k}^2$ is a root of f, i.e., $f(\mathbf{a},\mathbf{b}) = 0$. Show that $(\varphi(\mathbf{a}),\varphi(\mathbf{b}))$ is also a root of f.

   Deduce that for any elliptic curve E defined over $\mathbb{F}_p$, the map $\overline{\varphi}\colon E(\mathbb{F}_{p^k}) \to E(\mathbb{F}_{p^k})$ given by $\overline{\varphi}(\mathbf{a},\mathbf{b}) = (\varphi(\mathbf{a}),\varphi(\mathbf{b}))$ is well-defined.

4. Let E be an elliptic curve defined over $\mathbb{F}_p$ and $\overline{\varphi}\colon E(\mathbb{F}_{p^k}) \to E(\mathbb{F}_{p^k})$ be the map defined in part 3. Show that

$$\overline{\varphi}(P \oplus Q) = \overline{\varphi}(P) \oplus \overline{\varphi}(Q)$$

   for all $P, Q \in E(\mathbb{F}_{p^k})$. In other words, φ is a group homomorphism from E to itself.

Problem 2.

In this assignment you will implement the key step in the Elliptic Curve Diffie-Hellman Key Exchange.

Mathematics 3159A Assignment 5 Fall 2020

Statement

1. Write a function in Python3 called `solve` that takes as input:

   - a prime p (representing the field $\mathbb{F}_p$);
   - integers A and B (representing the elliptic curve $E : Y^2 = X^3 + AX + B$);
   - integers P_x and P_y (representing a point $P = (P_x, P_y)$ in $E(\mathbb{F}_p)$);
   - and a positive integer n.

   The output should be a pair (Q_x, Q_y), that represents the point $Q = nP$ in the curve $E(\mathbb{F}_p)$. (Make sure that $0 \leq Q_x, Q_y, < p$.) If Q happens to be the point at infinity, the output should be $(-1, -1)$.

2. Download the file `generate_input.py`, and use it to obtain a list of 10 tuples of the form (p, A, B, P_x, P_y, n) by importing the file

   ```
   from generate_input import generate_input
   ```

   and running the function

   ```
   generate_input("[last three digits of your student number]")
   ```

   (Note the quotation marks.)

3. Run your method `solve` on all these inputs.

Your submission must consist of a single PDF file containing:

1. the *Python code* implementing your solution;

2. the 10 *inputs you generated*, and the *output of your program* run on these inputs. One input and one output per line.

Examples

Here are some examples of what your function `solve` should do.

```
>>> solve(13, 2, 2, 2, 1, 2)
(6, 10)
>>> solve(5, 1, 3, 4, 1, 2)
(1, 0)
>>> solve(17, 4, 12, 5, 15, 3)
(-1, -1)
```

Notes

- We encourage you to check your answers using mathematical software, e.g., `Sage` (free software with syntax similar to Python's). Knowing how to use such software will be useful in the future, especially if you are interested in number theory, cryptography, computational algebra, and other areas.

- The integer n will usually be quite big, so make sure that your algorithm terminates when run on the inputs we provide.

- The numbers generated by `generate_input.py` are quite big, so a brute-force solution will not work.

- Make sure you are using a 64bit version of Python3.

- Your code should not make use of any external libraries such as `numpy` or `math`. All the auxiliary functions should be implemented by you, and should be included in your submission. You should only use the most basic arithmetic operations such as `+`, `-`, `*`, `//`, `%`.

- Comments in the code are not mandatory. However in the case of an incorrect solution, the comments can provide grounds for partial credit.