KRY: Wahlfach Kryptologie
Serie 14: Kryptographie auf elliptischen Kurven

Aufgabe 1 (P)
Schreiben Sie für die Klasse `BigInteger` der Package mybiginteger eine Methode

`BigInteger[] elliptMultiply` (`BigInteger y`, `BigInteger factor`, `BigInteger p`, `BigInteger a`, `BigInteger b`),

die falls möglich über dem Körper $K = GF(p)$ für die elliptische Kurve E: $y^2 = x^3 + ax + b$ den Punkt $Q = \texttt{factor} \cdot P = \underbrace{P + P + \ldots + P}_{\texttt{factor} \text{ viele Summanden}}$ berechnet und zurückgibt. Falls $P = (\texttt{this}, y)$ nicht zu E gehört, so soll eine `NumberFormatException` mit der Nachricht `Punkt liegt nicht auf der Kurve!` ausgegeben werden. Der unendlich ferne Punkt O soll in der Form $(p, *)$ entgegengenommen bzw. zurückgegeben werden, wobei $*$ ein beliebiger Wert sein darf (weil er nirgends weiter beachtet wird). Überprüfen Sie den Algorithmus in der Testumgebung "Prakt. 14.1".

Tests

(1) **Eingabe:** $p = 13$, $a = 2$, $b = 7$, $n = 3$, $P = (5, 8)$. **Ausgabe:** $(12, 2)$

(2) **Eingabe:** $p = 13$, $a = 2$, $b = 7$, $n = 4$, $P = (5, 8)$. **Ausgabe:** $(10, 0)$.

(3) **Eingabe:** $p = 13$, $a = 2$, $b = 7$, $n = 8$, $P = (5, 8)$. **Ausgabe:** O.

(4) **Eingabe:**

$p = 691573603372196506117063072108086208490009717980042331161253766952412410690 13$

$a = 124832081098922902417252851099469873364267901343128507865512867630615660222 89$

$b = 423124432334994640298163385046248801026295741834987716394135177258420452257 58$

$n = 456786854356754356786543567868543567865435 67$

$P = (46784,$
$166080230420209717208053211699167444707602942726694578749624097560205248179 99)$

Ausgabe:

$Q = (679861899935729423803368055815454943069341240093383970337825509469079468671 73,$
$402231538311002533428544542293220418797762157862635114800018283730069530800 09)$

Aufgabe 2 (P)
Vervollständigen Sie das Source-File `ElGamalEllipt` des Programms KryptoTrainer, indem Sie die folgenden Methoden ausprogrammieren.

(1) Die Methode `elliptEncrypt`, welche für einen gegebenen Klartext-Punkt die El Gamal Verschlüsselung auf elliptischen Kurven durchführt.

(2) Die Methode `elliptDecrypt`, welche für gegebene B und C den zugehörigen Klartext-Punkt bestimmt.

(3) Vervollständigen Sie die Methode `messageEncrypt`, welche aus einer gegebenen Klartext-Nachricht zuerst einen Klartext-Punkt als Element einer elliptischen Kurve konstruiert, und diesen danach mithilfe der Methode aus Teil (1) verschlüsselt. Die Darstellung der Ausgabe ist also analog zu Teil (1).

(4) Vervollständigen Sie die Methode `messageDecrypt`, welche für gegebene B und C mithilfe vom Programm aus (2) die zugehörige Klartext-Nachricht bestimmt.

Tests (für Testumgebung "Prakt. 14.2")

(1) **Eingabe:** $p = 11$, $a = 3$, $b = 9$, $P = (2, 1)$, $k_A = 7$, $k_B = 3$, $M = (10, 4)$
zugehörige Ausgabe: $B = (10, 7)$, $C = (3, 10)$

(2) **Eingabe:** $p = 10'009$, $a = 3$, $b = 5$, $P = (4, 10'000)$, $k_A = 6$, $k_B = 2$, $m = 34$
zughörige Ausgabe: $B = (9731, 9219)$, $C = (1681, 7275)$ **oder** $C = (8579, 5938)$

(3) **Eingabe:** p, a, b, P analog zu Test (4) aus Aufgabe 1. Dann k_A, k_B, m selbst wählen und kontrollieren, dass Verschlüsseln und Entschlüsseln wieder zu m führt.