КОМП'ЮТЕРНИЙ ПРАКТИКУМ №2
З КУРСУ
ТЕОРЕТИКО-ЧИСЛОВІ АЛГОРИТМИ В КРИПТОЛОГІЇ

Застосування алгоритму дискретного логарифмування

1 Мета роботи

Ознайомлення з алгоритмом дискретного логарифмування Сільвера-Поліга-Геллмана. Практична реалізація цього алгоритму. Пошук переваг, недоліків та особливостей застосування даного алгоритму дискретного логарифмування. Практична оцінка складності роботи алгоритму.

2 Необхідні теоретичні відомості

Звичайний логарифм $\log_a(b)$ — це пошук розв'язку x рівняння $a^x = b$ в полі дійсних (або комплексних) чисел. Розглянемо G — довільну скінченну мультиплікативну абелеву групу, а α і β є елементами цієї групи, тоді, аналогічно до звичайного, *дискретний логарифм* числа β за основою α ($\log_\alpha(\beta)$) — це ціле число x таке, що $\alpha^x = \beta$.

Ефективного методу для обчислення дискретного логарифма в загальному випадку не існує. Проте вони можуть ефективно обчислюватись в деяких особливих випадках. Деякі криптосистеми з відкритим ключем базуються на складності обчислення дискретного логарифма, наприклад, криптосистема Ель-Гамаля.

Задача дискретного логарифмування формулюється так: маючи просте число p, число α — генератор групи Z_p^*, число β — елемент Z_p^*, знайти число x, $0 \le x \le p-2$, таке, що $\alpha^x = \beta \bmod p$.

Найпростіший розв'язок цієї задачі — повний перебір всіх елементів Z_p^*. Складність при такому підході: $O(n)$.

Розглянемо дещо кращий алгоритм розв'язку задачі дискретного логарифмування.

Алгоритм Сільвера-Поліга-Гелмана.
Вхід алгоритму:

- α – генератор групи G;
- $\beta \in G$;

- n – порядок групи G.

Вихід алгоритму: x, $0 \le x \le n-1$, $\alpha^x = \beta$.

Кроки алгоритму:

1. Обчислюємо канонічний розклад числа n: $n = p_1^{l_1} \cdot \ldots \cdot p_m^{l_m}$, $l_i \ge 1$.
2. Для кожного $i = \overline{1,m}$ будуємо таблицю:
$$\{r_{p_i,j} = \alpha^{\frac{n \cdot j}{p_i}},\ j = \overline{0, p_i - 1}\}.$$
3. Для кожного p_i, $i = \overline{1,m}$, $p_i^{l_i} | n$ обчислюємо $x = \log_\alpha \beta \bmod p_i^{l_i}$. Позначимо
$$x = x_0 + x_1 p_i + \ldots + x_{l-1} p_i^{l-1} \bmod p_i^l,$$
тоді x будемо обчислювати за цим співвідношенням за допомогою значень $x_0, x_1, \ldots, x_{l-1}$.
4. Для обчислення x_0 розглянемо ланцюжок співвідношень:
$$\beta = \alpha^x; \tag{1}$$
$$\beta^{\frac{n}{p_i}} = (\alpha^x)^{\frac{n}{p_i}}; \tag{2}$$
$$\beta^{\frac{n}{p_i}} = (\alpha^{x_0 + x_1 p + \ldots + x_{l-1} p^{l-1}})^{\frac{n}{p_i}}; \tag{3}$$
$$\beta^{\frac{n}{p_i}} = \alpha^{\frac{x_0 \cdot n}{p_i}} \cdot \alpha^{x_1 \cdot n} \cdot \ldots \cdot \alpha^{x_{p_i - 1} \cdot p_i^{l-2} \cdot n}. \tag{4}$$
Оскільки $\alpha^n = 1$, то, спрощуючи співвідношення (4), маємо:
$$\beta^{\frac{n}{p_i}} = \alpha^{\frac{x_0 \cdot n}{p_i}}.$$
Ліву частину обчислюємо, значення з правої частини раніше обчислювались в таблиці. Порівнюємо значення зліва зі значеннями в таблиці, визначаємо j. Це і є шукане значення x_0.
5. Аналогічно обчислюємо наступні значення x_i:
$$\beta^{\frac{n}{p_i^2}} = \alpha^{\frac{x_0 \cdot n}{p_i^2} + \frac{x_1 \cdot n}{p_i}}; \tag{5}$$
$$\alpha^{\frac{x_1 \cdot n}{p_i}} = (\beta \cdot \alpha^{-x_0})^{\frac{n}{p_i^2}}. \tag{6}$$
При цьому значення x_0 вже відоме з пункту 4.
Для обчислення x_k:
$$\left(\beta \cdot \alpha^{-x_0 - \ldots - x_{k-1} \cdot p^{k-1}}\right)^{\frac{n}{p_i^{k+1}}} = \alpha^{\frac{x_k \cdot n}{p_i}}.$$
6. Обчисливши всі x_i, обчислюємо значення x для кожного значення $p_i^{l_i}$. Таким чином, отримаємо систему конгруенцій:
$$\begin{cases} x \equiv y_1 \bmod p_1^{l_1}; \\ x \equiv y_2 \bmod p_2^{l_2}; \\ \vdots \\ x \equiv y_m \bmod p_m^{l_m}. \end{cases}$$
7. Застосовуючи китайську теорему про лишки для розв'язку системи конгруенцій, обчислюємо $x \bmod n$, що і є шуканим дискретним логарифмом.

В загальному випадку, складність алгоритму Сільвера-Поліга-Гелмана оцінюється як

$$O\left(\sum_{i=1}^{m} l_i p_i + \log_2 n\right).$$

У випадку, якщо відомий канонічний розклад числа n, алгоритм має складність

$$O\left(\sum_{i=1}^{m} l_i(\log_2 n + p_i^{1-a_i}(1 + \log_2 p_i^{a_i})\right), 0 \leq a_i \leq 1.$$

3 Порядок виконання роботи і методичні вказівки

-1. Ознайомитись з порядком виконання комп'ютерного практикуму та відповідними вимогами до виконання роботи.

0. Уважно прочитати необхідні теоретичні відомості до комп'ютерного практикуму.

1. Написати програму, що реалізовує алгоритм Сільвера-Поліга-Геллмана для груп типу Z_p^*.

2. Встановити Docker (`https://docs.docker.com/get-docker/`). За допомогою команди

   ```
   docker run -it salo1d/nta_cp2_helper
   ```

   запустити допоміжну програму, яка генерує задачу пошуку дискретного логарифма. Вхідним параметром програми є просте число p — модуль кільця лишків. Програма очікує від користувача розв'язку та повідомляє про його правильність чи не правильність. Час створення задачі дискретного логарифма не перевищує 10 хвилин. Програма очікує введення розв'язку 5 хвилин.

3. Застосовувати реалізований алгоритм Сільвера-Поліга-Геллмана до задачі дискретного логарифма, яку формує програма з попереднього пункту, почергово зі збільшенням порядку p. Потрібно пам'ятати, що вхідний параметр p повинен бути простим числом. У випадку, якщо допоміжна програма не справляється зі завданням генерації задачі, або ваша реалізація не справляється з розв'язком задачі за відведений час, зупинити збільшення вхідного параметра p.

   Приклад збільшення параметру p:

   1. $p = 3$
   2. $p = 17$
   3. $p = 157$
   4. ...

4. Оформити звіт до комп'ютерного практикуму.

4 Оформлення звіту та порядок захисту комп'ютерного практикуму

Звіт про виконання комп'ютерного практикуму оформлюється згідно зі стандартними правилами оформлення наукових робіт. Рекомендується виконувати звіти за допомогою системи набору і верстки LaTeX, причому дозволяється використовувати розмір шрифту 12pt та одинарний міжрядковий інтервал.

Звіт обов'язково має містити:

- мету комп'ютерного практикуму;
- постановку задачі;

- хід роботи, опис труднощів, що виникали під час виконання завдання, та шляхи їх подолання;
- результати дослідження, зокрема інформація про всі ітерації процесу збільшення вхідного параметра p для допоміжної програми зі сформованими задачами і їх розв'язками на кожній ітерації;
- результати продуктивності роботи програмної реалізації алгоритму С-П-Г (заміри часу роботи програмної реалізації на кожній ітерації збільшення вхідного параметра p, візуалізацію залежності часу роботи від вхідного параметра p);
- оцінку максимального порядку вхідного параметра p, при якому процес побудови задачі і її розв'язання відбувався за відведений час;
- висновки до роботи (не перефразована мета).

Тексти коду програми дозволяється не включати у звіт. Тексти всіх програм здаються викладачу в електронному вигляді для перевірки на плагіат. До захисту практичної частини комп'ютерного практикуму допускаються студенти, які оформили звіт та надали його для перевірки викладачу. Для зарахування комп'ютерного практикуму студенту необхідно виконати захист практичної частини роботи та тексти програм даного студента повинні пройти перевірку на плагіат.

Оцінювання комп'ютерного практикуму

Можлива кількість рейтингових балів 8

Програмна реалізація 6

Звіт 2

Несвоєчасне виконання роботи .....-1 бал за кожне заняття пропуску

Академічний плагіат до 10% .....0 балів за комп'ютерний практикум, -10 балів до рейтингу

Академічний плагіат більше 10% .....анулювання балів за семестр та недопуск до заліку