

בעיית Equality

לאליס מחרוזת $A = (a_0, \dots, a_{n-1}) \in \{0,1\}^n$ ולבוב מחרוזת $B = (b_0, \dots, b_{n-1}) \in \{0,1\}^n$. אליס ובו ברוצים לבדוק אם המחרוזות שוות, שהמטרה היא לשלוח כמה שפחות ביטים.

בהרצאה הראונו פרוטוקול לבעיה, המבוסס על מספרים ראשוניים. עתה נראה פרוטוקול נוסף לבעיה, מבוסס על פולינומים:

אליס ובו מבחזקים $A = (a_0, \dots, a_{n-1})$ ו- $B = (b_0, \dots, b_{n-1})$ בהתאמה.

- יהא α מספר שייקבע בהמשך.
- אליס בונה את הפולינום $R_A(X) = \sum_{i=0}^{n-1} a_i X^i$.
- אליס מוצאת ראשוני q כך ש- $q > \alpha n$.
- אליס מגרילה מספר t כך ש- $0 \leq t < q$ בהתפלגות אחידה ומחשבת את $R_A(t) \bmod q$.
- אליס שולחת לבוב את q , את t ואת התוצאה של $R_A(t) \bmod q$.
- בוב בונה את הפולינום $R_B(X) = \sum_{i=0}^{n-1} b_i X^i$.
- בוב בודק האם $R_B(t) \bmod q$ שווה לתוצאה של $R_A(t) \bmod q$ שקיבל מאליס.
- אם המספרים שווים הוא מחזיר 'כן', אחרת 'לא'.

ניתוח השגיאה

נשים לב כי אם $A = B$ אזי $R_A(x) = R_B(x)$ ולכן הפרוטוקול יחזיר תמיד תשובה נכונה. לעומת זאת, אם $A \neq B$ אז $R_A(x) \neq R_B(x)$ ונראה שלרוב ה- t ים: $R_A(t) \bmod q \neq R_B(t) \bmod q$.

הערות:

מציאת ראשוני q כך ש- $q > \alpha n$:
הגרל מספר בין αn ל- $(\alpha + 1)n$. בדוק אם הוא ראשוני (קיים אלגוריתם בסיבוכיות $O(\log^3 n)$).
אם ראשוני – סיימו, אחרת הגרל שוב מספר.
צפיפות הראשוני היא $\frac{1}{\log n}$ ולכן סך הכל סיבוכיות הגרלת מספר ראשוני הוא בתוחלת לכל היותר $O(\log^4 n)$. פורמלית: יהי $p(n)$ מספר הראשוניים בקבוצה $\{1, \dots, n\}$, אז $p(n) \in \Theta\left(\frac{n}{\log n}\right)$.

המספרים $R_A(t)$, $R_B(t)$ הם מספרים גדולים ויקרים לחישוב. לא מחשבים את המספר בשום שלב באופן ישיר. אחרי כל פעולת כפל מבצעים $\bmod q$, לכן המספר תמיד מיוצג ע"י $\log q$ ביטים. ז"א מחשבים $R_A(X) = \left(\sum_{i=0}^{n-1} (b_i X^i \bmod q)\right) \bmod q$.

סה"כ זמן ריצה - $O(n)$ בגודל הקלט.

גודל הקלט $\log q$.

פעולות חיבור, כפל וכו' - $O(1)$. חישוב הפולינום $O(n)$.

הגרלת מספר ראשוני $O(\log^4 n)$.

מה המספר הביטים שנשלחים:

$q, t < q, R_A(t) \bmod q$

ובסך הכל $3 \log q$.

תזכורת:

לפולינום מדרגה לכל היותר $n - 1$ יתכנו $n - 1$ שורשים לכל היותר מודולו q .
 לדוגמא, נבחן את הפולינום $p(X) = x + x^2$ מודולו 5.
 $p(0) = 0, p(1) = 2, p(2) = 1, p(3) = 2, p(4) = 0 \pmod 5$
 דרגת הפולינום 2, מספר השורשים 2.

טענה: לאלגוריתם הנ"ל יש טעות חד כיוון של $1/\alpha$ לכל היותר.

מה הכוונה טעות חד צדדית כאן?

אם A ו- B מחוזות שוות, אז האלגוריתם תמיד צודק.
 אם A ו- B שונות אז האלגוריתם טועה בהסתברות $\frac{1}{\alpha}$ לכל היותר.

הוכחה:

אם A ו- B מחוזות שוות, אז $R_A(X) = R_B(X)$ ולכן לכל ראשוני q ומספר $0 \leq t < q$ יתקיים
 $R_A(t) \pmod q = R_B(t) \pmod q$ ולכן יוחזר 'כן' בהסתברות 1.
 אם A ו- B מחוזות שונות, אז נסתכל על הפולינום ההפרש
 $D(X) = R_A(X) - R_B(X) = \sum_{i=0}^{n-1} (a_i - b_i)X^i$
 דרגתו של $D(X)$ הוא קטנה מ- n .
 בהינתן ראשוני $q > \alpha n$ ופולינום $D(X)$ שונה מפולינום האפס (אחרת המחוזות שוות) ההסתברות
 לטעות היא ההסתברות שאליס מגרילה t כל ש- $D(t) \equiv 0 \pmod q$.
 כלומר אליס מגרילה מספר שהיא שורש של פולינום ההפרש מודולו q .
 כמה מספרים כאלה יש? אפשר לספור לפי התזכורת:

$$\Pr[\text{אליס מגרילה שורש}] = \Pr\left[\frac{\text{Alice chooses a root}}{\text{numbers Alice can choose}}\right] < \frac{n}{q}$$

ובסך הכל קיבלנו:

$$\Pr[\text{אליס מגרילה שורש}] = \Pr[\text{האלגוריתם טעה}] \leq \frac{n}{q} < \frac{n}{\alpha n} = \frac{1}{\alpha}$$

כנדרש.

לדוגמא, אם $\alpha = 2$ ההסתברות לטעות היא $\frac{1}{2}$. אם נריץ את הפרוטוקול R פעמים נקבל שגיאה $\frac{1}{2^R}$.

שאלות למחשבה:

איך מורידים את מסר הביטים הנשלחים אך שומרים על אותה רמת טעות?
 האם אשר לשפר את הדיוק של הפרוטוקול בלי לשנות את זמן הריצה?

בעיית התאמת מחרוזות

בסעיפים הבאים נעזר באלגוריתם הנ"ל בכדי להציע אלגוריתם להתאמת המחרוזות.

תזכורת: בבעיית התאמת המחרוזות נתונות שתי מחרוזות:

$$\langle P[1], P[2], \dots, P[n] \rangle, \langle T[1], T[2], \dots, T[m] \rangle$$

נניח כי $m > n$. יש להחזיר קבוצת האינדקסים S כך ש- $j \in S$ אם ורק אם

$$\langle P[1], P[2], \dots, P[n] \rangle = \langle T[j+1], T[j+2], \dots, T[j+n] \rangle$$

נציע את האלגוריתם הבא:

1. $S \leftarrow \emptyset$
2. Find prime q such that $q > 4mn$
3. Randomly generate $0 \leq t < q$
4. Build Polynomial $R(X) = \sum_{i=1}^n P[i+1]X^i$ From $\langle P[1], \dots, P[n] \rangle$
5. Calculate $\sigma = R(t) \bmod q$
6. For all $0 \leq j \leq m - n$
 - a. Build polynomial $Q_j(X) = \sum_{i=0}^{n-1} P[j+i+1]X^i$ from T
 - b. Calculate $\tau_j = Q_j(t) \bmod q$
 - c. If $\tau_j = \sigma$ then $S \leftarrow S \cup \{j\}$
7. Return S

סעיף א

בשלב j באלגוריתם אנחנו בונים את הפולינום $Q_j(X)$.

הוכיחו כי $Q_j(X) = XQ_{j+1}(X) - T[j+n+1]X^n + T[j+1]$.

סעיף ב

הסבירו איך ניתן לחשב ביעילות את τ_j מתוך τ_{j+1} ו- q ו- t^n .

סעיף ג

הציעו מימוש לאלגוריתם להתאמת מחרוזות שמתואר בתרגיל זה שהסיבוכיות שלו היא $O(n+m)$.

סעיף ד

הוכיחו כי האלגוריתם טועה בהסתברות לכל היותר $1/4$.

תשובות

סעיף א

$$Q_j(X) = \sum_{i=0}^{n-1} T[j+i+1]X^i = T[j+1]X^0 + \sum_{i=1}^{n-1} T[j+i+1]X^i$$

$$Q_{j+1}(X) = \sum_{i=0}^{n-1} T[j+i+2]X^i$$

$$XQ_{j+1}(X) = \sum_{i=0}^{n-1} T[j+i+2]X^{i+1} = T[j+n+1]X^n + \sum_{i=0}^{n-2} T[j+i+2]X^{i+1}$$

$$= T[j+n+1]X^n + \sum_{i=0}^{n-1} T[j+i+1]X^i$$

ולכן $Q_j(X) = XQ_{j+1}(X) - T[j+n+1]X^n + T[j+1]$.
 בנייתו זמן הריצה בסעיפים הבאים נניח שהפעולות על מספר המיוצג על ידי כמת לוגריתמים של ביטים לוקחות $O(1)$. זוהי הנחה סבירה כי במחשבים יש חומרה יעודית שמבצעת פעולות אריתמטיות על מילים ולא ברמת הביטים. ניתן לנתח בצורה יותר מדוייקת ולקבל תוספת סיבוכיות של $O(\log^k n)$ עבור k קבוע.

סעיף ב

בהינתן $\tau_{j+1} = Q_{j+1}(t) \bmod q$ נכפיל ב- t , נחסיר את $T[j+n+1](t^n \bmod q)$ ונוסיף את $T[j+1]$. נפעיל $\bmod q$ על התוצאה ונקבל את τ_j . מכיוון שאורך הייצוג של כל המספרים בחישוב הוא $O(\log q)$, זמן החישוב הוא $O(1)$.

סעיף ג

בשלב הראשון נחשב את $t^i \bmod q$ עבור $i = 1, \dots, n$.
 את שלב 6 באלגוריתם נעשה בסדר יורד, כלומר מ- $j = m - n$ עד $j = 1$ ולכן יהיה ניתן להשתמש בסעיף ג'.
 שלב 1-3 באלגוריתם $O(1)$.
 שלב 4 באלגוריתם $O(n)$.
 שלב 5 באלגוריתם $O(n)$.
 שלב 6 באלגוריתם: חישוב $\tau_{m-n} - O(n)$, חישוב $\tau_0, \dots, \tau_{m-n-1} - O(m)$.
 ולכן קיבלנו $O(n+m)$.

סעיף ד

נחשב את המספר ה- t ששיגרמו לאלגוריתם לטעות. האלגוריתם טועה אם הוא בחר t כך שבאחד מ- $m - n + 1$ הבדיקות מתקיים $R(t) \bmod q = Q_j(t) \bmod q$. לכל j ישנם לכל היותר n מספרים t כנ"ל. לכן, בסך הכל, יש לכל היותר $(m - n - 1)n > mn$ מספרים t שיגרמו לאלגוריתם לטעות וההסתברות לשגיאה היא לכל היותר:

$$\frac{mn}{q} < \frac{mn}{4mn} = \frac{1}{4}$$

הצעה נוספת: אפשר להשתמש בחסם האיחוד עבור הסתברויות ולרשום:

$$(m - n) \left(\frac{1}{q} \right) \leq (m - n) \left(\frac{1}{4m} \right) \leq \frac{m}{4m} = \frac{1}{4}$$