הרצאה 8 אלגוריתמים

המשך הוכחה מהרצאה 7:

צעד: הנחת האינדוקצייה היא שיש פתרון אופטימלי כך שלכל $1 \leq j \leq k$, $I_J \in X^* \Leftrightarrow I_j \in X$

נתבונן באינטרוול $k+1$. נחלק לשני מקרים לפי בחירת האלגוריתם ב- I_{k+1}:

1. $I_{k+1} \in X$ אם $I_{k+1} \in X^*$ סיימנו, אחרת $I_{k+1} \notin X^*$ ובהכרח קיים אינטרוול ב-X^*שנחתך עם I_{K+1}.
נבחר את האינטרוול I_n בעל האינדקס המינימלי מ-X^* שנחתך עם I_{k+1}. נשים לב ש-$r \geq k+2$
שכן אלו $r \leq k$ מפני ש-$I_r \in X^*$ נקבל ש-$I_r \in X^*$ (מהנ"א).
זוהי <u>סתירה</u> לכך שהאלגוריתם בחר את I_{k+1}. נשים לב ש-I_n הוא היחיד ב-X^* שנחתך עם I_{k+1}
(אחרת X^* לא אופטימלי).
נסתכל על $X^* \backslash \{I_r\} \cup \{I_{k+1}\}$ ונוכיח שהוא פתרון חוקי, והטענה תנבע מכך ש-$|X^*| = |X^* \backslash \{I_r\} \cup \{I_{k+1}\}|$.
נראה ש-I_{k+1} לא נחתך עם אף אינטרוול ב-$X^* \backslash \{I_r\}$.
   (א) האם I_{k+1} נחתך עם משימום עם אינדקסים $r+1, r+2, \ldots, n$? לא.
   (ב) האם I_{k+1} נחתך עם משימום עם אינדקסים $k+1, k+2, \ldots, r-1$? לא.
   (ג) האם I_{k+1} נחתך עם משימום עם אינדקסים $1, 2, \ldots, k$? לא.
   $\Leftarrow$ $X^* \backslash \{I_r\} \cup \{I_{k+1}\}$ פתרון חוקי.
2. $I_{k+1} \notin X$ לפי הנ"א, X^* לא יכול להכיל את I_{k+1} שכן במקרה כזה ב-X^* היו שני אינטרוולים שנחתכים.

שאלה:

לכלכ אינטרוול I_j נתון רווח $p_j \geq 0$. כיצד ממקסמים את סך הרווחים מהבקשות שמספקים?

קידוד Huffman

בגדול, בהנתן קובץ המורכב מאוסף תווים, רוצים למצוא כיצד "לקודד" אותו כך שאורך הקובץ המקודד
יהיה קצר ככל הניתן. בקידוד כל תו בקובץ יהפוך למחרוזות בינארית (באורך כלשהו).
מילת קוד היא מחרוזת בינארית $w = w_1 w_2 \ldots w_n$ $w_i \in \{0,1\}$
האורך של מילת הקוד w יסומן ע"י $l(w)$.
קוד הוא אוסף של מילות קוד.

דוגמה:

$C = \{\underset{(a_1)}{C_1}, \underset{(a_2)}{C_2}, \underset{(a_3)}{C_3}\}$ כאשר $C_1 = 01,\ C_2 = 0,\ C_3 = 10$

פעולת הקידוד נעשית ע"י החלפת כל תו במילת הקוד המתאימה לו.

כלומר $a_1a_2a_3 \underbrace{\rightarrowtail}_{\text{(code)}} 01010$

כיצד מפענחים קידוד ?

$a_2a_3 \to 010 \leftarrow a_1a_2$, $\underbrace{01}_{c_1}\underbrace{0}_{c_2}$ או $\underbrace{0}_{c_2}\underbrace{10}_{c_3}$?

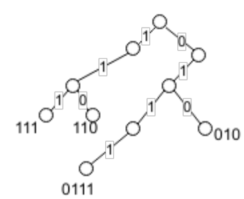
אנו נרצה להגביל את עצמינו לקידוד שיש רק דרך אחת לפענח אותו.
נתמקד בקודים חסרי רישאות, שבהם אין מילת קוד שהיא רישא של מילת קוד אחרת.
עבור קודים חסרי רישאות, הפיענוח פשוט ונעשה ע"י קריאה של הקובץ המקודד.
נתון: n תווים כך שלתו ה-i נתונה תדירות f_i.
מטרה: למצוא קוד שבו התו ה-i מותאם למילת קוד c_i, שיש דרך אחת לפענח אותו
המביאה למינימום $\sum_{i=1}^{n} l(c_i) \cdot f_i$.

טענה (לא להוכחה):

מבין הפתרונות האופטימליים קיים אחד לפחות שהוא קוד חסר רישאיות.

שאלה:

האם קיימת דרך נוחה לייצג קוד חסר רשאיות?
ניתן להציג קוד חסר רישאות ע"י עץ בינארי
(למשל שמאלה זה 1 וימינה 0) כאשר העלים הם מילות הקוד.



טענה 1

קוד אופטימלי מיוצג ע"י עץ מלא (לכל צומת שאינו עלה יש שני ילדים ישירים).

הוכחה בתרגול.

הרעיון:

שני העלים עם התדירות הנמוכה ביותר יהיו עלים אחים עמוקים ביותר בעץ.

האלגוריתם של Huffman (1952)

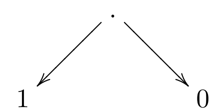
1. נמיין את התווים הנתונים לפי התדירויות: $f_1 \geq f_2 \geq \cdots \geq f_n$

2. נוציא את התווים ה-n וה-$n-1$, ובמקומם נכניס תו "מלאכותי" בעל תדירות $f_{n-1}+f_n$.
נפתור רקורסיבית את הבעיה עבור הקלט המצומצם ונקבל עץ T'.

3. ב-T' נוסיף לעלה שמייצג את התו המלאכותי שני לדים ישירים
כאשר האחד מייצג את התו ה-$n-1$ והשני את התו ה-n.
נסמן ב-T את התו המתקבל.

4. החזרת את T.

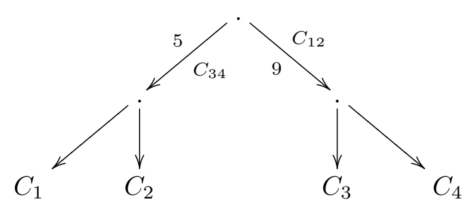
מה תנאי העצירה של האלגוריתם הרקורסיבי? למשל אם $n=2$ מחזירים עץ:

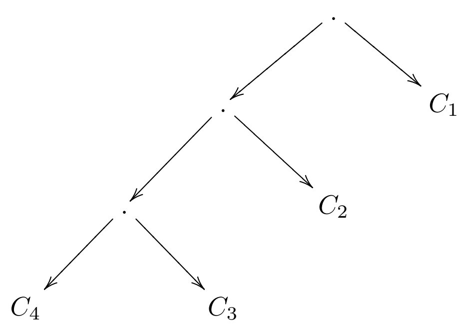


דוגמה:

$$f_1=1, f_2=2, f_3=3, f_4=4, f_5=5$$

$$\{C_1, C_2, C_3, C_4\}$$
$$\{C_1, C_2, C_{34}\}$$
$$\{C_{12}, C_{34}\}$$



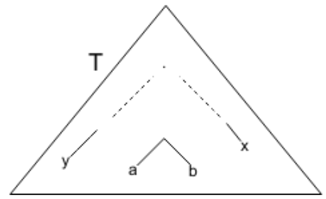


טענה 2

יהיו x ו-y שתי מילות הקוד בעלות התדרים הנמוכים ביותר.
אזי קיים פתרון אופטימלי T שבו x ו-y עלים אחים נמוכים ביותר.

הוכחה:

יהי T עץ אופטימלי: בה"כ, $f_x \leq f_y$ וגם $f_a \leq f_b$.



ניצור עץ חדש T' בו נחליף בין x ו-a ובין y ו-b.
מהו השינוי בערך של T?
$l_T(x) \cdot f_x + l_r(y) \cdot f_y + l_T(a) \cdot f_a + l_T(b) \cdot f_b$ (תרומת התווים a, b, x, y ל-T.
$l_T(x) \leq l_T(a)$ וגם $f_x \leq f_a$. בופן דומה, $l_t(y) \leq l_t(b)$ וגם $f_y \leq f_b$.
$\Leftarrow$ אם נבצע את ההחלפה, ערכו של T לא יכול לגדול.

מסקנה:

נסמן ב-x וב-y את שני התווים בעלי תדירויות הנמוכות ביותר ונסמן ב-z את התו המלאכותי שהוספנו במקום שניהם.
נסמן ב-$cost(T)$ את ערך העץ T.
נסמן ב-T' את העץ המתקבל מהקריאה הרקורסיבית, אזי:

$$cost(T) = cost(T') - l_T(z) \cdot f_z + (l_{T'}(Z) + 1)(f_x + f_y) = cost(T') + f_x + f_y$$

נכונות האלגוריתם נובעת מטענה 2 והמסקנה (ניתן להוכיח בשלילה).