בעיית חיתוך המוט [או משהו כזה]

נניח שיש לנו מוט באורך מסוים, k, ואנו רוצים לחתוך אותו לכמה מוטות במקומות ספציפיים.
ונניח שחיתוך מוט באורך x עולה $f(x)$.

לשם הדגמה, נניח כי $f(x)=x$ (כלומר חיתוך מוט באורך x עולה x, ולכן חיתוך מוט ארוך יותר זו פעולה יקרה יותר).

לדוגמה:

[ציור]

אם נחתוך קודם ב-a_2 ואז ב-a_4, נקבל:

[חישוב]

סה"כ עלות של 18.

לעומת זאת, אם נחתוך קודם ב-a_4 ואז ב-a_2, נקבל:

[חישוב]

סה"כ עלות של 14.

לכן אנו רואים שיש הבדל בעלות הכוללת עבור סדר חיתוכים שונה.
נרצה למצוא את הסדר שבו העלות תהיה מינימלית.

הגדרה פורמלית

- <u>מופע</u>: פונקציית עלות $f(x)$, אורך מוט וסדרה של מקומות לחיתוך $(a_1,\dots,a_n)$ כך ש-$0<a_1<a_2<\cdots<a_n<k$, $a_i\in\mathbb{N}$.
- <u>פתרון חוקי</u>: סדרה של חיתוכים $\left(a_{i_1},a_{i_2},\dots,a_{i_n}\right)$ כך שכל איבר מופיע פעם אחת בלבד [[כלומר $\{i_1,\dots,i_n\}=\{1,\dots,n\}$.
- <u>עלות</u>: עבור פתרון U, נגדיר את $cost(U)$ כסכום עלויות החיתוכים.

פתרון

הגדרת OPT

נגדיר $OPT(i,j)=$ העלות של חיתוכים אופטימליים עבור מוט שמתחיל ב-a_i ונגמר ב-a_j.

למשל, אם נרצה לדעת מה העלות הטובה ביותר לחיתוך מוט שמתחיל ב-a_i ונגמר ב-a_j כאשר בוחרים לבצע את החיתוך הראשון ב-a_{i+1}, נקבל את הערך:

$$f\left(a_j-a_i\right)+OPT\left(i,i+1\right)+OPT\left(i+1,j\right)$$

[זאת משום שהעלות לבצע את החיתוך הראשון היא $f\left(a_j-a_i\right)$ (כי אורך המוט שמתחיל ב- a_i ונגמר ב- a_j הוא a_j-a_i), והעלות הטובה ביותר לחיתוך החלקים שמתקבלים היא $OPT\left(i,i+1\right)$ (שזו העלות האופטימלית לחיתוך המוט שמתחיל ב- a_i ונגמר ב- a_{i+1}), ו- $OPT\left(i+1,j\right)$ (שזו העלות האופטימלית לחיתוך המוט שמתחיל ב- a_{i+1} ונגמר ב- a_j).

נרשום טבלה של הערכים הטובים ביותר לחיתוך מוט שכזה עבור כל אפשרות לנקודת חיתוך ראשונה:

a_{i+1}	$f\left(a_j-a_i\right)+OPT\left(i,i+1\right)+OPT\left(i+1,j\right)$
a_{i+2}	$f\left(a_j-a_i\right)+OPT\left(i,i+2\right)+OPT\left(i+2,j\right)$
$\vdots$	$\vdots$
a_{i+l}	$f\left(a_j-a_i\right)+OPT\left(i,i+l\right)+OPT\left(i+l,j\right)$
$\vdots$	$\vdots$
a_{j-1}	$f\left(a_j-a_i\right)+OPT\left(i,j-1\right)+OPT\left(j-1,j\right)$

כמובן, כיוון שאלה כל האפשרויות, האפשרות האופטימלית (עם המחיר המינימלי) תתקבל בתור המינימום שלהן:

$$OPT\left(i,j\right)=\min_{i<l<j}\left\{f\left(a_j-a_i\right)+OPT\left(i,l\right)+OPT\left(l,j\right)\right\}$$

נוסיף מקרי בסיס ונקבל:

$$OPT\left(i,j\right)=\begin{cases}0, & j=i+1\\ \min\limits_{i<l<j}\left\{f\left(a_j-a_i\right)+OPT\left(i,l\right)+OPT\left(l,j\right)\right\}, & j>i+1\end{cases}$$

[[מדוע $OPT\left(i,i+1\right)=0$?
כי זוהי העלות הטובה ביותר לחיתוך מוט שמתחיל ב- a_i ונגמר ב- a_{i+1} – אבל אנחנו לא רוצים לחתוך את המוט הזה, לכן העלות היא 0.]]

טענה 1 לנכונות הנוסחה

אם $j>i+1$ אז לכל $i<l<j$ קיים פתרון שעלותו היא:

$$cost\left(U\right)=f\left(a_j-a_i\right)+OPT\left(i,l\right)+OPT\left(l,j\right)$$

<u>הוכחה:</u>

יהיו U_1,U_2 פתרונות אופטימליים ל- $\left(a_i,a_l\right)$ ו- $\left(a_l,a_j\right)$ בהתאמה.

אז $\left(a_l\right)\circ U_1\circ U_2$ הוא פתרון חוקי בעלות:

$$cost(U) = f(a_j - a_i) + OPT(i,l) + OPT(l,j)$$

טענה 2 לנכונות הנוסחה

אם $j > i+1$ אז קיים $i < l < j$ כך ש:

$$OPT(i,j) = f(a_j - a_i) + OPT(i,l) + OPT(l,j)$$

<u>הוכחה:</u>

יהי U פתרון אופטימלי [[כלומר כך ש- $cost(U) = OPT(i,j)$]].

נגדיר את l להיות מיקום החיתוך הראשון ב- U [[כלומר נניח שהחיתוך הראשון ב- U מתבצע ב- a_l]].

יהיו U_1 ו- U_2 תתי-הסדרות של U עבור $[a_i, a_l]$ ו- $[a_l, a_j]$ בהתאמה:

$$U_1 = \{a_s | a_s < a_l, a_s \in U\}$$
$$U_2 = \{a_s | a_s > a_l, a_s \in U\}$$

[[הערה: זה לא באמת מגדיר נכון את הפתרונות, כי אלה בכלל קבוצות ללא סדר, והפתרונות הן סדרות. הפואנטה היא ש- U_1 זו תת-הסדרה שמתאימה ל- $[a_i, a_l]$ ו- U_2 זו תת-הסדרה שמתאימה ל- $[a_l, a_j]$.]]

אזי:

$$OPT(i,j) = cost(U) = f(a_j - a_i) + cost(U_1) + cost(U_2) =$$
$$= f(a_j - a_i) + OPT(i,l) + OPT(l,j)$$

[[פה המתרגל קפץ קדימה, אבל לדעתי יש להוכיח את השוויון האחרון המסומן באדום, לכן אוסיף הוכחה קצרה משלי:

אנו רוצים לומר ש- $cost(U_1) = OPT(i,l)$ ו- $cost(U_2) = OPT(l,j)$.

בוודאות $cost(U_1) \geq OPT(i,l)$, כיוון ש- $OPT(i,l)$ מוגדר להיות העלות הנמוכה ביותר מבין כל העלויות של כל הפתרונות (וקיימת עלות מינימלית כיוון שקבוצת כל הפתרונות היא סופית).

אם $cost(U_1) > OPT(i,l)$ אז קיים פתרון U_1' כך ש- $cost(U_1') = OPT(i,l) < cost(U_1)$.

לכן אם נחליף ב- U את תת-הסדרה המתאימה ל- U_1 בזו שמתאימה ל- U_1', נקבל פתרון U' שעלותו היא:

$$cost(U') = f(a_j - a_i) + cost(U_1') + cost(U_2) <$$
$$< f(a_j - a_i) + cost(U_1) + cost(U_2) = cost(U)$$

בסתירה לכך ש- U בעל עלות מינימלית (כלומר ש- $cost(U)=OPT(i,j)$).]]

מימוש

נציג שני מימושים אפשריים:

מימוש רקורסיבי

$\text{Rec\_cut\_order}(i,j)$:

1. If $j=i+1$:
    1.1. Return 0
2. $Min \leftarrow \infty$
3. For $l \leftarrow i+1,\ldots,j$ do:
    3.1. $Value \leftarrow f(a_j-a_i)+\text{Rec\_cut\_order}(i,l)+\text{Rec\_cut\_order}(l,j)$
    3.2. If $Value < Min$:
        3.2.1. $Min \leftarrow Value$
4. Return Min

במימוש זה נקבל עץ קריאות רקורסיבי מפלצתי:

[עץ!]

Memoization

$\text{Memoized}(k)$:

1. For $i \leftarrow 1,\ldots,n$ do:
    1.1. For $j \leftarrow 1,\ldots,n$ do:
        1.1.1. $M[i,j] \leftarrow \infty$
2. $\text{Lookup}(0,n+1)$

$\text{Lookup}(i,j)$:

1. If $M[i,j] < \infty$:
    1.1. Return $M[i,j]$
2. If $j=i+1$:
    2.1. $M[i,j] \leftarrow 0$
    2.2. Return 0
3. For $l \leftarrow i+1,\ldots,j-1$ do:
    3.1. $Value \leftarrow f(a_j-a_i)+\text{Lookup}(i,l)+\text{Lookup}(l,j)$

3.2. If $M\left[i,j\right]>Value$:

3.2.1. $M\left[i,j\right]\leftarrow Value$

4. Return $M\left[i,j\right]$

[הערה: הוספנו לסדרה שלנו $a_1,\ldots,a_n$ איברים נוספים, $a_0=0$ וְ- $a_{n+1}=k$, ע"מ שלא נחרוג מגבולות המטריצה בצורה פשוטה יותר שדורשת פחות בדיקות בקוד.
שימו לב: למעשה, אנו מחפשים את $OPT\left(0,n+1\right)$, שכן $OPT\left(1,n\right)$ מדבר על חיתוך מוט שמתחיל ב- a_1 ונגמר ב- a_n, אבל המוט שלנו מתחיל ב-0 (משמאל ל- a_1) ומסתיים ב- k (מימין ל- a_n).]

סדר מילוי התאים:

[טבלה]

ניתוח זמן ריצה

כמו בדוגמאות קודמות.

זה יוצא $O\left(n^3\right)$.

מימוש איטרטיבי [כי רקורסיה זה איכסה]

$\text{Iter\_cut\_order}\left(k,a\right)$:

1. For $i\leftarrow 0,\ldots,n$ do:

1.1. $M\left[i,i+1\right]\leftarrow 0$

2. For $diff\leftarrow 2,\ldots,n+1$ do:

2.1. For $i\leftarrow 0,\ldots,\left(n+1-diff\right)$ do:

2.1.1. $Min\leftarrow\infty$

2.1.2. For $l\leftarrow i+1,\ldots,\overbrace{i+diff}^{j}-1$ do:

2.1.2.1. If $f\left(\overbrace{a_{i+diff}-a_i}^{a_j-a_i}\right)+M\left[i,l\right]+M\left[l,j\right]<Min$:

2.1.2.1.1. $Min\leftarrow f\left(a_{i+diff}-a_i\right)+M\left[i,l\right]+M\left[l,j\right]$

2.1.3. $M\left[i,i+diff\right]\leftarrow Min$

דוגמת ריצה

עבור הנתונים:

$$f(x)=k-x$$
$$k=10, a_1=2, a_2-3, a_3=6, a_4=9$$

נקבל:

[מצגת]

שחזור הפתרון מהטבלה

מהטבלה שנוצרת במימוש ה-memoization ובמימוש האיטרטיבי (זו אותה הטבלה) ניתן לשחזר את סדרת החיתוכים.
נעשה זאת רקורסיבית:

$\underline{\mathrm{Rec\_cut\_seq}(i,j):}$

1. If $j=i+1$:
    1.1. Return $(\ )$ (סדרה ריקה)
2. $Min \leftarrow \infty$
3. For $l=i+1,\dots,j-1$:
    3.1. If $M[i,l]+M[l,j]<Min$:
        3.1.1. $Min \leftarrow M[i,l]+M[l,j]$
        3.1.2. $Cut \leftarrow l$
4. Return $(l)\circ \mathrm{Rec\_cut\_seq}(i,l)\circ \mathrm{Rec\_cut\_seq}(l,j)$