10 הרצאה

תכנון דינאמי

כפל מטריצות, התאמת מחרוזות

אופטימזציה של כפל מטריצות

תזכורת: כפל נאיבי של מטריצה בגודל a imes b עם מטריצה בגודל כפל לוקח לוקח a imes b עם מטריצה של מטריצה בגודל כפל מטריצה מ

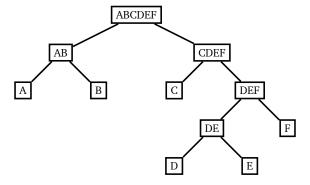
כאשר כופלים n מטריצה מגודל $x_i imes y_i$ מגדלים מהתאמה, אז תוצאת המכפלה תהיה מטריצה מגודל מגדלים A_1, \dots, A_n מספר מפעולות שיש לבצע תלוי בסדר בו נבחר לבצע את המכפלה.

? ABC ממה פעולות נבצע כדי לבצע את המכפלה

$$\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_{100} \end{pmatrix} \begin{pmatrix} b_1 & b_2 & \dots & b_{100} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{100} \end{pmatrix}$$

ו- AB אם נבצע את המכפלה לפי הסדר משמאל לימין אז נזדקק ל- $100\cdot 1\cdot 100=10,000\cdot 1\cdot 100$ פעולות עבור הכפל של אז נזדקק לסדר גודל של אז נזדקק לסדר גודל של A(BC). אם נחשב את המכפלה A(BC) אז נזדקק לסדר גודל של 200 פעולות בלבד A(BC)

בעיה: בהינתן n מטריצות, A_1,\dots,A_n מגדלים A_1,\dots,A_n מגדלים בחראמה, רוצים לחשב הינתן A_1,\dots,A_n מטריצות A_1,\dots,A_n מגדלים בעזרת עץ, למשל העץ הבא מתאים לחישוב לסדר הפעולות הוא בעזרת עץ, למשל העץ הבא מתאים לחישוב ייצוג טבעי לסדר הפעולות הוא בעזרת עץ, למשל העץ הבא מתאים לחישוב לחישור הפעולות הוא בעזרת עץ, למשל העץ הבא מתאים לחישור לחישור לחישור הפעולות הוא בעזרת עץ, למשל העץ הבא מתאים לחישור לובי לחישור לחישור לחישור לחישור לחי



נתייחס לעץ כזה כעץ ביטוי, עץ ביטוי הוא עץ בינרי מלא שבו העלים הם המטריצות מהקלט וכל צומת מייצג מכפלה של המטריצות המתאימות לעלים של תת העץ שלו.

אלגוריתם: עבור כל שצריך כדי לבצע מספר הפעולות מספר להיות לגוריתם: 1 בור כל את נגדיר את אלגוריתם: $1 \leq i \leq j \leq n$ אלגוריתם: עבור כל $A_i \cdot A_{i+1} \cdot \ldots \cdot A_j$

אז מתקיים ש:

$$\alpha(i,j) = \min_{i \le k < j} \alpha(i,k) + \alpha(k+1,j) + x_i \cdot y_k \cdot y_j$$

בנוסף מתקיים ש:

$$\forall 1 \leq i \leq n \ \alpha(i,i) = 0$$

סיבוכיות: אם מחשבים את ערכי נוסחת הנסיגה על ידי שימוש בטבלה, למשל, אז נדרש לחשב $O(n^2)$ ערכים. זמן החישוב של כל ערך הוא O(n) ולכן בסך הכל זמן ריצת האלגוריתם הוא:

התאמת מחרוזות

רוצים לבצע תיקון של שגיאות איות, למשל:

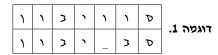


כדי לדעת אילו תיקונים להציע רוצים למדוד את המרחק בין המחרוזת שהוקלדה לבין המילה המוצעת. בהינתן א"ב $\Sigma' = \Sigma \cup \{\ \}$ ונגדיר:

s'אס אווי ה- פ's' מקבלים את $s' \in \Sigma^*$ אם לאחר מחיקת כל תווי ה- $s' \in \Sigma^{\prime *}$ מקבלים את את

בהינתן פונקציית משקל $w:\Sigma' imes\Sigma' o\mathcal{R}$ המרחק בין שתי הרחבות בעלות אורך זהה, l, הוא:

$$\sum_{i=1}^{l} w(s_1'[i], s_2'[i])$$



١	١	_	3	,	١	١	D	דוגמה 2.
١	١	,	3	_	_	_	D	

עבור פונקציית המשקל

$$w(\alpha, \beta) = \begin{cases} 0 & \text{if } \alpha = \beta \\ 1 & \text{otherwise} \end{cases}$$

אז המרחק בדוגמה 1 הוא 2 ובדוגמה 2 הוא 4.

הגדרה 2 (מרחק). הערחק בין שתי עחרוזות (לאו דווקא באורך זהה) עעל Σ הוא הערחק העיניעלי האפשרי בין כל שתי הרחבות שלהן עאורך זהה.

הערה: אם מניחים שפונקציית המשקל אי שלילית אז מספר ההרחבות הרלוונטיות הוא סופי.

מטרה: בהינתן שתי מחרוזות רוצים לחשב את המרחק ביניהן.

 $r[j\ldots n-1]$. ל- $s[i\ldots m-1]$ ל-היות המרחק בין ל-היות המרחק וויר|r|=n , |s|=m נסמן מסמור מחור.

$$\alpha(i,j) = \min \begin{cases} w(s[i],r[j]) + \alpha(i+1,j+1]) \\ w(_,r[j]) + \alpha(i,j+1) \\ w(s[i],_) + \alpha(i+1,j) \end{cases}$$

כמו כן מתקיים ש:

$$\begin{split} &\alpha(m,n) = 0 \\ &\alpha(m,k) = w(_,r[k]) + \alpha(m,k+1]) \quad \forall \ 0 \leq k < n \\ &\alpha(k,n) = w(s[k],_) + \alpha(k+1,n]) \quad \forall \ 0 \leq k < m \end{split}$$

סיבוכיות: אם מחשבים את ערכי נוסחת הנסיגה על ידי שימוש בטבלה, למשל, אז נדרש לחשב mn ערכים וחישוב של כל ערך לוקח O(mn) פעולות. בסך הכל מקבלים O(mn) פעולות.