תרגול 3
ניתוח לשיעורין

תאריך עדכון אחרון: 27 בפברואר 2011.

ניתוח לשיעורין $(amortized\ analysis)$ הוא טכניקה לניתוח זמן ריצה לסדרת פעולות, אשר מאפשר קבלת חסמי זמן ריצה נמוכים יותר מאשר חסמים המתקבלים כאשר מניחים את המקרה הגרוע ביותר בכל פעולה. בדרך כלל אנחנו מתייחסים למספר פעולות, כאשר חלק מהפעולות יקרות יותר, וחלקן ־ זולות יותר. בניתוח לשיעורין אנו מדברים על עלות כל פעולה בממוצע, עבור כל רצף של פעולות אפשרי של שימוש במבנה הנתונים. אנו מדברים על המקרה הגרוע ביותר; כלומר, לוקחים בחשבון את הסידרה הגרועה ביותר של פעולות שיכולה להיות.

ניתוח לשיעורין היא איזושהי אסטרטגיה לניתוח כל רצף של פעולות המראה שממוצע העלות לכל פעולה הוא קטן, למרות שישנן פעולות יחידות בתוך הרצף שעלולות להיות יקרות. נעיר שלמרות שאנו מדברים על ממוצעים, אנו לא מערבים הסתברויות בניתוח.

המטרה היא חישוב חסם עליון הדוק ככל שניתן לעלות של רצף כלשהו של n פעולות. נניח עלות הפעולה ה ־ i היא c_i, המטרה היא לחשב $T(n)=\sum_{i=1}^{n} c_i$. אזי עלות כל פעולה בממוצע היא $T(n)/n$.

נראה שלוש שיטות לניתוח לשיעורין ־ שיטת הצבירה, שיטת החיובים (שיטת ה"בנק") ושיטת הפוטנציאל. שיטת הפוטנציאל היא החשובה ביותר.

3.1 דוגמא ־ מחסנית

נתבונן במחסנית התומכת בפעולות הבאות:

- $\texttt{push}(x)$ ־ מכניס את x למחסנית. עלות ־ $O(1)$.
- $\texttt{pop}(x)$ ־ להוציא את האיבר האחרון שנכנס (אם יש כזה). עלות ־ $O(1)$.
- $\texttt{multi-pop}(x)$ ־ מוציאה את כל האיברים מהמחסנית. עלות: אם יש k איברים כרגע במחסנית אזי ־ $O(k)$.

נממש (בפסאדו קוד) את $\texttt{multi-pop()}$:

```
multi-pop()
    while not isEmpty()
        pop()
```

נשים לב שכל הפעולות עולות $O(1)$, מלבד הפעולה $\texttt{multi-pop}$. בפרט, הפעולה הנ"ל נראית יקרה ־ היא לינארית במספר האיברים במחסנית. כך, אם ישנם n איברים במחסנית, עלותה ־ $O(n)$ (מה שנשמע רע מאוד ביחס ל ־ $O(1)$). אבל ־ האם אנחנו לא מחמירים מדי כשאנו מנתחים את הפעולה בצורה כזו? כדי לחדד את השאלה, נתבונן בניתוח השגוי הבא. נשאל עבור סידרה של n פעולות ־ מהי העלות $worst\ case$ לסידרה. כעת, יכולים לטעון (בטעות) שכל פעולה מסוג $\texttt{multi-pop}$ עולה $O(n)$ בכל פעם, ולכן בסה"כ רצף של n פעולות עלול להגיע ל ־ $O(n^2)$. אך, ניתוח זה לא הדוק.

אצלינו, בכדי שיהיו n איברים במחסנית, צריכים להיות n פעולות push לפני כן. אם נתונים לנו רצף של פעולות מסוימות, נניח ־ n פעולות push ולאחריהן פעולת multi־pop ־ עבור n פעולות ה־push אנחנו משלמים n בסה"כ, ועל פעולת multi־pop אנו משלמים n. כלומר בסה"כ אנו משלמים עבור רצף $n+1$ הפעולות $2n$, מה שאומר שבממוצע לכל פעולה אנו משלמים 2. כלומר, העלות הממוצעת לכל פעולה ־ היא $O(1)$.
כעת נראה שיטות ואסטרטגיות לחישוב של ניתוח לשיעורין.

3.2 שיטת הצבירה

בשיטת הצבירה אנו מראים שלכל n, סדרה של n פעולות צורכת זמן של $T(n)$ לכל היותר. נקבל אם כן, שממוצע זמן של כל פעולה, או העלות לשיעורין של כל פעולה, הינה $T(n)/n$. זוהי העלות לשיעורין של כל פעולה גם כאשר הסידרה מכילה סוגים שונים של פעולות.

בדוגמת המחסנית ננתח כמה עולה n פעולות על המחסנית. יש שלוש פעולות אפשריות: push, pop, multi־pop. אמנם multi־pop עולה $O(n)$ במקרה הגרוע, אך נשים לב שכל עצם ניתן לשלוף רק פעם אחת עבור כל פעם שהוא נדחף למחסנית. לכן, מספר הקריאות ל pop (גם אלו שבתוך המימוש של multi־pop) הוא לכל היותר מספר הקריאות ל ־ push. מכיוון שאנו מתבוננים בסידרה מאורך n, אין יותר מ ־ n קריאות ל ־ push, ולפיכך מספר הקריאות הכולל ל ־ pop (גם אלו שבתוך multi־pop) הוא לכל היותר n. נקבל כי העלות הכוללת היא לכל היותר $2n$, ולכן בממוצע עלות כל פעולה היא 2, כלומר $O(1)$.

3.2.1 יותר פורמלי

בשיטת הצבירה אנו מחשבים סכום כל העלויות עבור סידרה כללית של פעולות. מכיוון שאנו מתבוננים על סידרה כללית, הניתוח יכסה גם את הסידרה הגרועה ביותר. בכדי לדבר על סידרה כלשהי ־ נתבונן בסידרה כללית של פעולות:

$$S = (\sigma_1, \sigma_2, \ldots, \sigma_n)$$

כל פעולה $\sigma_i \in \{1,2,3\}$, כאשר ־ 1 מציין את הפעולה push, 2 מציין ־ pop, ו ־ 3 מציין ־ multi־pop.
אם עלות הפעולה ה ־ i היא c_i, נרצה לחשב:

$$T(n) = \sum_{k=1}^{n} c_k$$

נתבונן בכל המקומות בסידרה כך ש ־ $\sigma_i = 3$ (multi־pop). נניח שכל הפעולות הנ"ל נמצאות במקומות $I = \{i_1, \ldots, i_t\}$, וכך ש ־ $i_1 < i_2 < \ldots < i_t$. (נעיר שאם I ריקה, אזי $T(n)$ הוא סכום של קבועים (אין פעולת multi־pop) ולכן הסכום כולו לינארי ב ־ n). בנוסף, נגדיר כי ־ $i_0 = 0$.
נסמן את סידרת הפעולות של S בין המקומות i_{j-1} ובין i_j ב ־ S_{i_j}. כלומר, הסידרה:

$$S_{i_j} = \{\sigma_{i_{j-1}+1}, \ldots, \sigma_{i_j}\}$$

כעת, נראה את הטענה הבאה:

טענה 3.1 לכל $1 \leq j \leq t$, עלות הפעולה *multi־pop* הנמצאת במקום i_j (כלומר, עלות הפעולה σ_{i_j}) הינה $i_j - i_{j-1} - 1$ לכל היותר. בנוסף, עלות הסידרה S_{i_j} הינה $2(i_j - i_{j-1} - 1)$.

הוכחה: יהי j נתון. לאחר הפעולה $\sigma_{i_{j-1}}$ המחסנית ריקה (שכן, פעולה זו הייתה פעולת multi־pop). כל שאר הפעולות אינן פעולות multi־pop (חתכנו את סדרת הפעולות בין זוג פעולות multi־pop). לכן, לכל פעולה בסדרה הנ"ל, הפעולה היא push או pop. במקרה הגרוע ביותר ־ כל הפעולות הן push, מה שאומר שכאשר

מגיעים לפעולה σ_{i_j} (פעולת ה ־ multi־pop) ־ ישנם $i_j - i_{j-1} - 1$ איברים במחסנית. לפיכך, העלות של פעולת ה ־ multi־pop היא $i_j - i_{j-1} - 1$. לכן, העלות הכוללת לסידרה S_{i_j} היא לכל היותר ־ $2(i_j - i_{j-1} - 1)$.
■

כעת, לכל סידרה S ניתן להתייחס ל ־ $I = \{i_1, \ldots, i_t\}$. נקבל:

$$T(n) \leq \sum_{j=1}^{t} 2(i_j - i_{j-1} - 1) + \sum_{j=i_t+1}^{n} c_j = 2(i_t - i_0) - 2t + (n - i_t)$$

נזכור ש ־ i_t הינו איזשהו אינדקס בתחום $1 \leq i_t \leq n$, וכמו־כן ־ i_0 הינו ־ 0. לפיכך:

$$T(n) = 2(i_t - i_0) - 2t + (n - i_t) = 2i_t - 2t + n - i_t = n + i_t - 2t \leq 2n - 2t \leq 2n$$

כלומר, הצלחנו להראות שלכל סידרה של פעולות מאורך n, העלות הכוללת היא לכל היותר $2n$. הנ"ל מראה שעלות ממוצעת של כל פעולה היא $O(1)$.

3.3 שיטת החיובים ("שיטת הבנק")

בשיטה זו ־ ניתן לכל פעולה עלות שונה מזו שהיא עולה באמת. עלות זו נקראת "עלות לשיעורין". חלק מהפעולות יקבלו עלות גדולה יותר מהעלות האמיתית שלהן; חלק מהפעולות יקבלו עלות קטנה יותר מזו שהן עולות בפועל (בדרך כלל, הפעולה היקרה ביותר בפועל תקבל את העלות הקטנה ביותר לשיעורין). העיקרון שצריך תמיד לשמור עליו, הוא שסידרה של n פעולות (לכל n) לפי העלויות לשיעורין שלנו ־ תהיה גדולה יותר מהעלות האמיתית שלהן בפועל.

נשתמש ב"בנק" בשביל "לשמור" לנו עלויות. נניח שעל פעולת push אנחנו משלמים 2 יחידות במקום יחידה אחת; יחידה אחת תהיה העלות האמיתית של הפעולה (שהיינו צריכים לשלם ממילא), והיחידה השנייה פשוט תכנס ל"בנק" בשביל "התחשבנות עתידית". כעת, כאשר נבצע פעולת multi־pop ־ לא נשלם מ"הכיס" אלא נמשוך את העלויות שצברנו בבנק; מכיוון שעל כל איבר שיש כרגע במחסנית ־ כבר צברנו יחידה אחת בבנק, נקבל כי יש בבנק מספיק כסף בשביל לממן את הפעולה multi־pop, ולכן לא נשלם עבור פעולה זו מהכיס בכלל.

הסבר נוסף ־ כאשר אני מכניס איבר למחסנית (ומשלם עליו 1), אני משלם גם עבור ההוצאה שלו כבר בזמן ההכנסה (עוד יחידה אחת). לכן, בסה"כ הכנסה עולה 2 יחידות; כעת, הוצאה יחידה, או הוצאה ע"י multi־pop הן למעשה ־ בחינם (שילמנו עליהן קודם; בבנק יש מספיק כסף בשביל לשלם על ההוצאה שלהן).

יותר פורמלי. נסמן ב־ $\hat{c}_i$ את העלות לשיעורין של הפעולה ה ־ i. נסמן ב ־ c_i את העלות האמיתית של הפעולה ה־ i. העלות לשיעורין של הפעולה ה ־ i בשיטת הבנק הינה:

$$\hat{c}_i = c_i + deposit - withdraw$$

כלומר ־ עלות הפעולה היא העלות האמיתית + ההפקדה שאנו מבצעים לבנק, פחות העלות של המשיכה (במידת הצורך).

כאשר אנו שומרים על העיקרון שלעולם "לא ניכנס למינוס" (לעולם לא נמשוך מהבנק סכום כסף שלא הפקדנו אותו לפני כן), מתקיים שסך כל ההפקדות (סכום כל הערכים של $deposit$) גדול מכל הפעמים שמשכנו מהבנק (סכום כל הערכים שביצענו $withdraw$), ולכן:

$$\sum_{i=1}^{n} \hat{c}_i \geq \sum_{i=1}^{n} c_i = T(n)$$

כלומר, העלות האמיתית של n הפעולות היא קטנה מסכום העלות לשיעורין. כעת, ננתח כל אחת מפעולות המחסנית:

- עבור פעולת `push` ־ עלות הפעולה היא 1, בנוסף, מכניסים 1 לבנק. נקבל:

$$\hat{c}_i = 1 + 1 - 0 = 2$$

- עבור פעולת `pop` ־ עלות הפעולה היא 1. לא מפקידים לבנק, ולא מושכים ממנו[1]. נקבל:

$$\hat{c}_i = 1 + 0 - 0 = 1$$

- עבור פעולת `multi-pop` ־ נניח שמספר האיברים כרגע הוא k. עלות הפעולה אם כן היא k. כעת, נמשוך מהבנק k יחידות (בדוק שהבנת מדוע אנו בטוחים כי בבנק יש k יחידות). נקבל:

$$\hat{c}_i = k + 0 - k = 0$$

אם כן, העלות לשיעורין של כל פעולה קטן מ ־ 2. מכיוון שלעולם לא נכנסים למינוס, נקבל כי:

$$\sum_{i=1}^{n} \hat{c}_i \geq \sum_{i=1}^{n} c_i = T(n)$$

אצלינו, לכל i, $\hat{c}_i \leq 2$. כלומר:

$$T(n) \leq \sum_{i=1}^{n} \hat{c}_i \leq 2n$$

ולכן סכום התשלומים עבור n פעולות קטן מ ־ $2n$. כלומר, העלות הממוצעת לכל פעולה היא 2.
חשוב לציין שכאשר מבצעים ניתוח בשיטת הבנק ־ חייבים תמיד לשמור על האינווריאנטה שסכום הכסף בבנק הוא חיובי. אסור בשום אופן "להכנס למינוס", שכן כאשר אנו "נכנסים למינוס" ־ העלות האמיתית של הפעולות היא פחות מהעלות לשיעורין שאנו מבצעים, ואז לא נוכל לטעון שהעלות הכוללת של העלויות לשיעורין גדולה מהעלויות האמיתיות.

3.4 שיטת הפוטנציאל

שיטת הפוטנציאל דומה מאוד לשיטת הבנק, רק שאנו ממדלים אותה באופן יותר מתמטי, ולכן היא יותר פורמלית (ופחות אינטואיטיבית..).
בשיטת הבנק "התשלום מראש" היה שמור כיתרה לזכותם של עצמים ספציפיים במבנה הנתונים (כל איבר שמר לעצמו את העלות להוצאתו בעתיד). כאן, ה"תשלום מראש" מובע בעזרת "אנרגיה פוטנציאלית" שניתן לשחרר כדי לשלם עבור פעולות בעתיד.
כמו בשיטת הבנק ־ פעולת `push` תגדיל לנו את הפוטנציאל של מבנה הנתונים (ונשלם על הגדלה זו), ופעולת `multi-pop` תשולם בעזרת "פירוק הפוטנציאל הזה", או, ריקון הפוטנציאל (ואין צורך לשלם עליה ־ כי כבר שילמנו על הפוטנציאל עצמו).

[1] יכולנו להגדיר שגם במקרה זה אנו מושכים את העלות מהבנק, ולא משלמים מהכיס. במקרה זה נקבל כי העלות לשיעורין היא 0. נציין שעדיין לא נכנס למינוס, וכל המשך הניתוח הוא בדיוק אותו הדבר.

שיטת הפוטנציאל. יהי D_0 המצב ההתחלתי של מבנה הנתונים שעליו מבוצעות n פעולות. נסמן ב ־ c_i את העלות של הפעולה ה ־ i. D_i הינו המצב של המערכת לאחר הפעולה ה ־ i, על המצב D_{i-1}. כלומר, מתחילים ממצב D_0, מבצעים פעולה כלשהי (σ_1) ומגיעים למצב D_1. לאחר מכן מבצעים פעולה כלשהי (σ_2) ומגיעים למצב D_2, וכן הלאה.

פונקציית הפוטנציאל ϕ ממפה כל מצב של מבנה הנתונים D_i למספר ממשי שמציין את הפוטנציאל המיוחס למבנה הנתונים.

נגדיר את העלות לשיעורין של הפעולה ה ־ i כ:

$$\hat{c}_i = c_i + \phi(D_i) - \phi(D_{i-1})$$

כלומר, העלות לשיעורין של הפעולה ה־ i היא העלות האמיתית c_i + הפרש הפוטנציאלים של מבנה הנתונים. במילים אחרות, העלות האמיתית של הפעולה ה ־ i הינה:

$$c_i = \hat{c}_i + \phi(D_{i-1}) - \phi(D_i)$$

העלות של כל סידרה כלשהי של n פעולות:

$$\begin{aligned} T(n) &= \sum_{i=1}^{n} c_i = \sum_{i=1}^{n} [\hat{c}_i + \phi(D_{i-1}) - \phi(D_i)] \\ &= \sum_{i=1}^{n} \hat{c}_i + \sum_{i=1}^{n} [\phi(D_{i-1}) - \phi(D_i)] \\ &= \sum_{i=1}^{n} \hat{c}_i + \phi(D_0) - \phi(D_n) \end{aligned}$$

כאשר הצעד האחרון נכון מכיון שהטור הינו טלסקופי. אם תמיד יתקיים $\phi(D_0) \leq \phi(D_n)$, נקבל כי :

$$T(n) = \sum_{i=1}^{n} \hat{c}_i + \phi(D_0) - \phi(D_n) \leq \sum_{i=1}^{n} \hat{c}_i$$

כלומר,

$$T(n) = \sum_{i=1}^{n} c_i \leq \sum_{i=1}^{n} \hat{c}_i$$

גם פה, כמו בשיטת הבנק, אנו שומרים על כך שסכום עלות n פעולות קטן יותר מסכום ה"עלות לשיעורין". לכן, הניתוח הופך לפשוט יותר: כל שצריך הוא לחשב את $\hat{c}_i$ לכל אחת מהפעולות הקיימות על מבנה הנתונים.

ניתוח מחסנית בעזרת שיטת הפוטנציאל. נגדיר את פונקציית הפוטנציאל $\phi(D_i)$ כמספר האיברים שיש כרגע (= מצב D_i) במחסנית. נקבל:

$$\phi(D_0) = 0$$

ובנוסף:

$$\forall 1 \leq i \leq n, \phi(D_i) \geq 0$$

כלומר, פונקציית פוטנציאל זו טובה ועונה על הדרישה.

הפיתוח הנ"ל אומר לנו שבכדי להראות שהעלות הממוצעת של כל פעולה בכל סדרה הינו קבוע, מספיק להראות ש־ $\hat{c}$ קבוע. לכן, נחשב עבור כל אחד מסוגי הפעולות.

- עבור פעולת push ־ עלות הפעולה הוא 1, ונניח שהיו t איברים במחסנית. לאחר הפעולה, ישנם $t+1$ איברים במחסנית. נקבל:

$$\hat{c}_i = c_i + \phi(D_i) - \phi(D_{i-1}) = 1 + (t+1) - t = 2$$

- עבור פעולת pop ־ עלות הפעולה הוא 1, ונניח שהיו t איברים במחסנית. לאחר הפעולה, ישנם $t-1$ איברים במחסנית. נקבל:

$$\hat{c}_i = c_i + \phi(D_i) - \phi(D_{i-1}) = 1 + t - 1 - t = 0$$

- עבור פעולת multi־pop. נגיד שיש כרגע t איברים במחסנית. אזי, עלות הפעולה הוא ־ t, ומצד שני ־ הפרש הפוטנציאלים הוא $-t$ (לפני הפעולה היו t איברים במחסנית, ולאחריה ־ 0). נקבל אם כן:

$$\hat{c}_i = c_i + \phi(D_i) - \phi(D_{i-1}) = t + 0 - t = 0$$

נקבל:

$$T(n) = \sum_{i=1}^{n} c_i \leq \sum_{i=1}^{n} \hat{c}_i \leq \sum_{i=1}^{n} 2 = 2n$$

ולכן העלות הממוצעת של כל פעולה ־ 2.

3.5 מונה בינארי

נניח יש לנו מונה עם k ביטים, הסופר בבינארית. אנחנו למעשה תומכים רק בפעולה אחת ־ $increment$ ־ העלה ב ־ 1. עלות כל פעולה = מספר הביטים שהוחלפו. לדוגמא: (לצד כל ערך בינארי מצויין עלות פעולת ה $increment$)

$$\begin{array}{ccc} 00000 & \rightarrow & 0 \\ 00001 & \rightarrow & 1 \\ 00010 & \rightarrow & 2 \\ 00011 & \rightarrow & 1 \\ 00100 & \rightarrow & 3 \\ 00101 & \rightarrow & 1 \\ 00110 & \rightarrow & 2 \\ & \vdots & \end{array}$$

כמה עולה הפעולה $increment$? כמה עולה רצף של n פעולות? שוב, ניתוח שגוי יאמר כי בכל פעם, במקרה הגרוע ביותר אנו מבצעים $O(k)$ החלפות (שכן יש k ביטים במונה). במקרה הגרוע ביותר ־ נחליף את כל הביטים במונה). ולכן, במקרה הגרוע ביותר ־ $O(k \cdot n)$. ניתוח זה שגוי. נבצע ניתוח לשיעורין.

3.5.1 ניתוח לפי שיטת הצבירה

במקרה זה, מכיוון שישנה רק פעולה אחת (*increment*) הניתוח לפי שיטת הצבירה פשוט. כל רצף של פעולות נראה בדיוק אותו הדבר.

נתבונן על n פעולות. הביט הראשון יתחלף n פעמים (בכל פעם). הביט השני לעומת זאת, יתחלף בכל פעם זוגית, כלומר ב ־ $\frac{n}{2}$ פעמים. הביט השלישי יתחלף כל פעם רביעית, כלומר בסה"כ ב ־ $\frac{n}{4}$ פעמים. הביט ה ־ i יתחלף $\frac{n}{2^{i-1}}$ פעמים. כלומר:

$$\begin{aligned} T(n) &= \text{number of bits that were changed} \\ &= \sum_{i=1}^{k} \Big[\text{number of times that the i}^{\text{th}} \text{ bit was changed}\Big] \\ &= \sum_{i=1}^{k} \frac{n}{2^{i-1}} = n\sum_{i=1}^{k} \frac{1}{2^{i-1}} \leq 2n \end{aligned}$$

כלומר, $T(n)/n \leq 2$.

3.5.2 ניתוח לפי שיטת הפוטנציאל

נגדיר פונקציאת פוטנציאל: $\phi(D_i)$ = מספר האחדות במונה[2]. מתקיים: $\phi(D_0) = 0$, ולכל $1 \leq i \leq n$ נקבל: $\phi(D_i) \geq 0$, ולכן פונקציית פוטנציאל זו עונה על הדרישה שלנו.

נראה כמה עולה פעולת *increment*. כאשר מבצעים פעולת *increment*, האפס הראשון מימין הופך ל ־ 1, וכל שאר האחדות מימין אליו ־ מתאפסים. נניח שישנם t אחדות רצופים מסוף המונה (מימין). במצב כזה, עלות פעולת *increment* תהיה $t+1$ (שכן נשנה את t האחדות לאפסים, ואת האפס שלאחריהם ל ־ 1). מצד שני, נשים לב שכעת יש פחות $t-1$ אחדות בביטוי (שכן t אחדות הפכנו לאפסים, ואפס אחד הפכנו ל ־ 1). נקבל:

$$\hat{c}_i = c_i + \phi(D_i) - \phi(D_{i-1}) = t + 1 - (t-1) = 2$$

לכן, רצף של n פעולות יעלה פחות מ ־ $2n$. כלומר, הממוצע לכל פעולה ־ 2.

3.6 מערך דינאמי

המערך תומך בפעולת *insert* ־ ומכניס את האיבר למקום הפנוי הבא. נניח שבמערך יש n מקומות, ונניח שהמערך מלא. כעת, כאשר נבקש להוסיף איבר נוסף, נבצע את הפעולות הבאות:

- נקצה מערך בגודל $2n$.
- נעתיק את n האיברים הישנים.
- נוסיף את האיבר החדש במקום $n+1$.

הנחה: זמן הקצאה: $O(1)$.

שוב, נעיר שניתוח של n פעולות יכול להיות שגוי ־ במקרה הגרוע ביותר, *insert* עולה $O(n)$, ולכן n פעולות יעלה $O(n^2)$. שוב, הניתוח הנ"ל שגוי מכיוון שכדי באמת להגיע לפעולה שעולה $O(n)$, צריך לבצע קודם לכן n פעולות שעולות $O(1)$ כל אחת, ולכן ממוצע כל פעולה הוא קבוע.

ננתח בעזרת שיטת הפוטנציאל בלבד.

[2] במדעי המחשב, הפונקציה שסופרת את מספר האחדות בביטוי בינארי נקראת "משקל המינג".

בשיטת הפוטנציאל, ננסה לתת מוטיבציה לבחירת פונקציאת הפוטנציאל. ככל שיש יותר איברים במערך ־ הסיכוי שנצטרך להעתיק אותו למערך חדש הוא גדל. "הפוטנציאל" המצטבר במערכת הוא יותר גדול, ולכן נרצה לשלם על כך מראש (כדי שפירוק הפוטנציאל יהיה "בחינם").

אם כן, נרצה לבדוק עד כמה אנו קרובים לנקודת פירוק הפוטנציאל. לשם כך נשתמש בשני משתנים ־ num ו ־ size. num ישמור את מספר האיברים שיש כרגע במערך. size ישמור את אורך המערך. נשים לב שכאשר $\mathsf{size} = \mathsf{num}$ ־ בהכנסה הבאה נצטרך לפרק את הפוטנציאל. כלומר, כאשר $\mathsf{num} = \mathsf{size} + 1$ נצטרך להגדיל את size פי שתיים. בנוסף, כאשר $\mathsf{size} \approx 2 * \mathsf{num}$ ־ נקבל כי הגדלנו את מבנה הנתונים פי שתיים רק זה מכבר, ונרצה שבנקודה זו ערך הפוטנציאל יהיה 0. ככל שיכניסו יותר איברים ־ ערך הפוטנציאל יעלה, עד אשר נגיע למצב $\mathsf{num} = \mathsf{size}$. כאן נרצה שערך הפוטנציאל יהיה מספיק בשביל לשלם עבור ההרחבה הבאה.

פונקציית פוטנציאל שמתארת את הדרישות שלנו היא:

$$\phi(D_i) = 2 \cdot \mathsf{num} - \mathsf{size}$$

מכיוון שהמערך תמיד מלא בלפחות חצי מהאיברים ($\mathsf{size} \leq 2\mathsf{num}$) נקבל כי לכל i $\phi(D_i)$ חיובי. בנוסף, $\phi(D_0) = 0$. כלומר, פונקציה זו עונה על הדרישה.

כלומר, אם גודל המערך הוא 16, ויש לנו 9 איברים, ערך הפוטנציאל הוא 2. כאשר נגיע ל ־ 16 איברים, ערך הפוטנציאל יהיה כבר 16, ופירוקו ישלם לנו עבור ההעתקה למערך החדש.

כעת נחשב את העלות לשיעורין. נחלק לשני מקרים:

- אם הפעולה ה ־ i לא גורמת להרחבת מבנה הנתונים, נקבל כי $\mathsf{size}_i = \mathsf{size}_{i-1}$, $c_i = 1$ ו $\mathsf{num}_i = \mathsf{num}_{i-1} + 1$. לכן:

$$\begin{aligned}
\hat{c}_i &= c_i + \phi(D_i) - \phi(D_{i-1}) \\
&= 1 + 2 \cdot \mathsf{num}_i - \mathsf{size}_i - (2 \cdot \mathsf{num}_{i-1} - \mathsf{size}_{i-1}) \\
&= 1 + 2 \cdot (\mathsf{num}_{i-1} + 1) - \mathsf{size}_i - 2 \cdot \mathsf{num}_{i-1} - \mathsf{size}_{i-1} \\
&= 1 + 2 + 2 \cdot \mathsf{num}_{i-1} - 2 \cdot \mathsf{num}_{i-1} \\
&= 3
\end{aligned}$$

- אם הפעולה ה ־ i כן גורמת להרחבת מבנה הנתונים, נקבל כי $\mathsf{size}_i = 2 \cdot \mathsf{size}_{i-1}$ (אנו מגדילים את גודל המערך פי שניים). כמו כן, $c_i = \mathsf{num}_i = \mathsf{num}_{i-1} + 1$ ־ העלות של הפעולה ה ־ i היא העתקת num_{i-1} איברים, והכנסת איבר נוסף אחד. מספר האיברים כעת הוא $\mathsf{num}_i = \mathsf{num}_{i-1} + 1$. לבסוף, מכיוון שהמערך הועתק ־ נקבל כי המערך היה מלא. כלומר $\mathsf{num}_{i-1} = \mathsf{size}_{i-1}$. אם כן:

$$\begin{aligned}
\hat{c}_i &= c_i + \phi(D_i) - \phi(D_{i-1}) \\
&= \mathsf{num}_i + 2 \cdot \mathsf{num}_i - \mathsf{size}_i - (2 \cdot \mathsf{num}_{i-1} - \mathsf{size}_{i-1}) \\
&= 3 \cdot \mathsf{num}_i - \mathsf{size}_i - 2 \cdot \mathsf{num}_{i-1} + \mathsf{size}_{i-1} \\
&= 3 \cdot \mathsf{num}_i - 2(\mathsf{num}_i - 1) - 2 \cdot \mathsf{size}_{i-1} + \mathsf{size}_{i-1} \\
&= 3 \cdot \mathsf{num}_i - 2 \cdot \mathsf{num}_i + 2 - \mathsf{size}_{i-1} \\
&= \mathsf{num}_i - \mathsf{size}_{i-1} + 2 \\
&= 1 + 2 = 3
\end{aligned}$$

שכן, מספר האיברים כרגע במערך (num_i) גדול מאחד מגודל המערך לפני ההגדלה.

קיבלנו כי לכל סוג של פעולה $\hat{c} \leq 3$, ולכן, העלות הכוללת של כל סידרת פעולות היא:

$$T(n) = \sum_{i=1}^{n} c_i \leq \sum_{i=1}^{n} \hat{c}_i \leq \sum_{i=1}^{n} 3 = 3n$$

ולכן, העלות הממוצעת לכל פעולה היא $T(n)/n = 3$.

3.7 לקריאה נוספת

- קורמן, לייזרסון, ריבסט: "מבוא לאלגוריתמים", הוצאת MIT. תורגם לעברית ע"י האוניברסיטה הפתוחה.