אוניברסיטת ת"א

סמסטר א' תש"ע

מבני נתונים – תרגיל מס. 2

1. ניתחנו בתרגול את ה-amortized cost של מונה בינארי אינסופי עם פעולת Increment.
   1. נוסיף פעולת Reset שמאפסת את כל הביטים. כיצד ניתן לממש את המונה כך שזמן הריצה ה-amortized לפעולה יהיה O(1) בכל סדרה של n פעולות Increment ו- Reset על מונה שערכו ההתחלתי 0?
   2. הראו שאם מוסיפים למונה בן k סיביות פעולת Decrement (חיסור), העלות לביצוע n פעולות עשויה להגיע עד Θ(nk).

1. א. נמספר את הביטים במערך בסדר עולה מימין (הביט הימני ביותר יקבל אינדקס 1). המשתנה b_max ישמור את האינדקס של הביט השמאלי ביותר שערכו 1.

- באתחול b_max=0.

Increment: אם האינדקס של הביט שהפך מ-0 ל-1 גדול מ- b_max אז ++b_max.

Reset: עבור על כל הביטים מהשמאלי ביותר ועד b_max : 1→0, בנוסף b_max = 0

ניתוח זמן הריצה ה-amortized:

שיטת הבנק – עלות הפיכת 0→1 היא 3.

עלות הפיכת 1→0 היא 0.

עלות הבדיקה האם ביט הוא 1 היא 0.

עלות בדיקת b_max היא 1 (נבדק פעם אחת בדיוק בכל פעולת insert).

מכאן, עלות amortized של insert היא 3 (הפיכת 0→1 + בדיקת b_max), ועלות פעולת Reset היא 1 (עדכון b_max) , משמע עלות amortized לפעולה היא O(1).

ה"יתרה" בבנק תמיד תהיה אי-שלילית מפני שלכל ביט שמודלק שומרים מטבע עבור הפעולה שתאפס אותו, וביט נוסף למעבר עליו במקרה של Reset כאשר הביט מאופס אבל ישנם ביטים דלוקים לשמאלו.

ב. דוגמא לסדרה של n פעולות שעלותן O(nk):

נתחיל ממונה מאופס ונבצע 2^k פעולות increment

עכשיו נריץ פעולות decrement ו-increment לסירוגין.

2. בכיתה ראיתם מימוש של מחסנית ע"י מערך: כאשר רוצים להכניס איבר חדש למערך מלא, מקצים מערך חדש שגודלו כפול מהנוכחי ומעתיקים אליו את המערך הישן.
המקום שנדרש במימוש זה אינו ליניארי במספר האיברים במחסנית (המערך לעולם אינו קטן).
לכן, נשנה את המימוש באופן הבא: אם פחות ממחצית מאיברי המערך מלאים, נקצה מערך חדש שגודלו מחצית מגודל המערך הקודם ונעתיק אליו את המערך הישן.

1. מצא רצף של n פעולות שלוקח $\Theta(n^2)$ זמן במימוש החדש.
2. נשנה את המימוש כך שהמערך החדש יוקצה רק כאשר פחות מרבע מהמערך מלא. גודל המערך החדש שיוקצה הוא עדיין מחצית מגודל מהערך הישן, משמע מחצית מתאי המערך מלאים מיד לאחר ההעתקה.
הוכיחו שבמימוש זה עלות הזמן של כל רצף של n פעולות המתחיל במחסנית ריקה היא $O(n)$.

1. נתחיל ממחסנית ריקה שגודלה המקסימאלי n ונבצע n-1 פעולות push. כעת נבצע 2 פעולות push ו-2 פעולות pop לסירוגין. בכל סדרה כזו של פעולות, פעולת ה-push השנייה תדרוש הקצאת מערך בגודל 2n והכנסת n+1 איברים לתוכו, ופעולת ה-pop השנייה תדרוש הקצאת מערך בגודל n והכנסת n-1 איברים לתוכו.
2. שיטת הבנק: כל הכנסה למערך עולה 3 מטבעות (כמו במימוש בכיתה).
כל מחיקה עולה 2 מטבעות – מטבע אחד משלם על העלות בפועל והשני על העתקה עתידית. מיד לאחר העתקה למערך (בין אם הקטנו או הגדלנו אותו) מחצית מאיברי המערך מלאים. מחיקה "יקרה " (שכוללת העתקה) תקרה אחרי לכל הפחות n/4 מחיקות "זולות", ותצריך העתקה של n/4 איברים, שעליהם נשלם באמצעות היתרה בבנק.

3. מצא סדר הגודל של נוסחאות הרקורסיה הבאות (תנאי הקצה מושמטים).

1. $T(n)=T(n-a)+T(a)+1$ (a קבוע)
הנחה: $T(n) = \Theta(n)$, הוכחה באינדוקציה
2. $T(n)=T(\sqrt{n})+1$

$T(n) = T(n^{1/2})+1 = T(n^{1/4})+2 = \ldots = T(n^{1/2^k}) + k$

$N^{1/2^k} = a \Rightarrow k = \log_2\log_a n \Rightarrow T(n) = T(a) + \log\log n \Rightarrow$

$T(n) = \Theta(\log\log n)$

3. $T(n)=4T(\frac{n}{2})+n^2$
$T(n) = \Theta(n^2\log n)$, מקרה 2 של שיטת המאסטר
4. $T(n)=4T(\frac{n}{2})+n^3$
$T(n) = \Theta(n^3)$, מקרה 3 של שיטת המאסטר

5. $T(n)=T(cn)+T((1-c)n)+1$ (c קבוע, 0 < 1 > c)

הנחה: T(n) = Θ(n), הוכחה באינדוקציה

6. $T(n)=T(9n/10)+n$

$$T(n)=\Theta(n^{\log_{10/9}1})$$ מקרה 3 של שיטת המאסטר

4. נסתכל באלגוריתם המיון הבא:

```
// Sorts the entries I through j in the array A
Sort(A, i, j)
if j = i+1 then
    if A[j] < A[i] then A[j] ↔ A[i]
    return;
end if;
length ← j – i + 1
Sort(A, i, ⌈j – length/3⌉)
Sort(A, ⌊i + length/3⌋, j)
Sort(A, i, ⌈j – length/3⌉)
return;
```

1. הוכיחו את נכונות האלגוריתם. הסימון $\lceil x \rceil$ משמעו ערך עליון של , והסימון $\lfloor x \rfloor$ משמעו ערך תחתון של x.
2. מהי סיבוכיות זמן הריצה האסימפטוטית?

א. טענה: האלגוריתם יחזיר מערך שממוין בסדר לא-יורד בין האינדקסים i ו-j

נוכיח באינדוקציה:

בסיס האינדוקציה: j = i+1 יחזיר [A[i] < A[j

הנחת האינדוקציה: Sort יחזיר מערך ממוין עבור רצף אינדקסים שאורכו קטן מ-j-i+1.

צעד האינדוקציה: נחלק את המערך [A[i..j לשלושה שלישים שווים בגודלם, עד כדי עיגול – a1, a2, a3.

אחרי הקריאה הראשונה ל-Sort כל האיברים ב-a2 גדולים או שווים לאיברים ב-a1.

אחרי הקריאה השנייה ל-Sort כל האיברים ב-a3 גדולים או שווים לאיברים ב-a2 וגם גדולים או שווים לאיברים ב-a1 (יש "מספיק" איברים כי חילקנו ל-3 שלישים שווים), וממוינים בסדר לא-יורד.

אחרי הקריאה השלישית ל-Sort כל האיברים ב-a1+a2 ממוינים ביניהם בסדר לא-יורד וגם קטנים או שווים לאיברים ב-a3.

קיבלנו שהאיברים בכל שליש ממוינים בתוכם, וכן שכל איבר ב-a1 קטן או שווה לאיבר ב-a2 וכל איבר ב-a2 קטן או שווה לאיבר ב-a3, משמע המערך ממוין.

ב. נוסחת הרקורסיה: $T(n) = 3T(2/3n) + 1$, מתאים למקרה 1 של שיטת המאסטר