מספר קורס: 0368.2158

סמסטר א'+ב', מועד ב', תשע"א

18.9.2011

מבחן במבני נתונים

פרופ' חנוך לוי, ד"ר ערן הלפרין, פרופ' אורי צוויק, ירון אורנשטיין ויהב נוסבאום

משך המבחן שלוש שעות

הבחינה עם חומר סגור, ללא מחשבים/מחשבונים. **<u>מותר להביא דף עזר אחד A4 דו-צדדי</u>**.

<u>הוראות כלליות:</u>

1. יש לרשום מספר ת.ז. ומספר מחברת בראש כל דף.
2. המחברות הן לטיוטה בלבד ולא תיבדקנה.
3. הסברים לשאלות האמריקאיות – **חובה**!!!
4. **תשובות סופיות לשאלות האמריקאיות יש לכתוב על הטופס המצורף**.
5. בכל שאלה אמריקאית יש תשובה **<u>אחת נכונה ביותר</u>**. רק תשובה זו תחשב לנכונה.
6. מותר להשתמש באלגוריתמים שנלמדו בכיתה כקופסאות שחורות.
7. בכל מקום בו מוזכרים חסמי זמן הכוונה היא למקרה הגרוע ביותר, אלא אם כן צוין אחרת במפורש.
8. **אין** לכתוב אלגוריתמים בפסאודו-קוד אלא להסביר (באופן משכנע) את הרעיון הכללי.
9. כתבו תשובות קצרות ומדויקות. תשובות לא ממוקדות, ארוכות, מסובכות או מסורבלות יקבלו ניקוד חלקי גם אם הן נכונות.
10. גם אם שאלה נראית לכם דומה לשאלה מבחינות קודמות, היא לא בהכרח זהה.

בהצלחה!!

שאלה 1 (9 נקודות):

נתונה סדרת קלט של N מספרים ממשיים שונים זה מזה. $0.99N$ המספרים הראשונים בסדרה נתונים בסדר אקראי שרירותי (כל סדר אפשרי בהסתברות שווה). $0.01N$ המספרים האחרונים מקיימים:

א) הם כולם קטנים מכל $0.99N$ המספרים הראשונים בסדרה.
ב) הם ממוינים בסדר עולה.

הכניסו את המספרים, על פי סדרם בקלט, בעזרת פעולת insert כפי שנלמדה בכיתה, לתוך עץ חיפוש בינארי (BST) רגיל (לא בהכרח מאוזן).

בחרו את הטענה הנכונה ביותר וספקו נימוק משכנע:

א. תוחלת העלות הכוללת של כל ההכנסות היא $\Theta(N \log N)$
העלות המקסימלית של הכנסה בודדת בסידרה היא $\Theta(N)$.

ב. תוחלת העלות הכוללת של כל ההכנסות היא $\Theta(N^2)$
העלות המקסימלית של הכנסה בודדת בסידרה היא $\Theta(N)$.

ג. תוחלת העלות הכוללת של כל ההכנסות היא $\Theta(N \log N)$
העלות המקסימלית של הכנסה בודדת בסידרה היא $\Theta(\log N)$.

ד. תוחלת העלות הכוללת של כל ההכנסות היא $\Theta(N^2)$
העלות המקסימלית של הכנסה בודדת בסידרה היא $\Theta(\log N)$.

ה. תוחלת העלות הכוללת של כל ההכנסות היא $\Theta(N)$
העלות המקסימלית של הכנסה בודדת בסידרה היא $\Theta(\log N)$.

ו. אף תשובה אינה נכונה.

נימוק קצר: (חובה)

ב

<u>שאלה 2 (9 נקודות):</u>

אנחנו רוצים למזג שני עצים אדומים-שחורים לעץ אדום-שחור אחד. גודלו של העץ האחד הוא m איברים, ושל העץ השני n איברים, כאשר $m > n^2$. שימו לב שטווחי האיברים של שני העצים אינם זרים. באיזה זמן ריצה ניתן לעשות זאת? בחרו את התשובה הטובה ביותר.

א. $\Theta(n)$.
ב. $\Theta(n^2)$.
ג. $\Theta(n+m)$.
ד. $\Theta(nm)$.
ה. $\Theta(n \log m)$.
ו. $\Theta((n+m) \log (n+m))$.

נימוק קצר: (חובה)

ה

שאלה 3 (9 נקודות):

בשאלה הזאת נדון בעץ בעל $n > 100$ צמתים שלשורש יש בדיוק 2 ילדים. נגדיר את חוסר האיזון של העץ כיחס בין מספר הצמתים בתת-העץ הגדול של השורש (זה שיש בו יותר צמתים) למספר הצמתים בתת-העץ האחר של השורש (יחס = המספר הראשון חלקי המספר השני). באיזה מן העצים הבאים נוכל לקבל את חוסר האיזון הגדול ביותר:

א. עץ שמייצג עץ אדום-שחור.

ב. עץ שמייצג עץ 2-4 ולשורש שלו יש שני ילדים.

ג. עץ שמייצג מבנה union-find (עם union by rank ו-path compression) ולשורש שלו יש שני ילדים.

ד. עץ שמייצג מבנה union-find ללא union by rank וללא path compression ולשורש שלו יש שני ילדים.

ה. עץ בערמת פיבונאצ'י שלשורש שלו יש שני ילדים.

ו. תשובות א' ו-ב' נכונות.

ז. תשובות ד' ו-ה' נכונות.

נימוק קצר: (חובה)

ז

<u>שאלה 4 (3 נקודות):</u>

נתון פסאודו קוד של אלג' מיון בשם Min-Max-Sort. האלג' מקבל כקלט מערך A של n מספרים ושני אינדקסים q,p שמקיימים $1 \le p \le q \le n$. האלג' ממיין את תת-מערך A[p..q]. כדי למיין את A כולו נקרא ל- Min-Max-Sort(A, 1, n).

```
Min-Max-Sort(A, p, q)
{
        (r, s) <- Rearrange(A, p, q)
        מסדר את המערך A[p..q], כך שכל ההופעות של המינימום במקומות A[p..r-1], //
       כל הופעות המקסימום  במקומות A[s+1..q] וכל השאר  ב-A[r..s] ומחזיר את r,s //
        If (r ≤ s)
                Min-Max-Sort(A, r, s)
}
```

<u>דוגמא לאלגוריתם Rearrange:</u>

נניח מערך A הוא: 23 20 3 10 2 10 4 5 1 1

וקוראים ל-Rearrange(A, 3, 8), אז תוצאה אפשרית היא: 23 20 10 10 3 4 5 2 1 1

האינדקסים המוחזרים הם (4, 6).

לכל תת מערך A[p..q] הפרוצדורה Rearrange רצה בזמן לינארי בגודל A[p..q].

נניח כי המספרים בתת מערך A[p..q] <u>אינם</u> כולם שווים זה לזה, ויהי k מספר הערכים השונים בתת המערך. מהו מספר הערכים השונים בתת המערך A[r,…,s] לאחר פעולת Rearrange?

א. k

ב. k-1

ג. k-2

ד. k/2

ה. 2

ו. אף תשובה אינה נכונה

נימוק קצר: (חובה)

ג

<u>שאלה 5 (7 נקודות):</u>

השאלה מתייחסת לאלגוריתם מהשאלה הקודמת (שאלה 4). נניח כי במערך A יש k ערכים שונים. מה זמן הריצה של האלגוריתם כפונקציה של n ו-k במקרה הגרוע, כשמריצים Min-Max-Sort(A, 1, n)?.

א. $\theta(n \log (n/k))$
ב. $\theta(n \log k)$
ג. $\theta(n \log n)$
ד. $\theta(nk)$
ה. $\theta(n^2)$
ו. אף תשובה אינה נכונה.

נימוק קצר: (חובה)

ד

שאלה 6 (9 נקודות):

מגדירים אלג' מיון חדש k-Merge-Sort למיון מערך A בגודל n:
1. אם המערך בגודל 1, עוצרים.
2. נחלק את המערך ל-k חלקים שווים בגודלם (לשם פשטות, הניחו ש-n חזקה שלמה של k).
3. נמיין כל חלק באופן רקורסיבי ע"י k-Merge-Sort.
4. נמזג את k החלקים למערך אחד ממוין (המיזוג מתואר להלן).

מיזוג k מערכים ממוינים למערך אחד ממוין:
1. נחזיק מצביע לתחילת כל מערך.
2. נכניס את הערכים עליהם מצביעים לערימת מינימום בינארית, ביחד עם המצביעים, כשהמפתח הוא הערך של המצביע.
3. נבצע n פעמים:
   א. נוציא את המינימום מהערימה (נסמן ערך x ממערך A_i).
   ב. נשים את x במערך הממוין המאוחד כערך הבא במערך.
   ג. נקדם את המצביע במערך A_i ונכניס את הערך עליו הוא מצביע כעת לערימה.
   (אם x הערך האחרון ב-A_i, אז לא מכניסים ערך נוסף לערימה).

בחר בחסם העליון ההדוק ככל שניתן לזמן הריצה של אלג' המיון k-Merge-Sort:

א. $O(n \log n / \log k)$
ב. $O(n \log n \log k)$
ג. $O(n \log k)$
ד. $O(n \log n)$
ה. $O(n (\log^k n))$
ו. אף תשובה אינה נכונה

נימוק קצר: (חובה)

ד

שאלה 7 (5 נקודות):

בשאלה זו נעסוק בצמדים סדורים של מספרים שלמים (x,y). זוג צמדים (a,b) ו- (c,d) נקראים **צמדים מתחלפים** אם a=d ו b=c. לדוגמא: (10, 17) ו (17, 10) הם זוג צמדים מתחלפים.

נתונים בקלט M צמדי מספרים (x, y) שונים. רוצים לכתוב אלגוריתם שימצא אם קיים בין M הצמדים לפחות זוג אחד של צמדים מתחלפים.

מעוניינים באלגוריתם שימצא את התשובה בזמן **worst case** הנמוך ביותר. מה זמן הריצה של האלגוריתם ב **worst case** ?

א. $\Theta(n)$ במקרה הגרוע.
ב. $\Theta(n \log n)$ במקרה הגרוע.
ג. $\Theta(n \log^2 n)$ במקרה הגרוע.
ד. $\Theta(n^2)$ במקרה הגרוע.
ה. $\Theta(n^2 \log n)$ במקרה גרוע.
ו. אף תשובה לא נכונה.

נימוק קצר: (חובה)

ב

<u>שאלה 8 (5 נקודות):</u>

אותם נתונים כמו בשאלה הקודמת (שאלה 7). מעוניינים באלגוריתם שתוחלת זמן הריצה שלו, על כל קלט, הוא הנמוך ביותר. מה תוחלת זמן הריצה של האלגוריתם?

א. $\Theta(\log n)$ בתוחלת.
ב. $\Theta(n)$ בתוחלת.
ג. $\Theta(n \log n)$ בתוחלת.
ד. $\Theta(n \log^2 n)$ בתוחלת.
ה. $\Theta(n^2)$ בתוחלת.
ו. אף תשובה לא נכונה.

נימוק קצר: (חובה)

ב

<u>שאלה 9 (9 נקודות):</u>

השאלה הבאה מתייחסת למבנה נתונים ערימת פיבונאצ'י כפי שנלמדה ונותחה בכיתה. כזכור הפעולות שבו תומך מבנה נתונים זה הן: make-heap (יצירת ערימה ריקה), insert (הכנסת איבר חדש לערימה קיימת), meld (מיזוג שתי ערמות), decrease-key (הקטנת המפתח של איבר נתון), find-min (מציאת האיבר בערימה בעל המפתח המינימלי) ו- delete-min (מחיקת האיבר בעל המפתח המינימלי מהערמה.)

מבצעים n פעולות על אוסף של ערמות פיבונאצ'י. בתחילה האוסף ריק. כל הערמות נוצרות ע"י פעולות make-heap שיוצרות ערמות ריקות. לאחר מכן מכניסים לערמות איברים, ממזגים ביניהן, מקטינים מפתחות, מוחקים איברים וכך הלאה.

בחרו את הטענה הנכונה ביותר וספקו נימוק משכנע:

א. ה-amortized cost של כל פעולת meld הוא $O(\log n)$,
עלולה להיות פעולת meld שעלותה האמיתית היא $\Omega(n)$

ב. ה-amortized cost של כל פעולת meld הוא $O(1)$,
עלולה להיות פעולת meld שעלותה האמיתית היא $\Omega(n)$

ג. ה-amortized cost של כל פעולת meld הוא $O(1)$,
עלולות להיות פעולת meld שעלותה האמיתית היא $\Omega(\log n)$

ד. ה-amortized cost של כל פעולת meld הוא $O(1)$,
העלות האמיתית של כל פעולת meld אף היא $O(1)$

ה. ה-amortized cost של כל פעולת meld הוא $O(\log n)$,
העלות האמיתית של כל פעולת meld אף היא $O(\log n)$

ו. ה-amortized cost של כל פעולת meld הוא $O(n)$,
עלולה להיות פעולת meld שעלותה האמיתית היא $\Omega(n)$

נימוק קצר: (חובה)

ד

שאלה 10 (6 נקודות)

נתונה ערימת מינימום H ובה n נתונים שונים זה מזה. מעוניינים לקחת את $\sqrt{n}$ האיברים הקטנים ביותר ולמיינם בזמן Worst case הנמוך ביותר.

עלות בצוע הפעולה (על ידי האלגוריתם הטוב ביותר שהנכם מכירים) על H היא:

א. $O(\sqrt{n})$.

ב. $O(\sqrt{n}\log n)$.

ג. $O(n)$.

ד. $O(n\log n)$.

ה. $O(n\sqrt{n})$

ו. $O(n\sqrt{n}\log n)$

נימוק קצר: (חובה)

ב

<u>שאלה 11 (12 נקודות):</u>

האלגוריתם "select" הדטרמיניסטי, אותו למדנו הכיתה, פועל באופן הבא:
a. המערך מחולק לקבוצות בגודל 5.
b. ממיינים כל קבוצה.
c. בוחרים pivot בתור החציון של החציונים (באופן רקורסיבי).
d. מבצעים "Partition" עם ה-pivot שנבחר.
e. מפעילים את האלגוריתם רקורסיבית על חלק המערך הרלוונטי.

<u>הניחו **שבמקום קבוצות בגודל 5** אנו לוקחים **קבוצות בגודל 3.**</u>
בשאלה זאת נעסוק בזמן הריצה של **<u>האלגוריתם החדש (עם קבוצות בגודל 3)</u>**

1. (2 נק) מהי עלות צעד b?

עלות: O(N)

הסבר: מיון כל שלשה קבוע, ויש N/3 שלשות

2. (2 נק') נסמן ב T(N) זמן ריצת האלגוריתם על מערך בגודל N. מהי עלות צעד c, מבוטאת בעזרת T().

עלות: T(N/3)

הסבר: מחפשים חציון בין N/3 מספרים

3. (8 נק) מהי העלות הכוללת של האלגוריתם? הראו את נוסחת הנסיגה לזמן הריצה ופתרו אותה.

עלות: O(N log N)

הסבר: נוסחת נסיגה $T(N) \le T(N/3) + T(2N/3) + O(N)$.
כאשר בוחרים חציון החציונים, יש לפחות 1/3 קטנים ממנו ו-1/3 גדולים ממנו לפי 2 מכל שלשה שגדול מהחציון שלה כפול חצי מהשלשות כפול שליש. ולכן ממשיכים רקורסיבית על לכל היותר 2/3 מהמערך. שאר הפעולות לינאריות (partition ומיון), ועלות של מציאת חציון החציונים מסעיף ב'.
אפשר לפתור ע"י אינדוקציה (ויש עוד דרכים).

שאלה 12 (17 נקודות):

נתון מערך בגודל n, המחולק ל-n/k קטעים באורך k כל אחד. ידוע שעבור כל שניים מבין k הקטעים, או שכל האיברים בקטע הראשון קטנים מכל האיברים בקטע השני, או שכל האיברים בקטע הראשון גדולים מכל האיברים בקטע השני.

נתעניין באלגוריתם מיון למערכים מסוג זה, המתאים למודל ההשוואות, וכן בחסם תחתון עבור אלגוריתמי מיון במודל ההשוואות למערכים מסוג זה.

א. (5 נקודות) הציעו אלגוריתם מיון יעיל ככל האפשר למערך מסוג זה ונתחו את זמן הריצה שלו (כפונקציה של n ו-k).

תשובה: ניתן למיין כל קטע בנפרד, בזמן כולל של $(n/k)k \log k = n \log k$. לאחר מכן, יש פשוט למיין את n/k הקטעים בזמן $(n/k)\log(n/k)$. התשובה הסופית היא
$n \log k + (n/k)\log(n/k)$

ב. (6 נקודות) הציעו חסם תחתון טוב ככל האפשר (כפונקציה של n ו-k) והוכיחו אותו.

מספר הסידורים האפשריים הוא $(n/k)!\,(k!)^{n/k}$.
כשלוקחים לזה log מקבלים $n \log k + (n/k)\log(n/k)$

ג. (6 נקודות) הציעו אלגוריתם יעיל יותר, אם ידוע שמספר הערכים השונים בכל קטע הוא לכל היותר log k. מה זמן הריצה של האלגוריתם?

מיון כל קטע יכול עכשיו להתבצע בזמן k loglog k

סוף