

מבני נתונים ואלגוריתמים - הרצאה 8

22 בנובמבר 2011

המשך מיוני השוואה

המיונים שראינו בהרצאה הקודמת:

- Bubble-Sort ו־Cocktail Sort
- Selection Sort
- Insertion Sort
- Quick Sort
- Heap Sort:

בערימה, כל פעולה היא $O(\log n)$ לכן גם במקרים הכי גרועים עלות המיון תהיה $O(n \log n)$.

אלגוריתם Merge-Sort 1

1. אם גודל המערך הוא 1, החזר את הערך.
 2. אחרת, הפעל Merge-Sort על חצי שמאלי וחצי ימני של מערך.
 3. אחד שני חצאים ממוינים.*
- *: איחוד שני מערכים A, B :
1. החזק שני משתנים i, j עבור מיקום A ומיקום B .
 2. אתחל $i = 0, j = 0$.
 3. אם i גדול מגודל A , החזר את הערך הבא ב B ובצע $j++$.
 4. אם j גדול מגודל B , החזר את הערך הבא ב A ובצע $i++$.
 5. אם i גדול מגודל A ו j גדול מגודל B - סיים.
 6. אחרת, השווה $A(i)$ ו $B(j)$, החזר את הקטן מביניהם וקדם את האינדקס המתאים.
 7. אם $A(i) = B(j)$ החזר $A(i)$ וקדם את i באחד.
 8. חזר ל3.
-

Non-Comparative Sorts

מיונים שאינם מיוני השוואה, שבהם אנו משתמשים בערכים שאנו ממיינים.

אלגוריתם Counting Sort 2

נניח שהנתונים הם בין 1 ל k .

1. ספור כמה פעמים מופיע כל נתון
 2. עבור מ 1 עד k וכתוב אותו כמספר הפעמים שהוא מופיע.
-

עלות האלגוריתם הזה היא $O(n + k)$.

אלגוריתם 3 Bucket Sort

נניח נתונים בין x ל y .

1. חלק את התחום בין x ל y ל k דליים.

2. הכנס כל ערך לדלי מתאים.

3. הוצא את הערכים לפי סדר הדליים (וסדר ההכנסה לדלי).

4. בצע Insertion Sort

זהו אלגוריתם יציב. אם k מספיק גדול, אז העלות היא בערך $O(n)$. אבל יש בעיה - יכול להיות שיש דליים כבדים, כלומר דליים שיש בהם כמעט את כל האיברים. אם דלי מכיל יותר מדי נתונים, בצע Bucket Sort על הדלי הספציפי הזה.

אלגוריתם 4 Radix Sort - Least Significant Digit

סידור לפי ערך

1. השלם את כל הערכים לאורך המקסימלי ע"י הוספת אפסים משמאל.

2. התחל מהאות האחרונה לראשונה וסדר כל אות לפי הערכים, תוך שמירה על סדר קיים.

עלות האלגוריתם - $O(n \cdot \text{num-of-digits})$

אלגוריתם 5 Radix Sort - Most Significant Digit

סידור בסדר מילוני

1. התעלם מאפסים משמאל למספר.

2. סדר כל ערך לפי ספרה שמאלית ביותר.

3. סדר ברקורסיה כל תת קבוצה עם אותו ערך בספרה הנוכחית כאשר חוסר ערך קודם לו.
