

מבני נתונים בסיסיים – תרגילים

שאלה 1

כתבו אלגוריתם המקבל מיקום של פרש על לוח שחמט שגודלו $n \times n$. ומחזיר בכל תא במערך את אורך המסלול הקצר ביותר של הפרש אליו.

שאלה 2

רשימה מקושרת יכולה להסתיים במצביע null (ואז נקרא לה רשימה קווית) או במצביע לאיבר קודם ברשימה (ואז נקרא לה שבלול).
כתבו אלגוריתם המקבל מצביע לרשימה מקושרת (המכילה לפחות איבר אחד) ומחזיר true אם ורק אם הרשימה היא שבלול. זמן הריצה הוא לינארי באורך הרשימה.

שאלה 3

תארו כיצד לממש "מחסנית מינימום" – מחסנית שחוץ מפעולות POP ו-PUSH, ממומשת עליה גם פעולת MIN המחזירה את הערך המינימלי במחסנית (רק מחזירה ולא מוחקת).
זמני הריצה של שלוש הפעולות יהיו קבועים.

שאלה 4

כתבו אלגוריתם המקבל תור והופך אותו. כלומר: האיבר הראשון יהפוך לאחרון, האיבר האחרון יהפוך לראשון וכו'.
מספר האיברים בתור, $n > 0$, ידוע.
האלגוריתם ישתמש ב- $O(1)$ זיכרון נוסף, וזמן הריצה שלו יהיה $O(n^2)$.

שאלה 5

נתונים שני מספרים עשרוניים גדולים מאוד x, y .
כל אחד מהמספרים נמצא במחסנית, כך שהספרה המשמעותית ביותר נמצאת בתחתית המחסנית.
כתבו אלגוריתם המקבל שתי מחסניות $S1, S2$ שבהן מאוחסנים המספרים x ו- y ומחזיר מחסנית $S3$ שבה נמצא הסכום $x + y$, (גם ב- $S3$ הספרה המשמעותית ביותר תהיה בתחתית המחסנית)
מותר להשתמש אך ורק בפעולות הבסיסיות המוגדרות על מחסנית.
בתום ריצת האלגוריתם $S1, S2$ תהיינה ריקות.

שאלה 6

The operations on a "minimum stack" are:

- $\text{push}(x)$ – insert x .
- $\text{pop}()$ – delete the last item that was pushed.
- $\text{min}()$ – return (but do not delete) the item with the minimal key.

Implement a "minimum stack" using two stacks ("ordinary" stacks):

write an algorithm for each operation (pseudo code).

Explain your implementation.

The running time of every operation is $O(1)$.

שאלה 7

Describe a way to implement a queue, using two stacks. Explain how to implement Enqueue and Dequeue. No need to write code in your answer, just describe the algorithm.

The amortized running time of Enqueue and Dequeue should be $O(1)$.

שאלה 8

Whereas a stack allows insertion and deletion of elements at only one end, and a queue allows insertion at one end and deletion at the other end, a deque (double-ended queue) allows insertion and deletion at both ends. Write four $O(1)$ -time procedures, in pseudocode, to insert elements into and delete elements from both ends of a deque constructed from an array.