Data Structures
Homework #3

Due: November 5, 2024

1. Let x and y be real number with $0 < x < y$. Prove that n^x is $O(n^y)$, but n^y is **NOT** $O(n^x)$.

2. Describe, step by step, the output for the following sequence of deque ADT operations: $\text{addFirst}(3)$, $\text{addLast}(8)$, $\text{addLast}(9)$, $\text{addFirst}(5)$, $\text{removeFirst}()$, $\text{removeLast}()$, $\text{first}()$, $\text{addLast}(7)$, $\text{removeFirst}()$, $\text{last}()$, $\text{removeLast}()$.

3. Suppose that you are given an $m \times n$ matrix A. Now you are asked to check if matrix A has an entry $A[i][j]$, which is the smallest value in row i and the largest value in column j. For example, the following matrix $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$ has such an entry at $A[3][1]$ with value 7 and the matrix $\begin{pmatrix} 1 & 2 & 3 \\ 7 & 8 & 9 \\ 11 & 10 & 5 \end{pmatrix}$ has no such an entry. Please have a solution to determine the location of such an entry if one exists in the given matrix. What is the running time of your method.

4. Suppose we are given an integer N and would like to have all the binary numbers smaller than or equal to N. In order to have the results generated, one can use a ***queue*** to help. Please give an approach that uses a ***queue*** to generate the results, write your approach with pseudo-code, and discuss the time as well as space complexity for your approach. Please note that the output should have the binary numbers in increasing order.

   ```
   Input: 10
   Output: ['1', '10', '11', '100', '101', '110', '111',
            '1000', '1001', '1010']

   Input: 13
   Output: ['1', '10', '11', '100', '101', '110', '111',
            '1000', '1001', '1010', '1011', '1100', '1101']
   ```

5. (**Programming problem**)
   Recall Problem 4. Please use Python to implement the pseudo-code you provide for solving the problem with the approach, named as `generateBinaryNumbers()`. Please note that your program *must use a **queue*** as an auxiliary tool. When you submit your homework, you need to indicate where the queue is in your program and how it helps. Hence, you need to

   (1) implement the approach as a function named as `generateBinaryNumbers()`,

(2) implement a **queue** data structure to manage the items used for the approach, and

(3) give a short statement for indicating where and how the queue is used in your program and discussing the time and space complexity.

About submitting this homework

1. For problem 1, 2, 3 and 4, Please

   (1) write all of your solutions on the **papers of size A4**,

   (2) leave you name and student ID on the first page, and

   (3) hand in your solutions for problem 1, 2, 3 and 4 to me in class

2. For problem 5, things to be submitted include:

   (1) please finish each problem right after the problem description in the `HW3.ipynb` file provided on the **i-school(Plus)** (`https://istudy.ntut.edu.tw/learn/index.php`) platform; and

   (2) please upload the completed `.ipynb` file with the filename as `HW3_studentID.ipynb` to **i-school(Plus)**

3. **Late work** is not acceptable. Remember, the **deadline** is the midnight of **November 5**, 2024.

4. Honest Policy: We encourage students to discuss their work with the peer. However, each student should write the program or the problem solutions on her/his own. Those who copy others work will get 0 on the homework grade.