Data Structures
Homework #2

Due: October 28, 2025 (Before class)

1. Recall the ***selection sort*** we discussed in class to illustrate how to have the pseudo-code. The illustrated pseudo-code for selection sort is iterative as below.

```
ITERATIVE_SELECTION_SORT(A)
    Input: An input array A of size n
    Output: A sorted array // Sort the array A[1 : n] in a nondecreasing order.
1   for i = 1 to n
2       j = i
3       for k = (i + 1) to n
4           if (A[k] < A[j])
5               j = k
6       t = A[i]
7       A[i] = A[j]
8       A[j] = t
9   return A
```

   (a) Count the total number of primitive operations used in ITERATIVE_SELECTION_SORT(A).

   (b) With the same idea, please provide a recursive version for the selection sort, RECURSIVE_SELECTION_SORT(A), with pseudo-code.

   (c) Please try to derive the total number of primitive operations used in the recursive version you provided in 1b.

2. Show that for any real constants a and b, where $b > 0$, $(n + a)^b = \Theta(n^b)$.

3. Consider the Fibonacci function $F(n)$, $F(1) = 1, F(2) = 2, F(3) = 3, F(4) = 5, \cdots, F(n) = F(n-1) + F(n-2)$. Show by induction that $F(n)$ is $\Omega((3/2)^n)$.

4. Let $f(n)$ be an asymptotically positive functions. Disprove with counterexamples or prove each of the following conjectures.

   (a) $f(n) = O((f(n))^2)$.

   (b) $f(n) = \Theta(f(n/2))$.

5. (**Programming 1 (Lucky Star)**)
   A group of people lines up in a circle and a bill is passed one by one clockwise. There is a counter starting from 1 to count the number of passes. If the counted number equals to a given magic number (integer), then the person having the bill will leave the circle and hand the bill to the next person. The counter resets to 1 and the process then continues until only one person is left. The last left person is the Lucky Star and can get the bill. Please write a program named `Finding_LuckyStar()` for this with a queue. The input will be a list of different names and a given magic number. The output will be the name

of the Lucky Star. A running example:

```
>>> print(Lucky_Star(["Bill","David","Susan","Jane","Kent","Brad"],7))
Susan
```

6. (**Programming 2**)

   A ***binary code*** is the value of a data-encoding convention represented in a binary notation with a sequence of 0s and 1s, called a *bit string*. We consider the positive integers includiong 0. One-bit binary codes "0" and "1" represent zero(0) and one(1) respectively. We use $B_1 = [0, 1]$ to denote them. For two-bit binary codes, there are "00", "01", "10", and "11" representing 0, 1, 2, and 3, repectively, as well as we use $B_2 = [00, 01, 10, 11]$ to denode. In fact, they are the binary representations for decimal integers. So, three-bit binary codes B_3 are listed in Table 1.

   A ***Gray code*** is an encoding of numbers so that two contiguous numbers have a single digit differing by 1. The term Gray code is often used to refer to a "reflected" code, or more specifically still, the binary reflected Gray code. For example, one-bit Gray code is $G_1 = [0, 1]$ and two-bit Gray code is $G_2 = [00, 01, 11, 10]$. Three-bit Gray code is in Table 2.

Table 1: 3-bit binary codes, B_3

Dec	Binary code
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Table 2: 3-bit Gray codes, G_3

Dec	Gray code
0	000
1	001
2	011
3	010
4	110
5	111
6	101
7	100

   This exercise asks you to list all the n-bit binary codes and Gray code using a ***queue*** and a ***stack*** respectively. Your Python program should read a value n, the number of bits of the bit-strings, and generate the n-bit binary and Gray codes as well as list them in order.

   Please note that your program must use a queue and stack you implement as auxiliary tools. **Please do not import any exisitng Python packages in your code**. There will be no grade if Python packages are imported in the code. When you submit your homework, you need to indicate where the queue and stack are in your program and how they help using a text cell in the notebook.

About submitting this homework

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

   (1) **write by hand** 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 before class on the due date.

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 **October 28**, 2025.

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.