CIS 55
Exercise 4, Due no later than 22-Oct-2021, 11:59pm Pacific Time

Below is your exercise for Stacks, Queues, and Priority Queues. For submitting your answers, you can either send me an updated Word document with your answers, email your answers directly, or even take a screen shot of your work on paper if you like. In all cases, email me your answers when you feel ready.

True or False (9 points)
For each statement below, indicate whether you think it is True or False. If you like, you can provide a description of your answer for partial credit in case you are incorrect.

1) Given a stack implemented as an array, and the following functions performed:

   Push(A), Push (B), Push(C), Pop(), Push(D), Push(E), Pop(), Push(F)

   a. The array will be [A][B][D][F] and the "top" of the stack will be at index 3 where [F] is
   b. The next element that will be removed when Pop is called will be A
   c. If we Push(G), [G] will be inserted at index 0

2) Given a queue implemented as an array with a maximum capacity of 5 elements and O(1) insert and remove, and the following functions are performed:

   Insert(7), Insert(6), Insert(5), Insert(4), Remove(), Remove(), Insert(3), Insert(2), Insert(1), Remove()

   a. The removeIndex value will be index 4
   b. The array will be [2][1][ ][4][3]
   c. The insertIndex value will be index 2

3) Given a priority queue implemented as a standard linked list with only a head pointer, insert performing at O(n) and remove at O(1), and the following functions performed (higher numbers represent higher priorities):

   Insert("Rick", 75), Insert("Carol", 80), Insert("Daryl", 65), Remove(), Insert("Negan", 60), Insert("Maggie", 25), Remove(), Insert("Michonne", 55)

   a. If Insert("Eugene", 40) is called, it will be added as the last node in the linked list
   b. The linked list from head to the end at this point is Rick, Negan, Michonne, and then Maggie
   c. Carol was the first element removed from the priority queue
   d. None of the above

(Short answer section on next page)

Short answer (4 points):

Answer the following in your own words:

1) Your friend says they implemented a stack as a linked list with reference pointers to both a head and the tail node (i.e. they implemented a stack as a double ended linked list). They said this significantly improved the performance for all functions (push, pop, search, and update). Are they correct? Why or why not?
2) Another friend implemented a queue as a linked list with only a head pointer (no tail node). She said this made performance of all functions (insert, remove, search, and update) categorically equal to a queue as a linked list with a head and a tail node. Is she correct? Why or why not?