CSIT 503 HW3
Topic: Stack & Queue & Linked List

Dr. Boxiang Dong

1 Problem Description

Instructions. You are provided four skeleton program named *Stack.java, Queue.java, ListNode.java and LinkedList.java.* The source files are available on Canvas in a folder named *HW3*. Please modify the skeleton code to solve the following tasks.

- Task 1 (30 pts). Implement the *empty(), push(int x), and pop()* function for Stack in Stack.java as discussed in Lecture 5.
- Task 2 (30 pts). Implement the *enqueue(int x), dequeue()* function for Queue in Queue.java as discussed in Lecture 5.
- Task 3 (40 pts). Implement the *search(int k), insert(int x), delete()* function in LinkedList.java as discussed in Lecture 6.
- Task 4 (10 pts Extra Credit). In the *push(int x)* function of Stack.java, by default, we never check if the stack is already full. If we insert an element into a full stack, we should get an error. Implement the capacity check feature for *push(int x)*. (Hint: Use System.err.println() to print the error message.)
- Task 5 (10 pts Extra Credit). In the *enqueue(int x)* function of Queue.java, we do not check if the queue is already full. Implement the capacity check feature for *enqueue(int x)*. (Hint: Use System.err.println() to print the error message.)
- Task 6 (10 pts Extra Credit). In the *dequeue()* function of Queue.java, we do not check if the queue is empty. If we dequeue an empty queue, we should also get an error. Implement the empty check feature for *enqueue(int x)*. (Hint: Use System.err.println() to print the error message.)

2 Submission Guideline

1. Work individually.

2. Please directly insert your code in the appropriate place in Stack.java, Queue,java, and LinkedList.java.

3. Create a zip file of your .java source programs and submit it on Canvas. A late penalty of 10 points for each late day applies. Any late for more than three days receives zero automatically.