Exercise Sheet 8

Handout: Oct 28th — Deadline: Nov 4th 4pm

Question 8.1 (0.25 marks)

Draw the following data structures after each of the following operations. Assume that the data structures are initially empty. You don't need to draw pointers for stacks and queues.

1. Consider a stack S and the operations PUSH$(S, 7)$, PUSH$(S, 4)$, PUSH$(S, 5)$, POP(S), PUSH$(S, 8)$, POP(S), POP(S).
2. Consider a queue Q and the operations ENQUEUE$(Q, 7)$, ENQUEUE$(Q, 4)$, ENQUEUE$(Q, 5)$, DEQUEUE(Q), ENQUEUE$(Q, 8)$, DEQUEUE(Q), DEQUEUE(Q).
3. Consider a singly-linked list L and the operations LIST-PREPEND$(L, 7)$, LIST-PREPEND$(L, 4)$, LIST-PREPEND$(L, 5)$, LIST-DELETE$(L, 4)$, LIST-PREPEND$(L, 8)$, LIST-DELETE$(L, 7)$, LIST-DELETE$(L, 8)$.

Question 8.2 (0.5 marks) Explain how to implement two stacks S_1 and S_2 in one array $A[1 : n]$ in such a way that neither stack overflows unless all the n elements of A are full. Present the pseudocodes for operations PUSH$S_1(A, x)$,PUSH$S_2(A, x)$,POP$S_1(A)$, and POP$S_2(A)$.

Question 8.3 (0.25 marks) Rewrite ENQUEUE and DEQUEUE to detect underflow and overflow of a queue.

Question 8.4 (0.5 marks) Show how to implement a Queue using 2 stacks S_1 and S_2. Provide the pseudo-code of the operations ENQUEUE and DEQUEUE. You don't need to check for underflow and overflow. Analyse the runtime of the two operations.

Question 8.5 (0.5 marks) Implement an Integer Calculator that takes a postfix expression in input using integers as operands and $\{+, -, *\}$ as operators. The algorithm should use a stack. (See Judge)

Question 8.6 (1 mark) Implement the problems "Finding Adjacent Value" and "Jet Bridge Allocation" on the Judge system.