Data Structures   Name -
Ch. 4 Practice Test

Label and completely answer the following on lined paper. You may use the "Quick Reference for apstack & apqueue classes" (i.e. apstack.h and apqueue.h) on this test.

1. Explain any advantages of using a linked list implementation of a queue ADT over a vector implementation?

2. Convert the following infix expression into postfix. Do this CAREFULLY and EXACTLY. Even though it was Exercise #1 on p. 253, practically no one did it perfectly when it was assigned as homework.

    A + B * (C - D) / (P - R)

3. Convert the following infix expression into prefix. Do this CAREFULLY and EXACTLY. Even though it was Exercise #1 on p. 253, practically no one did it perfectly when it was assigned as homework.

    A + B * (C - D) / (P - R)

4. Write a non-member, non-friend, void function named **DisplayLastThree** that displays the top three element of an apstack of integers in their original order. The three integers should be displayed on separate lines. For example if the apstack is 1, 34, 23, 11, 6, 8 with 8 being the top of the stack (last in) then the output should be 11, 6, and 8 on separate lines.

5. Determine the output of the following code segment:

    ```
    apstack <int> Q;
    int x = 5;
    int y = 11;
    Q.push(x);
    Q.push(y);
    Q.push(y + 16);
    Q.pop(x);
    Q.push(x);
    y = Q.length();
    for (x = 0; x < y; x++)
    {
       Q.pop(y);
       cout << y << endl;
    ```

}

6. Redo #1 on p. 272. This graded homework has been returned to you. No one did this assigned homework exercise perfectly.

7. Write a definition (i.e. function implementation) for the apstack member function

   void push(const itemType & item);

   using either the vector or linked list implementation of apstack. Be sure to specify though which implementation of apstack you are assuming.