

```

10 import java.util.LinkedList;
6
7 public class Infix_Postfix {
8
9     public static void main(String[] args) {
10         // TODO Auto-generated method stub
11         Scanner sc = new Scanner(System.in);
12         Stack st = new Stack(); // FILO
13         String infix = sc.nextLine(); // รับสมการ
14         String postfix = "";
15         StringTokenizer token = new StringTokenizer(infix, " "); // แยกด้วย " "
16         Queue<String> q = new LinkedList<>(); // FIFO
17
18         while (token.hasMoreTokens()) { // แยกจนกว่าจะหมด
19             String temp = token.nextToken();
20             String txt;
21             if (temp.equals("(")) {
22                 st.push(temp);
23             } else if (temp.equals("+") || temp.equals("-")) {
24                 if (st.empty() || st.peek().equals("("))
25                     st.push(temp);
26                 else {
27                     q.add((String) st.pop());
28                     st.push(temp);
29                 }
30             } else if (temp.equals("*") || temp.equals("/")) {
31                 st.push(temp);
32             } else if (temp.equals(")")) {
33                 while (!st.peek().equals("(")) {
34                     q.add((String) st.pop());
35                 }
36                 st.pop();
37             } else
38                 q.add(temp);
39         } // while
40         while (!st.empty()) {
41             q.add((String) st.pop());
42         }
43         System.out.println(q);
44         System.out.println(computePost_Fix(q));
45     } // main
46

```

```

47- static double computePost_Fix(Queue<String> q) {
48     Stack st = new Stack();
49     double x, y;
50     while (q.size() != 0) {
51         String token = q.poll();
52         if (token.equals("+")) {
53             y = (double) st.pop();
54             x = (double) st.pop();
55             st.push(x + y);
56         } else if (token.equals("-")) {
57             y = (double) st.pop();
58             x = (double) st.pop();
59             st.push(x - y);
60         } else if (token.equals("*")) {
61             y = (double) st.pop();
62             x = (double) st.pop();
63             st.push(x * y);
64         } else if (token.equals("/")) {
65             y = (double) st.pop();
66             x = (double) st.pop();
67             st.push(x / y);
68         } else
69             st.push(Double.parseDouble(token));
70     }
71     return (double) st.pop();
72 }
73 }

```