Expression Parser

Algorithm - parse

1. ***Set*** postfix queue to empty.
2. ***Set*** operation stack to empty.
3. ***For each*** token in the infix queue.
4. ***If*** the token is an operand ***then***,
5. Append the token to the postfix queue.
6. ***Else if*** the token is a function ***then***,
7. Push the token on to the operation stack.
8. ***Else if*** the token is an argument separator ***then***,
9. ***While*** the top meta-operation on the operation stack is not a left parenthesis ***do***,
10. Pop the operation from the operation stack.
11. Append that operation to the postfix queue.
12. ***Else if*** the token is an left parenthesis ***then***,
13. Push the token on to the operation stack.
14. ***Else if*** the token is a right parenthesis ***then***,
15. ***While*** the top meta-operation on the operation stack is not a left parenthesis ***do***,
16. Pop the operation from the operation stack.
17. Append that operation to the postfix queue.
18. ***If*** the operation stack is empty ***then***,
19. ***Exception*** "Right parenthesis, has no matching left parenthesis"
20. Pop the left parenthesis from the operation stack.
21. ***If*** the top of the operation stack is a function ***then***,
22. Pop the function from the operation stack
23. Append that function to the postfix queue.
24. ***Else if*** the token is an operator ***then***,
25. ***While*** the operation stack is not empty ***do***
26. ***If*** the top of the operation stack is not an operator ***then***,
27. ***Exit*** the while loop.
28. ***If*** the token is a non-associative operator ***then***,
29. ***Exit*** the while loop.
30. ***If*** the token is a left-associative operator ***and***
has greater precedence than the top of the operation stack ***then***,
31. ***Exit*** the while loop.
32. ***If*** the token is a right-associative operator ***and***
has greater or equal precedence than the top of the operation stack ***then***,
33. ***Exit*** the while loop.

34. Pop an **operator** from the top of the **operation stack**.
35. Append that **operator** to the **postfix queue**.
36. ***End while***
37. Push the **token** on to the **operation stack**.
38. ***Else***
39. ***Exception*** "**Unknown token**".
40. ***End for-each***
41. ***While*** the **operation stack** is not empty ***do***
42. ***If*** the top of the **operation stack** is a **left-parenthesis** ***then***,
43. ***Exception*** "**Missing right-parenthesis**".
44. Pop an **operator** from the top of the **operation stack**.
45. Append that **operator** to the **postfix queue**.
46. ***Return*** **postfix queue**