CS2002301 & EC2002302 Data Structures

Homework #1

Due date: 10/19/2020 10:20

TA email: B10630040@gapps.ntust.edu.tw

Announcements

1. This assignment is weighed 10 points.
2. Submit your code to the OJ system (http://nlp.csie.ntust.edu.tw:2020/).
3. Additionally, submit a report and source code to the Moodle system.

1. (5 pts) Delete Strictly Increasing Sequences

Suppose a non-negative integer sequence S and a positive integer T, perform the following operations:

1. Scanning sequence S from left to right, find the first strictly increasing sequence S' with length T.
2. Remove S' in S.

You should continue doing these operations until you can't find any S' in S.

For example, Suppose that $S = [4, 5, 1, 2, 3, 6, 7]$, $T = 3$:

(1) The first strictly increasing sequence with length = 3 is [1, 2, 3].

(2) Remove [1, 2, 3] in S. Then, $S = [4, 5, 6, 7]$.

(3) We find another S': [4, 5, 6].

(4) Remove [4, 5, 6] in S. Then, $S = [7]$.

(5) Now, we can't find any S' in S. The program terminates here.

Sample I/O and input constraints are showed on OJ system.

* In this problem, only 60% of the test cases will be showed on OJ system, while the remaining 40% are hidden. Note that we will use your **latest submission** on OJ system to test the hidden cases, so make sure your latest submission is the best version.

* In the report, briefly explain the way you solve the problem.

* Do not use any libraries (string, vector...) except for standard I/O.

2. (5 pts) Calculator

Please implement a calculator to calculate mathematical expressions. Your program should be able to handle the following operators: `+, -, *` and parentheses. However, the input may not be a valid expression; you have to make sure that your program can handle invalid expressions.

Precedence	Operators
1	()
2	*
3	+ −

Table 1: The precedence of operators used in this problem.

If the expression is a valid expression, please output (1) its postfix expression and (2) its calculation result; otherwise, output "invalid".

For example, `"(93+89)*2"` is a valid expression. In this case, please output its postfix expression `"93 89 + 2 *"` at the first line, and then output the calculation result `"364"` at the second line.

For invalid prefix expressions, such as `"2+2***",` please output "invalid".

Note that every number in the expression is a non-negative integer. You don't have to care about integer overflow (using `int` to calculate the result is enough). That is, every valid expression is calculable.

Sample I/O and input constraints are showed on OJ system.

* In the report, briefly explain the way you solve the problem.

* Do not use any libraries (string, vector...) except for standard I/O.