Indian Institute of Technology Mandi
August-December 2023 Semester

CS515: Advanced Computer Science Practicum

Lab3 - Programming Assignment Problem Statements

Implement the following problems using C++ programing language.
Note:

1. Write a separate main program to evaluate the functions in linked list, stack and queue data structures.
2. Implement the linked list data structure using the list.hpp
3. Implement stack and queue data structures using stack.hpp and queue.hpp. Most important is, ***use the abstraction of linked lists to implement the functionality of stack and queue***.
4. Use Stack or Queue data structure as per the need of the problem given to you.
5. Write a separate main program to evaluate each of the 5 problems.

Problems:

1. Design and implement a calculator that can calculate prefix expressions like:
   + 2 4
   * 8 (+ 7 12)
   (+ 7 (* 8 12) (* 2 (+ 9 4) 7) 3)
   Note: All items are space delimited. Use stack data structure.
   Example answers:
   + 2 4 => 2 + 4 = 6
   * 8 (+ 7 12) => 8 * (7 + 12) = 152
   (+ 7 (* 8 12) (* 2 (+ 9 4) 7) 3) => 7+8*12+2*(9+4)*7+3 = 148

2. Given a mathematical expression, count the number of brackets (pair of open and closes) and then remove the redundant brackets from the expression.
   Example:
   Input: (a + (b*c)) * (d * (f * j))
   Output should be: (a + b * c) *d * f * j
   Note: Use stack as data structure.

3. Nikhil learnt two new commands **pwd** and **cd** on the first day of Operating Systems lab.
   pwd - command displays the current working directory and,
   cd - changes the location of working directory.

   If the **cd** parameter contains **".."** (without quotes), that means to step one directory back.
   The absolute path of directory is separated by slashes **"/"**(without quotes).
   The default root directory is **"/"**.
   Your task is to print the current working directory.

   Input

   Input description.
   - The first line of input contains **T**, denoting the number of test cases.
   - The second line of input contains **N**, denoting the number of commands.
   - Then **N** lines follow, each containing either a **cd** command or **pwd** command.

   Output

   Output description.
   - For each **pwd** command, output the absolute path of current directory.

Constraints

Should contain all the constraints on the input data that you may have. Format it like:

- **1 ≤ T ≤ 100**
- **1 ≤ N ≤ 100**
- Maximum length of cd parameter ≤ 200

Example

Input:

1
9
pwd
cd /home/csed
pwd
cd /lab/root/../dir
pwd
cd /home
pwd
cd lab
pwd

Output:

/
/home/csed/
/lab/dir/
/home/
/home/lab/

Note: Use queue as data structure.

4. Write a program to print all numbers from 1 to *N* in binary notation.
   Input:
   5
   Output:
   1
   10
   11
   100
   101
   Note: Use queue as data structure.

5. Children line up in a circle and pass an item from neighbour to neighbour as fast as they can. At a certain point (say after a round) in the game, the action is stopped and the child who has the item is removed from the circle. Play continues until only one child is left.
   - Players: There are *n* children.
   - Elimination Rule: Every *i* th child will be removed after a round.

   Devise a winning strategy, it mean that one can choose a strategy to be the last person left who will be declared as the winner.

   Input Data and Format:

   Enter number of children: *n*

   Elimination Rule: *i*

   Expected Output for Correct and Incorrect Inputs:

   Return: winning strategy i.e. a safe position *s* for some given values of *n* and *i*.

   Example: If $n = 7$ and $i = 3$, then the safe position is $s = 4$. The persons at positions 3, 6, 2, 7, 5, 1 are removed in order, and person at position 4 survives.

   Sample Output:

   Please enter values of *n* and *i*: 5 2
   The removal sequence is as follows -
   [1] Firstly, the person at position 2 is removed.
   [2] Then person at position 4 is removed.
   [3] Then person at position 1 is removed.
   [4] Finally, the person at position 5 is removed. Hence the person at position 3 survives. (WINNER)

   Note: Use queue as data structure.