Online-1

Postfix expression. A postfix expression is an arithmetic expression where every operator follows its two operands. For example, "12+" is a postfix expression whose result is 3. Similarly, "25+36+*" is a postfix expression. You may evaluate the expression by applying the operator to its preceding two operands whenever possible. The evaluation of "25+36+* will go in this way:

  25+36+*
=25+36+*
=736+*
=736+*
=79*
=79*
=63

So, the result is 63. In this task, you will evaluate a postfix expression using your list. In this problem, a postfix arithmetic expression will be given as input. You may input this in a ***string*** variable. Your program will evaluate the expression and output the result. The algorithm to evaluate a postfix expression is as follows:

1. Initialize your list at the beginning of your program.
2. Take input string from user.
3. Scan the characters of the input string one by one starting from the leftmost position. For each character, do one of the following-
   a. If it is a digit ('0', '1', ..., '9'), convert it to an integer value. Then insert the value in the list by calling the *insertItem* function. Remember that in C programming language, a character implies a value equal to the ASCII value of the corresponding character. For example, the character "0" has an ASCII value of 48. So, to convert the character value to integer value, you have to subtract 48. So, '0' - 48 = 48 - 48 = 0, '8' - 48 = 56 - 48 = 8, and so on.
   b. If it is an operator such as "+", "-", "*", and "/", then remove two items from the list calling *deleteLast* function twice. Then apply the operator on the items, find the result, and insert the result again in the list. For example, suppose that the current character is a '*', and 4 and 7 are removed from list. Then, applying '*' operator to 4 and 7 results in 4*7 = 28. Then 28 is again inserted in the list.
4. After scanning of all characters, the result of the full expression will be in the list. Remove the result from the list and output it.
5. Clear your list by calling the *clear* function.

You may assume that input will contain any of the following four operators only: '+', '-', '*', and '/'. You may also assume that there will only be digits not numbers. Sample input and output for Task 7 are given in the following table.

Input	**Output**
12+	3
12+85-*	9
5432++*	45