1. Comment ALL of the code in the file "stack linked list.txt" and "postfix.txt"

Stack linked list.txt

The stack linked list is a code that stacks the linked list. There is struct that can create a node, and class of linked list that has several functions with constructors and destructor

First, code starts with adding header file, iostream that defines input and output stream. The header file cassert, include C header, and ifndef and define header checks if the file is defined, and continues if it is. The last header file included was separate header file made by coder named stackADT.h. The next code we have is the struct with template parameters, the struct nodetype has datatype of value named info and link pointer node link. Then, we have class named linkedstacktype, abstract data type with public stackADT from the header file. The class has overload the assignment operator. The function isEmptystack determines if the stack is empty and returns true if the stack is empty, false if not the case. The isFullStack is also Boolean function, checks if the stack is full and returns true if it is, and returns false if not. The function return type void named initializeStack makes the stack to an empty state, and set the top of the stack null pointer. The void function push is a function that adds new data into the top of the stack if the stack exists and not full. The return data type top function returns the element of the top, if the top of the stack is not empty and if exists, if the top is empty then it terminates the program. The void function pop removes the top element of the stack. Then we have default constructor that has nullpointer top. Also copy constructor that takes otherstack as the parameter that you are trying to copy, using function copyStack. The destructor deletes the stack by function initializeStack function.

Postfix.txt

The postfix.txt is a program that can calculates postfix equations. There are functions evaluateExpression, evaluateOpr, discardExp, and printResult.

The header file that was added were iostream, iomanip, fstream and mystack.h. The iomanip is called to set the precision on data type double. The fstream is to call reader file and output file. The mystack.h is header file created by the user.

The function evaluateExpression takes parameter of input file and output file double data type of the stack, takes data type char as ch, and Boolean pointer. The function declares the num as the

double and goes into while loop checking the char from parameter is not equal to =. In while loop, there is a switch statement that takes ch. And if the ch is #, then set num that was declared inside the function as the input file, and also as output file. And check if the stack is full to push the num into the stack if not full. If the stack is full then terminate and break the case statement with the default switch statement. After checking the case statement there is if statement, checking isExpOk has no error and sets inpF and outF to ch. But if case ch is not a #, then outF is empty. The void function evaluateOpr takes the out as a parameter of output file, stack, char, and Boolean function isExpOk that checks is expression is okay. In this function, there are op1 and op2 of double data type for temp to calculate, check with if statement if the stack is empty and return false on Boolean function. If stack is not empty then set op2 equals to the stack top, and delete also the op1 equals to the stack top after popping the stack top was used on op2. The else statement checks if the equation is addition, subtraction, multiplication, or division(one more step in division return false if the denominator is 0). And push the result of op1 and op2 answer into the new stack push. The function discardExp takes in the reader file, output file and pointer char ch. While ch not equal to =, get ch from reader file and set it to output file. The printResullt function prints the result. In the function result is declared as the double data type, if the Boolean expression isExpOk is true, check the stack if it is empty as well and find error or other cases by else statements. Until the stack is empty initialize the result as the stack top and pops the each of the stack top and if the stack is empty return with the value of the result.

In the main function, we start with declaring the Boolean expressionOk, char ch, stack and set it to the size of 100, and reader file and output file. The code would check if the reader file and open output file as well. Then set the value of the double into decimal point to 0.00 by setprecision(2). And by using evaluateExpression function, calculates the postfix equation and print result by calling function printResult.