

Assignment: Write a program that will evaluate postfix expressions.
Prohibition: Use of the C++ Standard Template Library is prohibited in the implementation of this project.
Program Files: Project 3 consists of files **p03.cpp**, **Scan03.h**, **Scan03.l**, **Stack03.h**, **Stack03.cpp**, and **p03make**. Project file names are exactly as given. Failure to employ the foregoing names will result in a score of **zero (0)** for this project

Project files must be stored in the **root directory of your student account**. Failure to store project files in the root directory of your student account will result in a score of **zero (0)** for this project.
Program Files: Project 3 consists of the following files.

Project files must be stored in the root directory of your student account. Failure to store project files in the root directory of your student account will result in a score of **zero (0)** for this project.

File	Description
p03.cpp	File p03.cpp contains functions that process command line arguments and evaluates postfix expressions.
Scan03.h	File Scan03.h is the interface to functions and data types exported by file Scan03.l . File Scan03.h is found in the class directory ~tt/cs2613/ .
Scan03.l	File Scan03.l is in the class directory ~tt/cs2613/ and extracts tokens from the input file. Each time function yylex is called a token is read from the input file.
Stack03.h	File Stack03.h contains the definition of class Stack . Class Stack implements an integer Stack by dynamically allocating linked elements . Each element on the stack contains an integer. File Stack03.h is not implemented as a template.
Stack03.cpp	File Stack03.cpp contains the implementation of member functions of class Stack .
p03make	File p03make contains instructions for creating executable file p03 . Instructions in file p03make are accepted by the UNIX utility make . Since file Scan03.l requires special translation, use file p03make in the class directory to create program p03 .

Command Line: Project **3** can be invoked with zero, one, or two program parameters. The first program parameter is the input file name. The second parameter is the output file name. Sample command lines together with corresponding actions by program **p03** are shown below. Boldfaced type indicates data entered at the keyboard by the user.

\$ p03

Enter the input file name: **i03.dat**

Enter the output file name: **o03.dat**

\$ p03 i03.dat

Enter the output file name: **o03.dat**

\$p03 i03.dat o03.dat

Input File: The input file consists of a sequence of integers and arithmetic operators in postfix form as shown in Figure 1.

Output File: Figure 2 contains the output of program **p03** given the input shown in Figure 1. Print the contents of the stack after the last token has been processed.

Figure 1. $7 \ 6 \ * \ 3 \ + \ 8 \ 8 \ 1 \ 7 \ + \ /$

Input file format

Figure 2. Stack[tos]=11

Output file format Stack[tos-1]=45