ICSI213 Data Structures

Notice that students are expected to start the lab as soon as the description is available and seek feedback during the lab. Labs are contiguous study of the lecture or used as stepping-stones for the projects. Skipping lab activities would impact the learning significantly.

Lab 03 Stack Application

For this lab, you will work with a partner to solve the following problems. Submit one PDF file with all solutions included.

Convert the infix expression a –(b + c)/d + e into postfix form using the infix to postfix algorithm discussed in class. You must show actions/operations and the status of the stack after each step of the algorithm. The worksheet is on the next two pages.

11 Tokens	11 points
Total/ 11	11 points
Total/ 5	5 points

Student name:

Convert the infix expression *a* –(*b* + *c*)/*d* + *e* into postfix form. You must show actions/operations and the status of the stack after each step of the algorithm in the table. The following algorithm is one we discussed in class.

Convert an infix expression to postfix:

As long as there are more tokens, get the next token.

- if the token is an operand, append it to the postfix string.
- if the token is "(", push it onto the stack.
- if the token is an operator, (order operators by precedence)
  - if the stack is empty, push the operator onto the stack.
  - if the stack is not empty, pop operators of greater or equal precedence from the stack and append them to postfix string, stop when you encounter "(" or an operator of lower precedence or when the stack is empty. And then, push the new operator onto the stack.
- when you encounter a ")", pop operators off the stack and append them to the end of the postfix string until you encounter matching "(". (Ignore/remove "(".)
- when you reach the end of the infix string, append the remaining content of the stack to the postfix string.

Student name:

Token	**Actions/Operations**	**Stack(bottom to top)**	**Postfix string**
a			
-			
(			
b			
+			
c			
)			
/			
d			
+			
e			