Report:	**r02**
Assignment	Send a note to me, trturner@uco.edu, having a single attachment that is your report describing project p03, Postfix Evaluation as specified below.
Report Specifications	The report must be organized exactly as given in the r02 Report Template http://cs2.uco.edu/~trt/cs2613/r02ReportTemplate.pdf.
Cover Page Instructions	Change all text that is colored red in the r0s Report Template (http://cs2.uco.edu/~trt/cs2613/r02ReportTemplate.pdf). 1. **Header**: Change the name in the given in the header to your name. 2. **Author Identification Block**: Complete the **Author Identification Block** as given in the r02 Report Template. 3. **Scoring Block**: Make no changes to the **Scoring Block**.
Naming Specifications	1. The ***subject*** of the note should be your *CRN-last-name-first-name*-**r02**. For example if your name is Alan Turing and you are enrolled in CRN 00000, the subject line should be **00000-Turing-Alan-r02.** 2. The name of your ***report*** should be should be your *CRN-last-name-first-name*-**r02.docx**. For example if your name is Alan Turing and you are enrolled in CRN 00000, the name of your summary should be **00000-Turing-Alan-r02.docx.**

Scoring Block			
Component	**Available**	**Earned**	**Explanation**
Cover page	1	1	
Interface: Command Line	3	3	
Interface: Class Descriptions	4	4	
Algorithm: Problem	1	1	
Algorithm: Solution	1	1	
Algorithm: Steps	2	2	
Algorithm: Example	2	2	
Algorithm: Diagrams	1	1	
Test: Explanations for each test	3	3	
Test: Data for each test	2	2	
Test: Results for each test	2	2	
Source code	3	3	
Total	**25**	**25**	

Report Specifications

The report must be typed, contain no grammatical errors, and must be organized as follows:

1. **(1 point)** Complete the cover page template found on the course web page. Change the name of the file containing the completed template to the name of the ***report*** as specified below. The cover page is the first page of your summary.
2. On a new page, begin your report.
3. **(7 points) Interface description(s).** An interface description contains a precise specification of inputs, outputs, and how a program or function is called.
   - 3.1. Describe command line arguments and parameters of functions. Give an example that demonstrates how the program is called.
   - 3.2. Describe the interface for the classes exercised in this project. Interface descriptions contain an overview of the class and each function in the class. Member functions specifications include a description of the parameters and return value. Be sure to include an example of how each class is used.
4. **(7 points) Algorithm description(s).** A description of the algorithm includes a clearly written discussion of the problem, insight into the solution, data structures used to solve the problem, and a ***step-by-step*** process that results in a solution to the problem. It is impossible to have an adequate description of the algorithm without an example and illustrative ***diagrams***.
5. **(7 points) Test description**. A description of testing includes an explanation of how the project was tested and the tests that were applied. A discussion for the purpose of each set of test data is appropriate. For example, does this test exercise a particular exception? All the functionality of the project should be tested. Tests can also be used to determine the test coverage. Test coverage contains the percent of lines of code in the project that were executed.

(3 points) Source code. Include a copy of the actual code of the project.