CS173 HW 4 -Collections

Due Date: March 15, 2019, at 23:00 pm. Submit your source files with all .txt files on notebowl.

This is a group of two students project. Group members should be different than the group of hw3.

Q1(40p)
Chapter 5- Exercise: 6

Q2 (60p)
Write a client program that uses the Stack abstract data type to compile a simple arithmetic expression without parentheses. For example, the expression

a + b * c - d

should be compiled according to the following table

Operator	Operand1	Operand2	Result
*	b	c	z
+	a	z	y
-	y	d	x

The table shows the order in which the operations are performed (*, +, –) and operands for each operator. The result column gives the name of an identifier (working backward from z) chosen to hold each result. Assume that the operands are the letters *a* through *m* and the operators are (+, –, *, /).

Your program should read each character and process it as follows:

- If the character is blank, ignore it.
- If the character is neither blank nor an operand nor an operator, display an error message and terminate the program.
- If it is an operand, push it onto the operand stack.
- If it is an operator, compare its precedence to that of the operator on top of the operator stack.
    - If the current operator has higher precedence than the one currently on top of the stack (or if the stack is empty), it should be pushed onto the operator stack.
    - If the current operator has the same or lower precedence, the operator on top of the operator stack must be evaluated next.
        - This is done by popping that operator off the operator stack along with a pair of operands from the operand stack and writing a new line in the output table.
        - The character selected to hold the result should then be pushed onto the operand stack.
        - Next, the current operator should be compared to the new top of the operator stack.
        - Continue to generate output lines until the top of the operator stack has lower precedence than the current operator or until it is empty.
- At this point, push the current operator onto the top of the stack and examine the next character in the data string.
- When the end of the string is reached, pop any remaining operator along with its operand pair just described.
- Remember to push the result character onto the operand stack after each table line is generated.

Q3 (Extra- 40p)

In Scrabble, knowing the two-letter word list is important because those short words make it easy to “hook” a new word into tiles already on the board. Another list that Scrabble experts memorize is the list of three-letter words that can be formed by adding a letter to the front or back of a two-letter word. Write a program that generates this list for each two-letter word. You can find the all English words in the file of `EnglisWords.txt`

Enter Two-letter word << of
I found 2 three-letter words contains of
Three-letter words list
off
oft
Enter Two-letter word << in
I found 19 three-letter words contains of in
Three-letter words list
ain
bin
din
fin
gin
hin
ins
inn
ink
jin
kin
lin
pin
tin
win
zin
yin
rin
sin

Enter Two-letter word << enter
exit