Faculty of Engineering and Information Technology

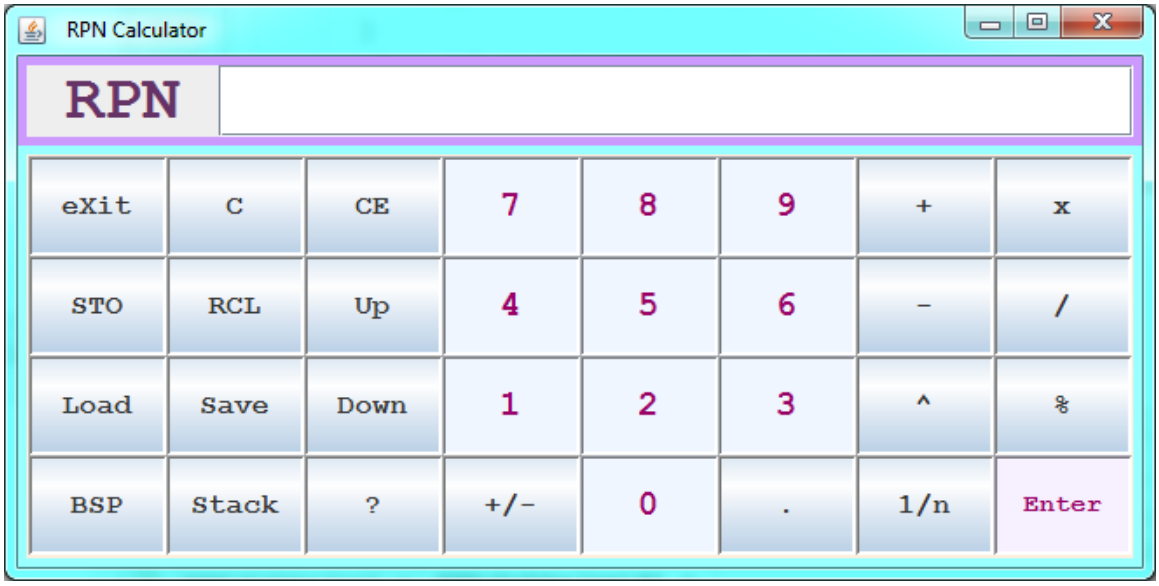
Computer Programming II

Practical Project , First Term 2019/2020

Deadline Date: Thursday 12/12/2019

RPN Calculator

Your assignment is to write an Object-oriented program in Java to run an RPN calculator simulator. A GUI Swing interface as shown in figure#1 (*RPNForm.class*) with minimum functionality is available for you to use.



Figure#1:RPN form

RPN stands for "Reverse Polish Notation." This type of calculator works very well for computer science and engineering problems. It uses ***post-fix*** notation instead of ***in-fix*** notation--that is, operations are written after the numbers operated on. For example, in ***in-fix*** notation, adding two numbers, then subtracting a third number might be written as "4 + 5 - 6". In ***post-fix*** notation, the same expression could be written as " 4 5 + 6 -" or " 4 5 6 - +".

You will also need to create and utilize an **RPNCalculator** class with the following minimum requirements:

class **RPNCalculator**

- should at least have *protected* **properties**:
  - **protected Deque<Double> theStack**
  - **protected Register [ ] registers** -- the 26 registers for number storage

- Member **methods** should include:

- a *constructor*
- a method to act on a given command

The program needs to perform the required operations for each of the following key button presses:

- **+** -- adds top two elements in the stack, and replaces them in the stack with the answer.
- **-** -- subtracts the top two elements in the stack, and replaces them in the stack with the answer.
- ***** -- multiplies the top two elements in the stack, and replaces them in the stack with the answer
- **/** -- divides the top two elements in the stack, and replaces them in the stack with the answer
- **^** -- exponentiation -- *e.g.* 4 5^ would be 4 to the 5^{th} power (4^5).
- **%** -- mod
- **+/-** -- change sign of element displayed -- (*blank if not a number*)
- ? -- help mode. Displays help for the next button pressed.
- **1/n** -- replaces the number in the display with its multiplicative inverse (*blank if not a number*)
- **Enter** -- push value to the stack.
- **C** -- **C**lear -- removes all elements from the stack
- **CE** -- **C**lear **E**ntry -- removes the top element from the stack
- **Down** -- rotate the stack ***Down*** -- remove the bottom element and place it at the top.
- **Save** -- **Save** the stack and the registers to Files – [two files –one file (stack file) to save the stack and the second file (register file) to save the registers].
- **RCL** -- **Rec**all a number from one of the 26 internal storage registers.
- **?** -- gives help on the next button pressed
- **Load** -- **Load** from files -- loads the stack list element-by-element from the file to **the stack** – and loads the registers form the file to the **registers**.
- **BSP**— (**B**ack**sp**ace) used to correct mistakes in the display field by erases the last typed character on the display.
- **Stack** -- show the stack **(see figure#2)**
- **STO (A - Z)** -- **Sto**re number to one of 26 internal **registers**. These can be saved when the save button is clicked and can be restored when load button is clicked.
- **Up** -- rotate the stack ***Up*** -- remove the top element and place it at the bottom.
- **eXit** -- **Exit** the program
- **0 - 9, ".", "c", +, -, * (x, X), /, %, ^, ? and Enter** -- can be entered from the buttons or the keyboard utilizing the display textfield
- should accept an operation after entering a number without an "Enter" in between

Any method encountering an error (such as trying to do a binary operation without having at least two elements in the stack) should display an error message and leave the stack unchanged.

Addition, Subtraction, etc.

Basic math operations (Add, Subtract, Multiply and Divide) in an RPN calculator follows easy-to-understand rules and is unambiguous. First you need to get the two numbers you want on the stack

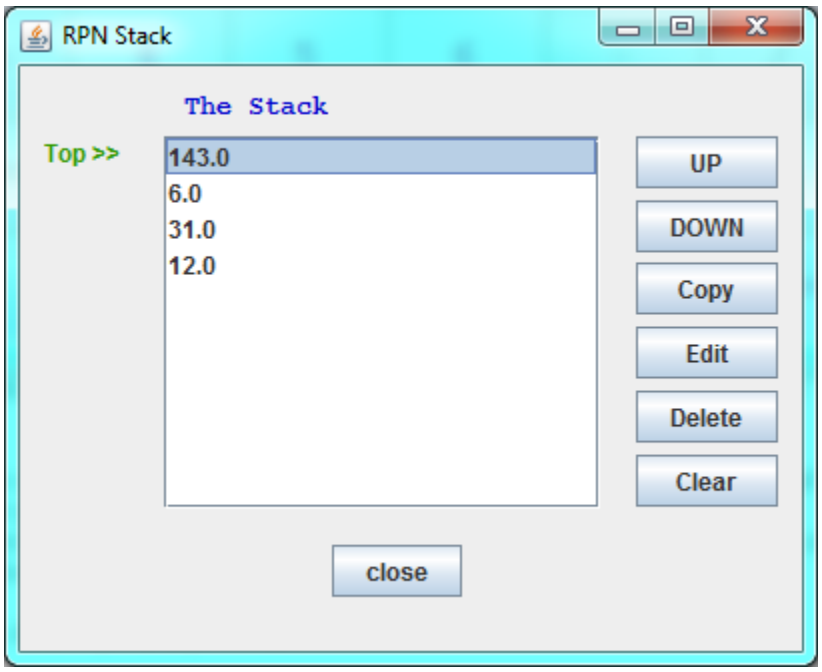
The RPN calculator makes quick work of long chain calculations. The trick is to remember what's on the stack (it easy since you can see all stack locations on screen as shown in figure#2). The trick with long calculations is to work from the innermost set of parenthesis out and knowing the precedence of operations (multiply and divide before addition and subtraction).

For example, the equation 45.2 + (6 / 2.98) * 44.33

You would enter 6, enter 2.98 and press the Divide key. Enter 44.33 and press the multiply key. Once again, the result is in the top of stack. Enter 45.2 and press the plus key. The result is the final result of the equation above.

The Memory Stack

The automatic memory stack is at the heart of all calculations in this (or any) RPN calculator. It provides for an efficient storage of intermediate results during complex equations and allows for easy visualization of where your entered numbers are stored. The automatic memory stack is shown as follows (The stack form is shown when stack button is clicked)



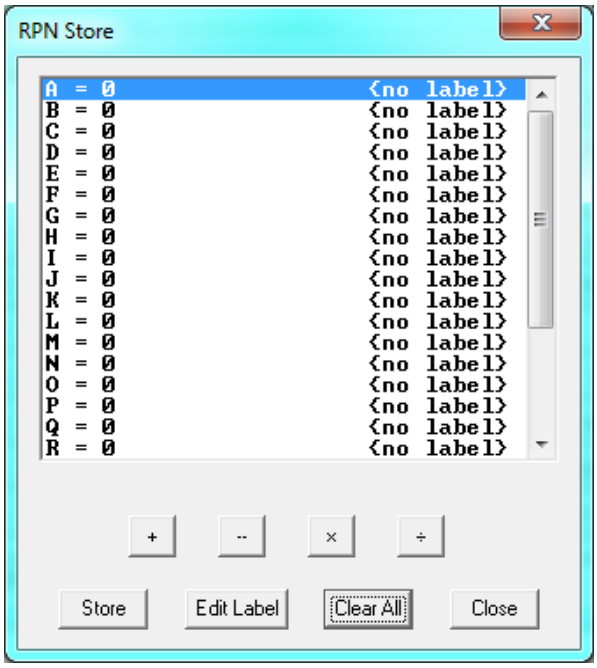
Figure#2:Stack form

Internal Registers

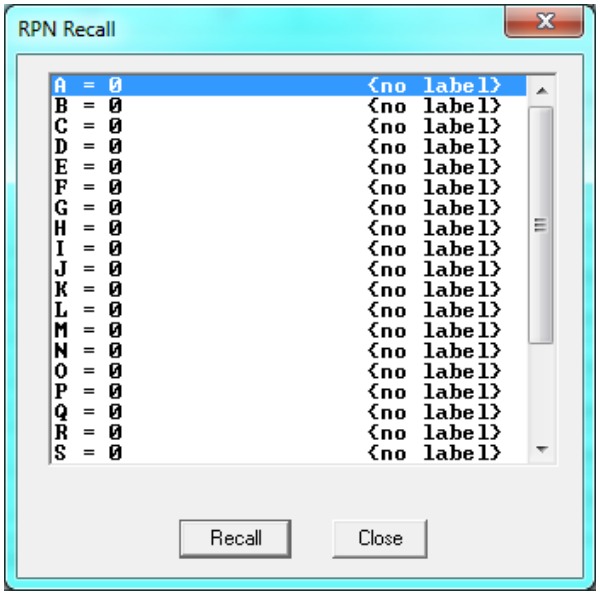
The RPN Calculator will also have 26 internal registers (memory locations) which will be used to store the last value that was computed. When the Calculator is "turned on", internal memory locations will store zeros, and its internal stack will be empty.
The internal registers can be used for storing and retrieving data. They may be accessed using the following button operations:

STO : show RPN Store form(see figure#3), The form supports the Full register arithmetic.

RCL: show RPN Recall (see figure#4), used to recall an register value to the display field and push the value on the stack.



Figure#3: Register Store



Figure#4:Register Recall

Instruction:

-Your solution must contain multiple class..

*- This project could be done **individual or** by group of a **three student at maximum**.*

*- Submit in CD only – **CD includes your name, ID** and Section Number.*

You must submit the following :

All source files + All class files stored in a folder, The folder name is your name .

If a student copies the Project of another student, he/she will be assigned a zero grade for the Project.

The date of Handover & discussion : Thursday 12 / 12/ 2019

With Our Best Wishes