Report of Designing the Evaluation of Postfix Expression

This mid-term project is expected to design a program that is able to evaluate postfix expression (eg. [2 4 5 + *]). It is suggested to implement a stack to execute operations in the array of expression. Since we read the expression form left to right which has operators at the end, we can start to initialize an empty stack to store the operands to perform the operations with the following methods:

1. Scan the characters in expression form left to right, if the character is operand, push it to the stack.
2. If the character scanned is an operator, pop two operands from the stack and execute the operation, then push the result back to the stack.
3. While the entire postfix expression is scanned, the top element of the stack would be the final result.

Here is the pseudocode:

```
function EvaluatePostfix(expression):

    new Stack Stack

    for 0 <= i < expression.length:

        if expression[i] = “ ”:

            continue

        end if

        if expression[i] is operator:

            operand2 = Stack[top]

            Stack.pop()

            operand1 = Stack[top]

            Stack.pop()

            Result = ExecuteOperation(expression[i], operand1, operand2)

            Stack.push(Result)

        end if

        if expression[i] is numeric number:
            Stack.push(expression[i])
        end if
    end for
end function


function ExecuteOperation(operation, operand1, operand2):
    if operation = “+”:
        return operand1 + operand2
    end if
    if operation = “-”:
        return operand1 - operand2
    end if
    if operation = “*”:
        return operand1 * operand2
    end if
    if operation = “/”:
        return operand1 / operand2
    end if
end function
```

Some challenges in this project are identify a correct workflow to manage the operation in postfix expression with using Stack to pop the operands and push back the result. Some other difficulty is while reading the operand and push the operand into the stacks, a convention for numeric number has multiple digits that is required to be aware on their actual values. Overall, the conditional workflow is widely required for this project.