Calculator Requirements (v3.0)

**Introduction:**

The main goal of this project is to create a calculator that must perform the following requirements.

**Non-Functional Requirements:**

1. The Calculator will run on the command line.
2. The Calculator program will be written in the Python language.
3. The Calculator shall take an expression and evaluate it, as well as show the components that make up the expression.

For example:

Inputting 11 +9 will produce the following:

Evaluation: 20

Components:

Integer, Value 11

Whitespace

Operator, Value '+'

Integer, Value 9

Note: Any remaining examples will only show the “Evaluation” portion of the output.

**Functional Requirements:**

1. The Calculator shall be able to take in two integers and perform addition on them.

For example:

Inputting 1+10 will produce 11 as a result.

1. The Calculator shall be able to take in two integers and perform subtraction on them.

For example:

Inputting 10-9 will produce 1 as a result.

1. The Calculator shall be able to take in two integers and perform multiplication on them.

For example:

Inputting 10\*9 will produce 90 as a result.

1. The Calculator shall be able to take in two integers and perform integer division on them. If the result of the division is a decimal, the Calculator will round up or down to the nearest integer.

For example:

Inputting 90/10 will produce 9 as a result.

Inputting 5/2 will produce 3 as a result.

Inputting 1/3 will produce 0 as a result.

1. The Calculator shall check if the denominator of a division query is 0, and if so it will produce the result of “undefined”, regardless of what the numerator is.

For example:

Inputting 9/0 will produce the result “undefined”.

1. The Calculator shall allow as input equations that contain consecutive operations to be performed.

For example:

Inputting 1+1+1+1 will produce 4 as a result.

Inputting 10-1-1-1 will produce 7 as a result.

Inputting 10+1-2 will produce 9 as a result.

1. The Calculator shall allow unary operation (+/-) on any integer.

For example:

Inputting -9 will produce the result -9.

Inputting --9 (that’s two minus signs) will produce the result 9.

Inputting +9-1 will be the same as inputting 9 -1 which will produce the result of 8.

Inputting -9-1 will produce the result of -10.

1. The Calculator shall check the order of operators and determine what will be executed first based on the precedence. Multiplication (\*) and division (/) have the highest precedence from left to right, follow by addition (+) and subtraction (-).

For example:

Inputting 2+3\*4/2 will produce 8 as a result.

1. The Calculator shall allow for multiple calculations to be performed before a session is terminated. If the user does not type in “quit”, the program will keep asking the user to input another equation.
2. The Calculator program will allow whitespace between operands and operators.

For example:

If we have 1\_+\_10 as an input (where “\_” is a whitespace), the program will take it as 1+10 and produce the result of 11.

1. The Calculator program will verify whether an equation that is received as input is formatted correctly or not.

For example:

Inputting 1+ is incorrect, since there is a missing operand.

Inputting +1, however, will be considered as a unary operation and will produce 1 as the result.