



the c++ core-language - ■ ■

ECUE apprentissage de la programmation - the c++ language

Valérie Roy et Basile Marchand
Mines ParisTech

① the core-language

The goal of this exercise is to develop a program in order to take in input argument a mathematical expression, write using RPN and evaluate this expressions.

The Reverse Polish Notation

Is a mathematical notation in which operators follow their operands

For example the expression

$$1 + 2 * (3 - 1)$$

can be written as follow

$$1231 - * +$$

The evaluation of the RPN expression can be achieved as follow

```
for each token in expression:
```

```
  if token is an operator:
```

```
    operand_2 <- pop from the stack
```

```
    operand_1 <- pop from the stack
```

```
    result <- evaluate token with operand_1 and operand_2
```

```
    push result back onto the stack
```

```
  else if token is an operand:
```

```
    push token onto the stack
```

```
  endif
```

```
endfor
```

```
result <- pop from the stack
```

HINTS for the code of the calculatrice

in file `calc.cpp`

```
// your code here
int main (int argc, char* argv []) {
    // your code here
}
```

```
$ g++ calc.cpp -o calc
$ ./calc 3 5 ! x
-15
```

- the negation is the symbol `!` to avoid confusion with the binary minus `-`
- the multiplication is the symbol `x` to avoid confusion with the shell symbol `*` meaning zero or more of any character when searching for a pattern
- to deal with the stack operations, write functions and pass the stack as an argument to the functions
- write a function to verify that a characters string is a positive integer
- write a function to verify that a character is an operator