

## Assignment 6 - Modular arithmetic

Authors :

- Prakhar Mathur 1906328
- Sanjay Marreddi 1904119
- Rishabh Tripathi 1904129

### Problem Statement

1. (25 points) Recall the modular arithmetic that we studied in the class. Create a `class noModN`<sup>1</sup> using a C++ template with non-type parameter for  $N$ . It has a single private data which is an integer between 0 and  $N - 1$ . This data should be initialised using a constructor that takes an arbitrary integer (may be positive or negative) and converts it to modulo  $N$  representation. Do not use `%` operator.

For this, you will need to implement the `divide(int, int)` algorithm which returns the quotient and the remainder. You can use an array or `struct` for returning two values.

Also create a default constructor.

Overload the operators `+`, `*` and `++`<sup>2</sup> for this class.

Recall that in multiplication, you should go modulo  $N$  in each of the intermediate steps.

Now, in the `main` procedure, take integer inputs  $a, b, c$  from the user and create objects of the `class noModN`. Output  $(a+b)*c$ ,  $a++$ ,  $++a$  in this exact sequence. The output should be clearly understandable.

Here is the Sample Input and Output:

```
-19 23 17
`A` object of the class nModN with N = 10, with input data value = -19 got converted into -9
`B` object of the class nModN with N = 10, with input data value = 23 got converted into 3
`C` object of the class nModN with N = 10, with input data value = 17 got converted into 7
Performing the given operation on the 3 objects of the class nModN, i.e. (A+B)*C yields = -2
Performing Post fix operation on the object `A` of the class nModN, i.e. A++ yields -9
Performing Pre fix operation on the object `A` of the class nModN, i.e. ++A yields -7
```