

# OOP Assignment - 4

Satyaprasad

Due on 4 Oct 2020

# 1 Matrix Operations with Big Integers

We know that `int` in C++ has limited range. The range of positive integers for `long long unsigned` is 0 to 18,446,744,073,709,551,615. It covers upto 19 digit numbers. What if we need to workout with more than 19 digit numbers?

Let suppose

$$a = 987453464345678445645645985456456445$$

$$b = 693369835987453464345678445645645985$$

then,

$$a + b = 1680823300333131909991324431102102430$$

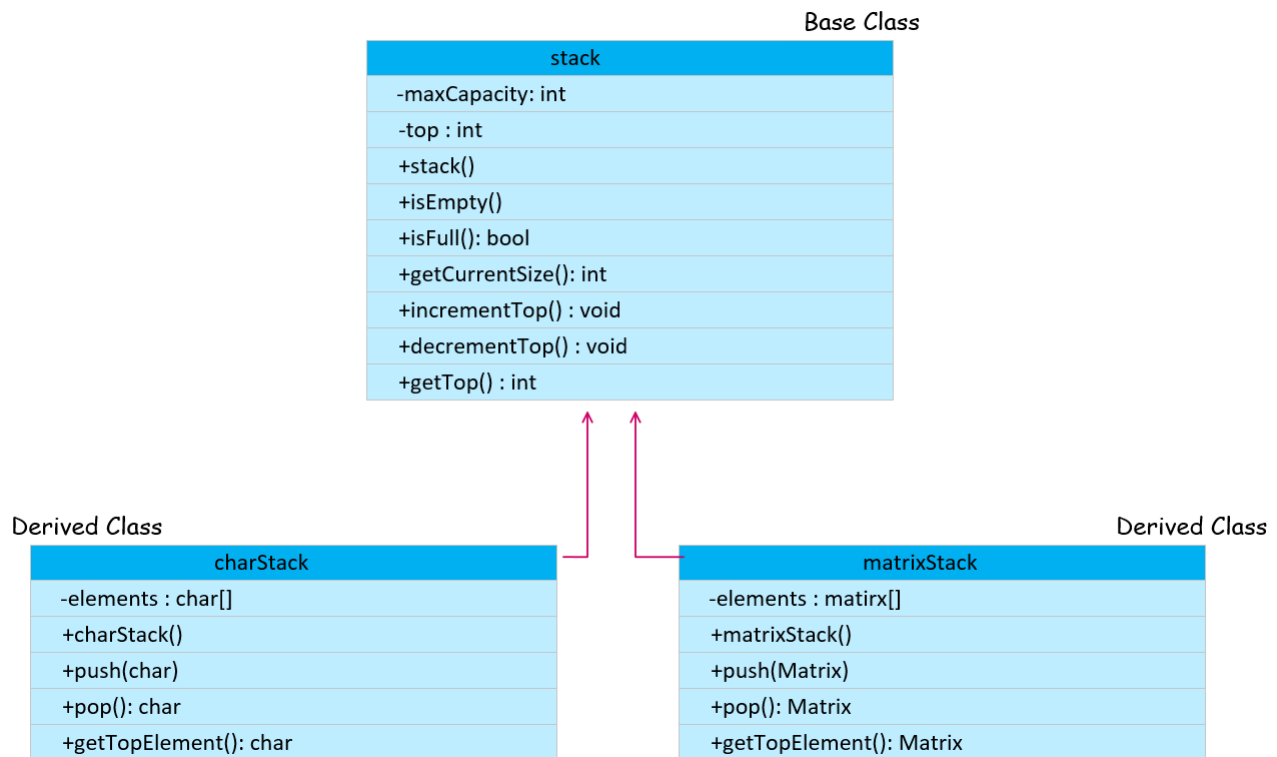
$$a - b = 294083628358224981299967539810810460$$

$$a * b = 684670446618605791068660078689706882000939978585608022359157521041623325$$

these operations can't be performed with primitive data types such as long int, long long int etc. To perform such operations, we need to define our own number class. Let us define a class `Number` and overload the operators `+`, `-`, `*`, `==`, `>>` and `<<`. Class `Number` design is provided in the template code. In order to perform above operations, you have to define them.

Using above `Number` class, implement a `Matrix` class such that every element in a matrix is of type `Number` and overload the operators `+`, `-`, `*`, `==`, `>>` and `<<` for `Matrix` class. See the code for class structure.

Now define following structure,



Using above class definitions [Number](#), [Matrix](#), [stack](#), [charStack](#) and [matrixStack](#) answer the following:

Let  $A, B, C \dots Z$  represent matrices from [Matrix](#) class and  $a, b, c \dots z$  represent numbers from [Number](#) class. Evaluate the given expression such as  $A + B * A - a + (b + C) * B$ .

Where  $A + B, A * B, A - B$  are usual matrix operations and  $a + b, a - b, a * b$  are usual number operations.

$A \pm a$  is defined as each element of matrix  $A$  is added/subtracted by the number  $a$ .

$A * b$  is defined as each element of matrix  $A$  is multiply by the number  $b$ .

For example,

$$A+a = \begin{bmatrix} 987453464345678445645645985456456445 & 456456459854564999 \\ 844564564598545 & 99999999999999999999999999999999 \end{bmatrix} + 99565665465132165465$$

$$= \begin{bmatrix} 987453464345678545211311450588621910 & 100022121924986730464 \\ 99566510029696764010 & 100000000000099565665465132165464 \end{bmatrix}$$

**NOTE:** Assume that the use of variables in the expressions A to Z (or a to z) must be subsequent. Meaning, using an alphabet without using earlier alphabet is illegal. E.g.,  $A+C*a$  is illegal, Since C is being used where as B is not used.

#### Input:

An expression with matrix and number variables, inputting particular matrices and numbers in the order.

First line is expression.

Second line is for reading first matrix number of rows and num of columns.

Third line is for reading data for first matrix.

Fourth line is for reading second matrix number of rows and num of columns.

Fifth line is for reading data for second matrix.

[Repeat this until exhaust all matrix variables in the expression.](#)

Next line is for reading first number.

Further next line is for reading second number.

[Repeat this until exhaust all number variables in the expression.](#)

#### Output:

A matrix. (Which is result of expression evaluated at given matrices and numbers)

#### SAMPLE RUN:

##### Input:

B+A\*a

2 2

5 6 7 8

2 2

9 1 0 5

3256545654654654654984654651894584

##### Output:

16282728273273273273274923273259472929

19539273927927927927929907927911367505

22795819582582582582584892582563262088

26052365237237237237239877237215156677