

## CS222: Assignment 4 - Arithmetic over arbitrary base numbers

1. Submission deadline: Monday, 30 January at 3:00 pm.
  2. Follow good coding practices to gain more marks.
  3. No copying among the students or from the Internet or any other source.
  4. The assignment can be submitted in groups of size  $\leq 2$ .
  5. Submit a `.cpp` file and a `.pdf` file.
  6. Write the names and roll numbers of the students at the top of each file.
  7. The files should be called  
`base_b_firstRollNumber_secondRollNumber.cpp`,  
`base_b_firstRollNumber_secondRollNumber.cpp`,
  8. The pdf should contain the output obtained when each program was run and the time complexity of each function, assuming that the computer stores the numbers in base  $b$ .
  9. In case you do not know about C++ templates, check <https://www.learncpp.com/cpp-tutorial/template-non-type-parameters/>.
  10. For more information check: <https://stackoverflow.com/questions/66546257/in-c-can-we-create-a-class-for-each-integer>
- 

1. Create a `class base_b` using a C++ template with non-type parameter for  $b$ . It has a vector as a private data. This data should be initialised using a constructor that takes an arbitrary positive integer and converts it to base  $b$  representation.

Also create a default constructor.

Overload the operators `+`, `-`, `*`, `/` and `++`<sup>1</sup> for this class.

Now, in the `main` procedure, take integer inputs `x`, `y`, `z` from the user and create objects of the `class base_b`, where `b` is 37. Output `(x+y)*z`, `x/y2`, `x++`, `++x` in this exact sequence. The output should be clearly understandable.

[If you are brave, you can output/input using `cout/cin`. I.e. by overloading the `<<` and `>>` operators. Write in the common assignment comment on Google classroom whether you have implemented it. No credits for this!]

---

<sup>1</sup>Recall that there are two versions of the `++` operator: prefix `++` as in `++i` and postfix `++` as in `i++`

<sup>2</sup>`x/y` should output `x = qy+r`, where `x`, `y`, `q`, `r` are objects of `class base_b`