

Task 1 : Number System Program

Weightage-10

Create a C++ program to store and perform operations on different types of Coordinate systems. The selected few are: Complex plane, cylindrical, spherical notation and vector notation. Create a base class System and derive several classes based on different systems. Each class must be able to implement basic and elementary operations valid for all like add, subtract, shift of origin, conversion of each type to others and particular specialized operations must be suitable defined. Create a menu driven class Run to perform each action.

FUNCTIONALITY:

Perform Operator overloading for overloaded operations.

Implement suitable inheritance model.

Maintain an enumerator for each number system.

Implement File IO to save particular results.

Store history of operations performed using suitable data structure.

Implement suitable functionality to convert number from some number system to another.

Delete an entry from the log.

REQUIRED:

Create a separate functionality to store headers and definitions. (.h file)

Proper commenting and alignment as given alongside.

<http://cse.iitkgp.ac.in/~abhij/course/lab/CompLab-I/misc.html>

Deallocate memory as and when needed.

Try to implement better and innovative features alongside those mentioned here.
"INNOVATION is the key."

=====

Hint: You can use STL

Task 2:ROS Basics

Weightage-10

You will be given two files to read from, which will contain two sets of numbers. Make two nodes in ROS each reading from one file. Both will publish the read numbers on separate topics say Topic_1 and Topic_2. A third node will subscribe to these topics sort the numbers and print them in a file.

For example:

The first file contains let's say:

9 11 2 34 51 61 3 7 0 62 \eof

The second file contains:

10 78 99 5 68 1 4 8 6 57 \eof

The nodes Topic_1 and Topic_2 should read these files and publish them.

The third node say sorter will subscribe to both and prints into the third file:

0 1 2 3 4 5 6 7 8 9 10 11 ... And so on in a sorted fashion.

Task 3:ROS Turtlesim

Weightage-20

Read about Turtlesim and graph algorithms. You will be given a file containing a matrix of 0's and 1's. 0's represent free path and 1's represent obstacles. In a node, first of all move the turtle sim to the top left corner of the window. Then you have to map the matrix for the turtlesim window and traverse the window avoiding the obstacles, to finally reach the opposite corner of the window.

Example matrix:

```
0 0 1 0 0 0 0 0 1 1 0 0 0
0 0 1 0 0 0 0 0 1 1 0 0 0
0 0 1 0 0 1 0 0 1 1 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0
```