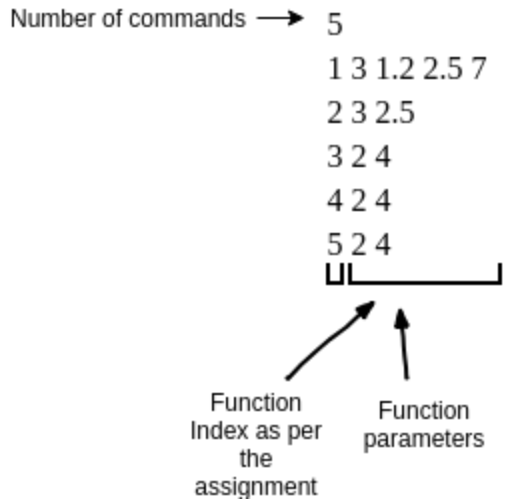
INPUT FORMAT GUIDELINES FOR ASSIGNMENT (Prog_Eval-1)

Important guidelines: Please make sure that the code is neat and commented, and does not contain any unused/ unnecessary functions (which are not required to execute the program). Also, make sure that the input, output, and the submission formats are strictly followed (absence of which may lead to deduction of marks).

Remark: t>=current is allowed for insert and delete operations.

INPUT FORMAT:

To run a command for n (=5) functions, the following format is followed:



The above runs five function calls (as specified by the first row of the image). All the next rows describe the function calls.

Each of the next rows has the following format:

{Function_Id}, {Input_Parameter_1}, {Input_Parameter_2}, …, {Input_Parameter_n}

Function_Id or the function index refers to the index/ order of the functions specified in the Programming Assignment.

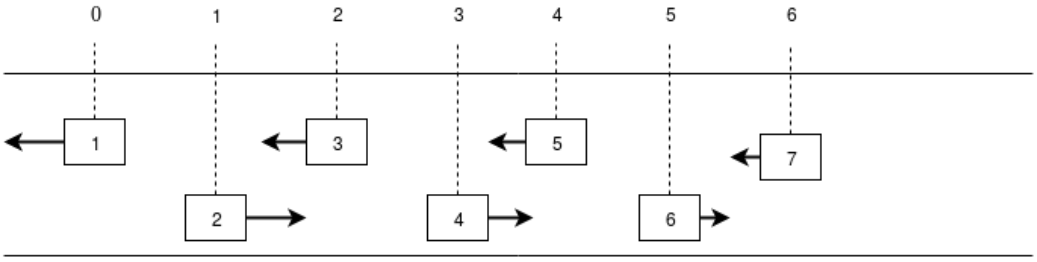
For instance, the first row in the above image specifies that we need to execute 5 function commands.

The second row specifies that: the First function needs to be executed (as '1' is the first element of this row) i.e. int insert(int r, float x, float t, int d); 3, 1.2, 2.5, 7, represent the input parameter values corresponding to r, x, t, and d as specified in the function call.

OUTPUT FORMAT:

Students need to return an output for all the functions except *insert* and *delete* (since ``delete" is a keyword in C++, you will name this function ``remove").

For instance, consider the example scenario below at t = 5 (Refer Fig: cars-fig1.jpeg):



For INPUT function calls:

15
1 8 8 0 1
2 8 1
1 1 4 1 0
1 3 5 2 0
1 4 1 3 1
1 5 5 4 0
1 6 4 4 1
1 7 6 5 0
1 2 1 5 1
3 4 5
4 5 5
5 4 5
6 4 5
7 2 5
7 2 7

Following are the respective function outputs for functions 3-7:

3
6
3
3
0
2

Here the first and second row represents the registration numbers returned by 3rd and 4th functions (viz., find_immediate_left and find_immediate_right, respectively), the third, fourth, and fifth row represents the counts returned by 5th, 6th, and 7th functions (viz., count_left, count_right, and number_of_crossings, respectively).

SUBMISSION GUIDELINES:

You need to submit your file with your "<entry number>-eval1" as the file name. For instance, 2015csx0004-eval1.cpp