Implement a program in C or C++ that does the following:

Given a list of objects, we would like to know the number of objects that lies in a range of values.

The list of objects doesn't change after the first time (i.e no objects are added, removed or changed. Hence you should optimize the data structure for reads and not worry about updates).

For example: Assume the list of objects are random 10 million integers. An example query might be like "get number of objects >1000" or "get number of objects >10000 and <120000" or "get number of objects = 1000"

Write few test cases with different types of objects(integers, floats or strings of arbitrary length, user defined object type) of lengths from 10K to 100M, and print out the time taken (in microseconds) for few sample representative queries. Note that the list only stores only 1 type of object, however your data structure should be flexible enough to store any data type (if the caller wants to store non primitive data types, the ordering will be defined by a comparator function of the caller’s data type).

You may not use any external libraries like Boost or 3rd party libraries. Using standard STL containers is fine. Your class should have at the minimum 2 public apis (one for adding an object during initialization and another for querying objects in a range). Make reasonable assumptions if you have any questions. Assume you are designing a library and your objective is to provide reasonable apis to future users of this library.

Your code will be run on an Unix environment, so please make sure your code is compilable with gcc and include a Makefile and README file in your submission. Running "make" should compile the program and run the test cases.

Submit the source file(s), Makefile, README etc in a tar pkg. Your code will be judged based on correctness and efficiency of algorithm (both run-time and storage), cleaness, style, using thoughtful design, tests etc.