Candidate exercise – Ordered List Module (No' 2)

- Development tool –Visual C# (.Net Application) / C++ / Java
- Help tools –Internet.
- Time frame – As much as you need. (Here at Novidea).
- Needed products:
  - QuickPopOrderedList class:
    - The class should implement a list that can be inserted with any object type. When pulled out of the list, the objects will be pulled out ordered (by any ordering algorithm they implement). For example: if we insert integers into the list in the following order- 1,5,3,2,7 and them pull out integers from the list, the integers will be pulled out in the following order- 7,5,3,2,1
    - The class should be optimized for quick 'pop' operation.
    - The class should perform well in a multi-threaded environment
    - A small test console application should be written to test the class
  - QuickPushOrderedList class
    - Same as QuickPopOrderedList class, only with optimization for quick push. Pop time is irrelevant.
  - A test application that tests the efficiency of the lists according to their optimizations.
- Your goals
  - Design the class interface
  - Design the object model used for implementation

  (Note: after finalizing each goal it should be presented, before moving to the next one)
- Limitations
  - You may not use any of the .Net System.Collection objects, the standard library (in C++) or Java collection
- General directions:
  - The first priority is a closed solution that implements all the things, which are asked, with as little as can be BUGs.
  - Operational code standards should be supplied both on the real matter issues (Performance, readability…) and the cosmetic issues (Clean, tabbed, documented…)

Good Luck!