

CS1020E: DATA STRUCTURES AND ALGORITHMS I

Tutorial 4 – ADT, Containers, Sequence Containers

(Week 6, starting 12 September 2016)

1. *Container ADT vs List ADT*

A **Container** is a collection of objects. The C++ standard library has defined some containers.

Reference: <http://www.cplusplus.com/reference/stl/>

Some of these containers are **Lists** (not just the STL list class). In C++ terminology, they are termed sequence containers.

- (a) What is the relationship between a Container and a List (sequence container)?
- (b) What do both ADTs (Container vs sequence container) have in common?
- (c) What extra functionality do List ADTs (sequence containers) have?

2. *List ADT Implementations*

In lectures, we have learned two List implementations – **array-based** and **reference-based** (i.e. in C++, using pointers). STL `vector<T>` is an array-based list implementation, while `list<T>` is a reference-based implementation. Let us **compare and contrast** both implementations.

For a list containing N elements, **how many elements would be accessed/modified** when:

- (a) Adding to end of the List (new index == N / tail)
- (b) Adding to front of the List (index == 0 / head)
- (c) Removing from front of the List (index == 0 / head)
- (d) Getting element at any index, on average (from index == 0 to index == N-1)

- ☺ -

Have you been revising
and ***practicing*** daily?