

Mod 4 Prompt Questions - Noboru Hayashi

Q1:

Some things with a LIFO characteristic, instead of a FIFO, is possibly implemented with queues.

Let's say there's multiple tennis balls in a case, in which balls must be put or taken one by one from the top of it. So if we place tennis balls in that box, it follows a LIFO characteristic that the latest ball put into can be taken.

In the other hand, assuming we implement the tennis ball case with a queue A - the ball is placed to the top and taken from the bottom of the case A, and if we want it still follows a LIFO order, we can another empty FIFO tennis ball case, let's say queue B. We can place a ball from the bottom of A to B until the last one, which is the last-in ball. The second pop can be done by dequeuing from B to A until the last one, and so on..

Q2:

The key features of a list is storing multiple data elements as a group, and it's ordered. If we want to access the smallest or the largest element, or want to store data in a specific order, the list data type would be useful.

Q3:

Unlike actual implementation, hints or references are not required for the ADT of the list implementation, since ADT only defined an interface of the list

Q4:

I think the method to deep copy the list, also one to search specific value inside the list would be helpful

Q5:

Since there are multiple ways to implement a list type, such as an array, or a linked list, etc., due to characteristics of each implementation, it becomes important to find out what the requirement of the implementation of a project or program. For example, if the program is read-heavy one, the

linked list requires sequential reads whereas the array implementation provides random accesses, so the latter is preferred in this situation.

Q6:

Pros of an array implementation of a list: it allows random accesses, easily implemented

Cons: limited size even though shifting on deletion, or garbage collection prevents false overflow.

Q7:

Pros of a linked list implementation of a list: unlimited size, insertions and deletions are easier by changing pointers, even though the search is sequential.

Cons: is read sequentially.