

## Module 4 - Discussion Prompt Questions

**Adam Rich**

Section EN.605.202.87

Spring 2018

1. What are some of the key distinguishing features of a list? Under what conditions would these be useful?

All list elements can all be accessed, front, back or middle. Stuff can be inserted into the middle of the list, shifting the "index" of all subsequent items. There is no limit on the number or nature of items and they do not need to be homogeneous.

Lists would be useful if there cannot be an upper bound on the number of items and one would need to access more than just the front or back element.

2. Should a list ADT include hints or references to the list implementation?

No, but it may do. Optional methods may give hints as to what implementation might be easier, but that does not mean that the ADT must be implemented that way.

3. What might you consider to be a nice-to-have in a list ADT (not necessary, but potentially useful)?

toString: print out the contents of the list, element by element, as ordered pairs or with a selected delimiter.

4. When might the implementation details of a list be an important consideration?

Every time you implement a list! The developer of the list implementation needs to consider lots of things like how the list is most likely to be used, whether the list is to be sorted or not, should the list be singly or doubly-linked. The language of the implementation might support certain implementations over others.

5. List some pros and cons of an array implementation of a list.

Pro: random access is easier; no need to transverse the list (if using the standard implementation). Cons: has a limit on size (unless the class periodically replaces the array with a deep copy into a bigger one). Insertion and deletion require shifting on the order of  $O(N/2)$ .

6. List some pros and cons of a linked implementation of a list.

Pro: able to add as many elements as you wish. Can make the list doubly linked to walk it in either direction. Cons: because memory is allocated dynamically it may not be available (whereas array memory is all grabbed at once). Also, will have to walk the list to get to the element you want to get to.