

# hw33: a list of $n$ items, defined while-style and recursively

---

## desideratum

The definition should apply to a list having zero items.

## helpful definitions

A *null reference* is a reference to no object.

*node* a repository for a value. A node can be linked to a single other node.

## while-style

A *list* of  $n$  items...

starts with a node called the *head*.

The *head* can be null, to indicate a list having zero items.

Each of the  $n$  items is represented in a separate node.

Each node is linked to exactly one *next* node,

except for one node, which has a null *next* node. Call this node the *tail* node.

## recursively

A *list* comprises a reference to a *chain of nodes*

The reference is called the *head*.

The minimal *chain of nodes* is a null reference, to indicate a list having zero items.

Denote that *list* as being of size 0.

A larger *list* (that is, a *list* of size  $n$  where  $n > 0$ ) comprises a *node* linked to the beginning of a *chain of nodes*.

To be precise, we have to define the “beginning” of a *chain*,

For a null *chain*, the *beginning* of the *chain* is a null reference.

A larger chain has one or more nodes. The *beginning* of such a *chain* is the node that has no links to it.

## commentary on the recursive definition

This definition of *beginning* aims to preclude star shapes from being considered linked lists. Linked lists are linear.

Ideally *beginning* is defined in a way that corresponds to programmers’ intuitive understanding of that term.

This definition of a *chain* of *nodes* implies that there can be at most one node with no links to it.

This definition cries out for improvement. Maybe it would help to define the head of a chain?

The recursive definition of a *list* relies on a *list* in its recursive portion. The while-style definition does not.

A *chain of nodes* differs from a [Chain of Fools](#). (not comp sci; merely an important cultural reference, included here as a reward for having made it this far)