The Problem: Part 1

- The list box is initialized by a list of items. When initialized, the current selection is the first element in the list.

- Create 2 operations: **`arrowUp`** and **`arrowDown`**, which allow you to change the position of the current selection.
  - If you **`arrowUp`** when the first element of the list is selected, nothing should happen (no-op).
  - If you **`arrowDown`** when the last element of the list is selected, nothing should happen (no-op).

- If the controller is given an empty list, there is no currently selected item.

Remember, in later iterations we will expand this behaviour.

The Problem: Part 2

- Add the concept of the "`window`", the portion of the list box which is currently displayed on the GUI.
  - Imagine a list box with 100 elements. The window may be showing elements [10..19].

- The `arrowUp` and `arrowDown` operations have special behaviour at the top and bottom of the window- they move it.

  For example: If the window is showing elements [10..19] and 10 is selected:
  - `arrowUp` changes the current selection to 9 and shifts the window up to show [9..18].
  - `arrowDown` changes the current selection to 11.

Note: You **do not** have to deal with the cases where the **window** touches the top or bottom of the list. (We have to save some fun for next time!)

The Problem: Part 3

- Account for the window hitting top and bottom of the list
  - An `arrowUp` operation at the top of the window when the window is showing [0..x] is a no-op.
  - An `arrowDown` operation at the bottom of the window when the window is showing [x..last] is a no-op also.

- The windows size should be represented by `windowSize`.
  - The default `windowSize` is 10
  - If the list size is less than 10, then `windowSize` becomes the size of the list.

- **Refactor** the code to improve its structure.

The Problem: Part 4

- Add the operations **pageUp** and **pageDown**:

- **pageUp** moves the window up **windowSize** elements so that it is just above it's previous first element.
  - When the operation is complete, no element which was visible in the window before is visible in the window afterward.
  - If **pageUp** can not go up **windowSize** elements because it would hit the top of the list, it stops at its last possible move up.
  - The current selection after a **pageUp** is the last element in the window.

- The **pageDown** operation has the exact opposite behaviour. It moves the window downward, and makes the current selection the top element of the window.