fileSort.c

Input: which sort and file to be sorted
Output: sorted file, each word or number in a new line

Given command line arguments, we detect which sorting method has been passed and which file is to be opened.

Open/Reading from the file:

- We read in byte per byte (as a character) and append each character to a token
- Repeat until we see delimiter
- Token is now considered done
- Create a new node and initialize node's value to token
- Add new node to existing linked list
- Check first linked list node to detect which data type we are currently working with. Store this into a enum mode {Integer, String} which is global.

After reading file is done, a linked list is constructed, where each node holds a token (word or number)

Sorting:

- Two comparators - integer and string
- Using global enum mode, determine which comparator to set the fnptr to
- Using first command line argument, determine which sorting algorithm to implement
- Use comparator to quicksort/insertion sort the linked list, done by swapping the values of linked list nodes.

Print sorted file:

- Go through each node of sorted linked list and print the node's value followed by an endline character