Salazar

Assignment 5

Assignment 5 (Auto-sorting list operations)

Completion requirements

Follow along with the in-class exercise on this, do your best to get it working, and turn in what you come up with here!

Be sure to include at least one test for each function or piece of functionality that should verify that your code is working!  No slacking smile, you should start writing some tests before you write your implementations (just spend a few minutes thinking about the design and then write a few tests using natural language (English is preferred for me to be able to read it smile ))

Create an array-based list or a linked-list (and a bonus for attempting both) that:

automatically inserts values in the correct position based on some order of sorting (perhaps ascending integers or lexicographical sorting of words)

efficiently searches for elements (likely binary search for the array list, but what about the linked-list?)

Make a chart to compare the algorithmic complexity (use Big-O notation) of your insert, remove, and search algorithms you used for your structures

Once you have implemented and tested your code, add to the README file what line(s) of code or inputs and outputs show your work meeting each of the above requirements (or better, include a small screen snip of where it meets the requirement!).

Note: This assignment is to get you to think about and practice with structures that have automatically enforced properties (perhaps auto sorted, since we know sorting can be expensive... and how these might be able to be spread out over time)

//Design

Will use an array-based list of pokemon

The arrange function will automatically rearrange the list alphabetically

Search function will iterate through the whole array till it finds what it is looking for.

Remove function, might use the search function , then delete the results

// test for functions

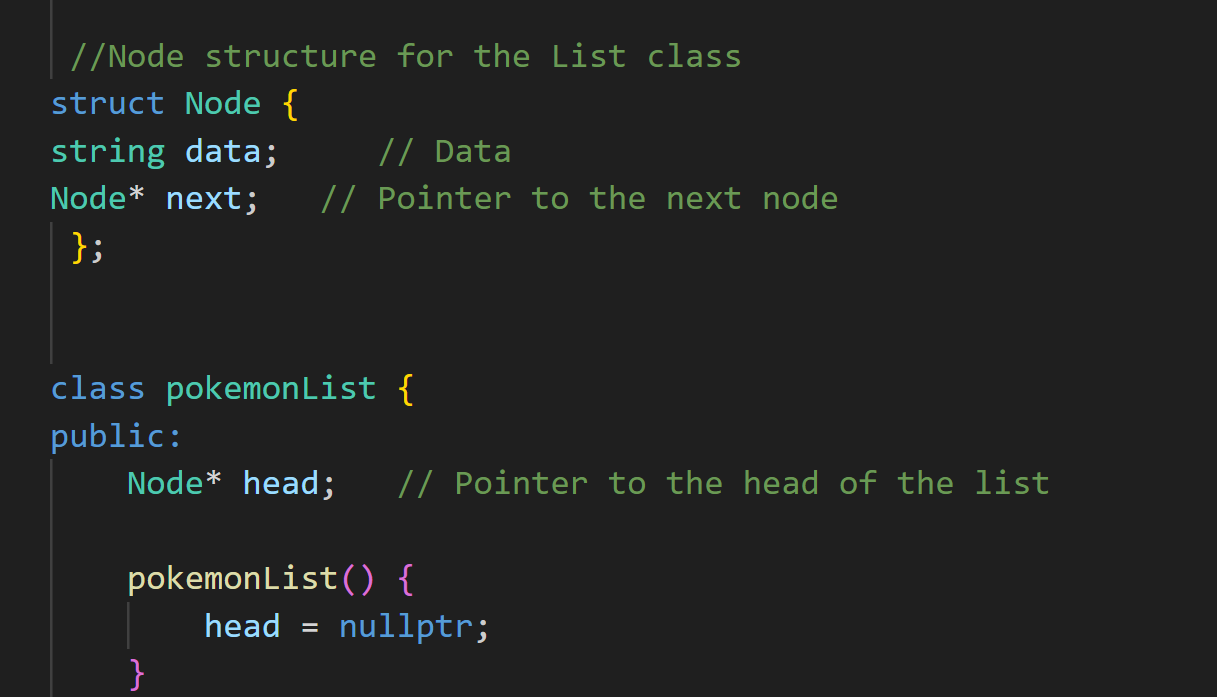
For the auto sorting, I'll just print out the list; it should now be alphabetized.

For the search function, I'll print out its position location, and compare to where it should be

Remove function, after removing the pokemon I can check the lenthg of the list or print out the array and confirm its gone.

//requirements

//Array based list



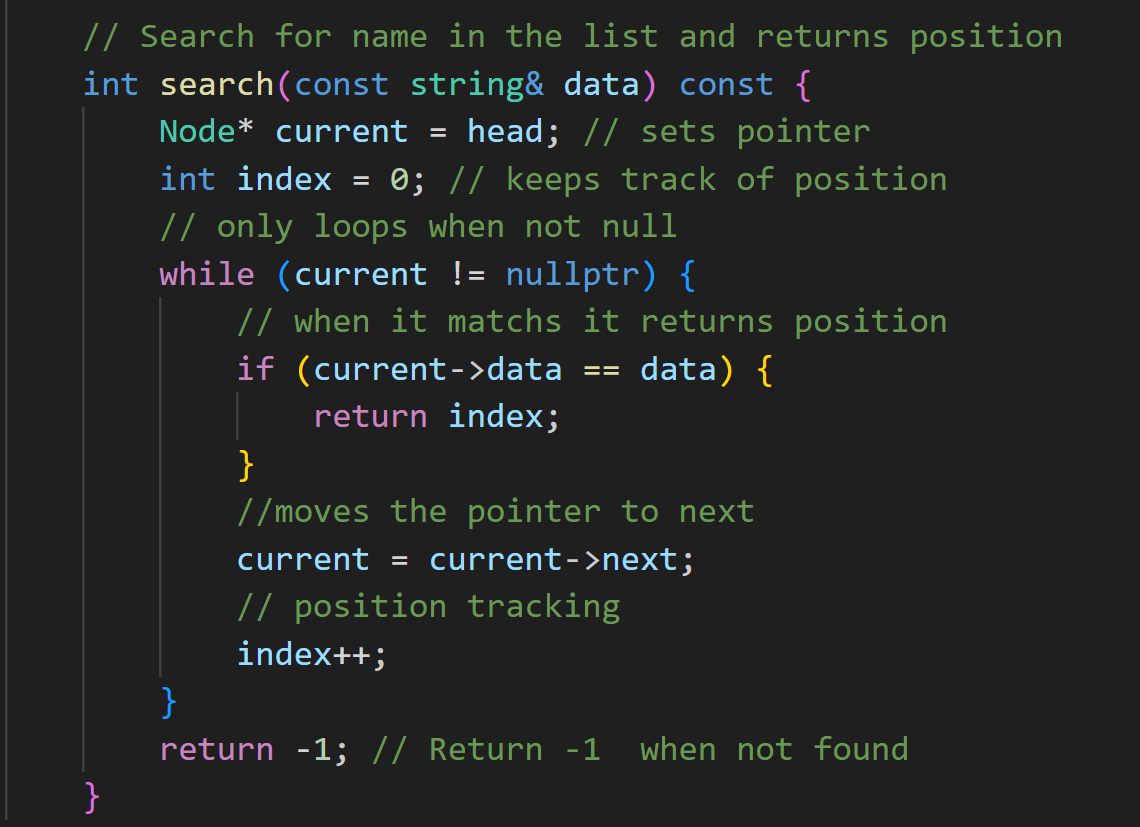


// auto sort

A screenshot of a computer program

Description automatically generated

//search function

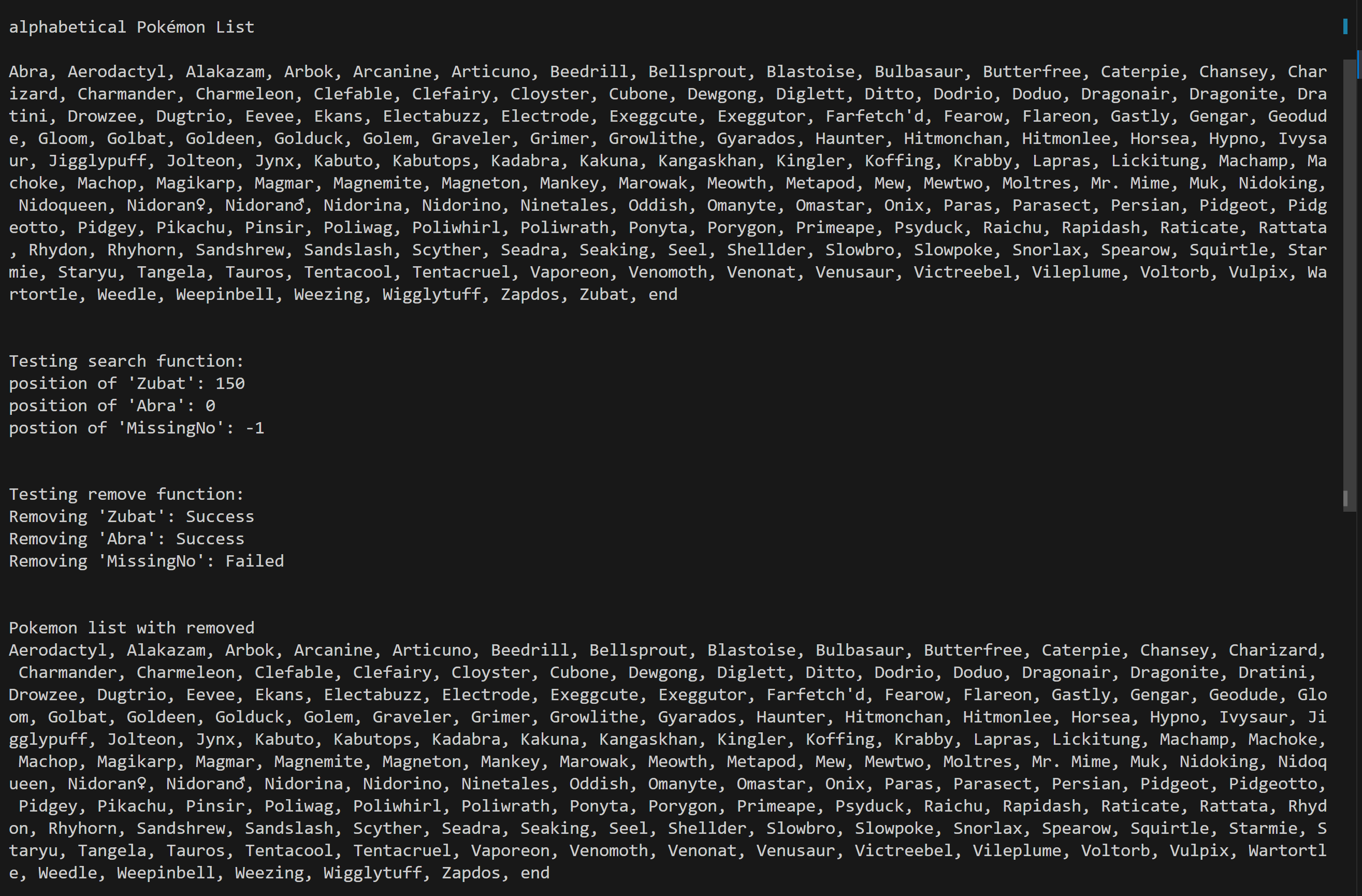


// remove function

A screenshot of a computer program

Description automatically generated

// results from testing



//testing is all done in one big function that test each of the functions to save space from list

A screen shot of a computer program

Description automatically generated

// in terms of complexity all the functions are going to be the same on O(n).

The sort function needs to go through the whole list.

If the Pokemon is not in the list or at the very end, the search function must go through the whole list.

The remove function also does the same while looking through.