CSC 242
Lab 4

Warmup: The solution to the following exercise is provided in the Chapter 6 Student Files. Please review this code to better understand how the contains method is improved to logarithmic runtime.

A sorted bag behaves just like a regular bag but allows the user to visit its items in ascending order with the `for` loop. Therefore, the items added to this type of bag must have a natural ordering and recognize the comparison operators. Some examples of such items are strings and integers.

Define a new class named **`ArraySortedBag`** that supports this capability. Like **`ArrayBag`**, this new class is array based, but its **`in`** operation can now run in logarithmic time. To accomplish this, **`ArraySortedBag`** must place new items into its array in sorted order. The most efficient way to do this is to modify the **`add`** method to insert new items in their proper places. You also have to include a **`__contains__`** method to implement the new, more efficient search. Finally, you must change all references to **`ArrayBag`** to be **`ArraySortedBag`**. (Hint: copy the code from the **ArrayBag** class to a new file and begin making your changes from there.) Write a driver program to test the full functionality of the **`ArraySortedBag`** class—this means, you should write a separate Python program that will use each of the newly implemented class methods of the **`ArraySortedBag`** class.

Now, here is the actual assignment:

Add the method **`clone`** to the **`ArrayBag`** and **`LinkedBag`** classes. This method expects no arguments when called, and returns an exact copy of the type of bag on which it is called. For example, the variable bag2 would contain the numbers 2, 3, and 4 at the end of the following code segment:

```
bag1 = ArrayBag([2,3,4])

bag2 = bag1.clone()

bag1 == bag2  # Returns True

bag1 is bag2  # Returns False
```