Strings and Arrays

Tuesday, January 25, 2022 3:19 PM

1.1 Given a string, determine if it is all unique characters.

Example: isUnique("abc") == true and isUnique("aabc") == false

Brute-Force:

```
HashSet letters = new HashSet();
char[] chars = str.toCharArray();
for(int i=0; i<chars.length; i++)
{
    if(letters.contains(chars[i]))
    {
        return false;
    }
    letters.add(chars[i]);
}
return true;
```

Using a hashset for O(1) adds and lookups.
If a character is in our set, then return false.
else, if we make it to the end of the array then all chars are unique. O(N) in running time and O(N) in space complexit. N = size of string.

Next Step: Don't use a data structure

Sorting a string?

"abcbdb" ⇒ "abbbcd"

```
Char[] chars = str.ToCharArray();
Chars.sort();
```

```
For(int i=0; i<Chars.length-1; i++)
{
    if(chars[i] == chars[i+1])
    {
        return false;
    }
}
return true;
```

Sort the array of chars so that any duplicate chars will be next to each other.

This will run in O(nlogn) time, because of sorting, but take O(1) space.