Improving Linear Search

We can improve our algorithm a little bit by putting the data in sorted order. To do that, just call the `sort()` function which I've supplied, immediately after you've loaded the data, like this:

```
const int kSize = 1000;
Person directory[kSize];
load(directory, kSize);
sort(directory, kSize);
```

Now, when you run the program and look for **Thompson**, it is not immediately found, since Kimberly is no longer at the beginning of the list, but much further down. So, that doesn't seem to be much of an improvement.

Remember, though, that before you sorted the data, you needed to **check every single element** before you could be sure that the name was not found. Now, **you can stop** whenever the name in the directory is greater than the name you are looking for. Just add this code at the end of your loop `do-while` loop:

```
if (contacts[i].name > key) return -1;
```

The average performance time is still `O(n / 2)`, but the **worst case time has improved** from `O(n)` to `O(n / 2)` as well. In general it will take 500 comparisons to see if a name is missing, instead of the 1,000 comparisons you made earlier.

However, this is the best improvement we can make with linear search. To go faster, you'll **need a better algorithm**.