Understanding code: 211 - TreeSet

```
Contact daffy = new Contact( name: "Daffy Duck");
Contact daisy = new Contact( name: "Daisy Duck");
Contact snoopy = new Contact( name: "Snoopy");
Contact archie = new Contact( name: "Archie");

for (Contact c : List.of(daffy, daisy, last, snoopy)) {
    System.out.printf("ceiling(%s)=%s%n", c.getName(), fullSet.ceiling(c));
    System.out.printf("higher(%s)=%s%n", c.getName(), fullSet.higher(c));
}
System.out.println("--------------------------");
```

Dataset (`fullSet`)

The `fullSet` contains contacts sorted by name:

- **Charlie Brown** with phone `(333) 444-5555`
- **Daffy Duck** with email `daffy@google.com`
- **Linus Van Pelt** with email `lvpelt2015@gmail.com`
- **Lucy Van Pelt** with phone `(564) 208-6852`
- **Maid Marion** with phone `(123) 456-7890`
- **Mickey Mouse** with phone `(999) 888-7777`
- **Minnie Mouse** with phone `(456) 780-5666`
- **Robin Hood** with phone `(564) 789-3000`

Explanation of the Output

1. **`ceiling(Daffy Duck)`**
   - Finds the smallest contact greater than or equal to `"Daffy Duck"`. Since `"Daffy Duck"` exists in `fullSet`, it returns `"Daffy Duck"` with email `[daffy@google.com]`.
2. **`higher(Daffy Duck)`**
   - Finds the smallest contact strictly greater than `"Daffy Duck"`, so it skips `"Daffy Duck"` and returns `"Linus Van Pelt"` with email `[lvpelt2015@gmail.com]`.
3. **`ceiling(Daisy Duck)`**
   - `"Daisy Duck"` doesn't exist in the `fullSet`, so `ceiling` finds the next closest contact alphabetically, which is `"Linus Van Pelt"`.
4. **`higher(Daisy Duck)`**
   - Similar to `ceiling`, `higher` returns `"Linus Van Pelt"` as it's the next greater contact after `"Daisy Duck"`.
5. **`ceiling(Robin Hood)`**
   - `"Robin Hood"` is in the set, so `ceiling(Robin Hood)` returns `"Robin Hood"` itself with phone `(564) 789-3000`.
6. **`higher(Robin Hood)`**

- `"Robin Hood"` is the last entry in the set, so `higher(Robin Hood)` has no greater contact, resulting in `null`.

7. **`ceiling(Snoopy)` and `higher(Snoopy)`**
   - `"Snoopy"` is beyond the last entry in the set and isn't present, so both `ceiling` and `higher` return `null`.

Summary

The `ceiling` method finds the closest match at or above the specified element, while `higher` only looks for strictly greater elements. When the specified contact doesn't exist, they return the nearest larger contact or `null` if none exists.

```
for (Contact c : List.of(daffy, daisy, first, archie)) {
    System.out.printf("floor(%s)=%s%n", c.getName(), fullSet.floor(c));
    System.out.printf("lower(%s)=%s%n", c.getName(), fullSet.lower(c));
}
System.out.println("-------------------------");
```

In this example, `floor` and `lower` methods are used to navigate the sorted `TreeSet` called `fullSet` to find elements based on their relative position to each `Contact` object.

- **`floor(Contact c)`**: Finds the largest element less than or equal to the specified contact.
- **`lower(Contact c)`**: Finds the largest element strictly less than the specified contact.

Explanation of Output with Data:

- **Charlie Brown**: [] [(333) 444-5555]
- **Daffy Duck**: [daffy@google.com] []
- **Linus Van Pelt**: [lvpelt2015@gmail.com] []
- **Lucy Van Pelt**: [] [(564) 208-6852]
- **Maid Marion**: [] [(123) 456-7890]
- **Mickey Mouse**: [] [(999) 888-7777]
- **Minnie Mouse**: [] [(456) 780-5666]
- **Robin Hood**: [] [(564) 789-3000]And the sample output:

1. **`floor(Daffy Duck)`** and **`lower(Daffy Duck)`**:
    - `floor("Daffy Duck")` returns `"Daffy Duck"` itself, as it's present in the set.
    - `lower("Daffy Duck")` returns `"Charlie Brown"` as it's the largest element before `"Daffy Duck"`.
2. **`floor(Daisy Duck)`** and **`lower(Daisy Duck)`**:
    - `"Daisy Duck"` is not in `fullSet`.
    - `floor("Daisy Duck")` returns `"Daffy Duck"` (the nearest, largest element less than or equal to `"Daisy Duck"`).
    - `lower("Daisy Duck")` also returns `"Daffy Duck"` as it's strictly before `"Daisy Duck"`.
3. **`floor(Charlie Brown)`** and **`lower(Charlie Brown)`**:
    - `floor("Charlie Brown")` returns `"Charlie Brown"` as it's in `fullSet`.
    - `lower("Charlie Brown")` returns `null` because there's no element before `"Charlie Brown"`.
4. **`floor(Archie)`** and **`lower(Archie)`**:
    - `"Archie"` isn't in `fullSet`, and it's alphabetically before `"Charlie Brown"`.
    - Both `floor("Archie")` and `lower("Archie")` return `null` since there's no element less than or equal to `"Archie"` in the set.

These methods provide efficient ways to search within a sorted `TreeSet` when an element's position matters, such as finding elements that are near or immediately precede a given key.