Worksheet #13:
Parallelism in Java Streams, Parallel Prefix Sums

1. What output will the following Java Streams code print?

2. Which stream operation in this example could benefit from a parallel prefix sum implementation, and why? (Assume a larger array when answering this question, so that overheads of parallelism are not an issue.)

```java
Arrays.asList("a1", "a2", "b1", "c2", "c1")
.parallelStream()
.filter(s -> s.startsWith("c"))
.sorted()
.map(String::toUpperCase)
.forEach(System.out::println);
```