Use the ForkJoin framework to write the following method in Java:

```java
public static int[] filterEmpty(String[] arr)
    Returns an array with the lengths of the non-empty strings from arr (in order).
```

For example, if `arr` is `"", "", "cse", "332", ", "hw", ", "7", "rox"`,
then `filterEmpty(arr) == [3, 3, 2, 1, 3]`.

A parallel algorithm to solve this problem in $O(\lg n)$ span and $O(n)$ work is
the following:

1. Do a parallel map to produce a bit set
2. Do a parallel prefix over the bit set
3. Do a parallel map to produce the output

In lecture, we discussed `parallelPrefix` together, and code for it is included
in the gitlab repository. Rather than re-implementing that piece yourself,
you should just use it. For the other two parts though, you should write
them. Do not bother with a sequential cutoff for this exercise.