CSE 332: Data Structures and Parallelism

Exercises (Parallelism)

Directions: *Submit your solutions using gitlab. You* must *fill out the partners form for* `para` *to get access to your repository. Choosing a partner is disabled, because these exercises are solo.*

EX13. `getLeftMostIndex` (20 points)

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

```
public static int getLeftMostIndex(char[] needle, char[] haystack, int
sequentialCutoff)
```

Returns the index of the *left-most* occurrence of `needle` in `haystack` (think of `needle` and `haystack` as `Strings`) or -1 if there is no such occurrence.

For example, `getLeftMostIndex("cse332", "Dudecse4ocse332momcse332Rox") == 9` and `getLeftMostIndex("sucks", "Dudecse4ocse332momcse332Rox") == -1`.

Your code must actually use the `sequentialCutoff` argument. You may assume that `needle.length` is much smaller than `haystack.length`. A solution that peeks across subproblem boundaries to decide partial matches will be significantly cleaner and simpler than one that does not.