```
// Longest Common Prefix in an Array
// Given a array of N strings, find the longest common
prefix among all strings present in the array.

// Expected Time Complexity: O(N*max(|arr_i|)).
// Expected Auxiliary Space: O(max(|arr_i|)) for result.
```

**Idea**: First, find the maximum length string in the array and store it in a variable $s$. Then, traverse the array, when you find that $s$ is not equal to the an element, compare the substring equal to the $s$'s length from $0^{th}$ index, if it is also not equal to $s$…then, start from the last letter of $s$ and reduce its size whenever the corresponding letter (same index letter) is not equal to it.

A while loop will be inside a for loop. Because the while loop will only traverse the letters of a string, it will not hamper the time complexity.