Kick Start 2022 - Coding Practice with Kick Start Session #2

Analysis: Irregular Expressions

View problem and solution walkthrough video

The problem translates to determining whether the given string contains a substring (`spell`) satisfying the given conditions:

- The spell must have 3 "words": `start, middle, end`
- The start word must equal end word, and must be at least 2 syllables
- The middle word must have at least 1 syllable

Recall that a word is defined as a string of characters of any length with at least one syllable. A syllable is a string that must have at least one vowel in `a, e, i, o, u` and can be of any length.

Let $S = E$ be the start and end word, and M be the middle word. Let v_i represent the i-th vowel in the string.

We can make a few observations and begin iterating through the string.

1. We can assume that S begins at v_1. We can ignore all the consonants before v_1, and assume they are either not part of the spell (in the case for S), or they are part of M (in the case for E), since M's only constraint is that it contains at least 1 vowel.
2. We must assume that S stops at v_2. This is the first string we encounter which satisfies the constraint that S has 2 syllables, and stopping here ensures we do not miss any valid spells. Similar to the above observation, we can do this because we can simply tack on any extra consonants after S to M, and any extra consonants after E does not matter.
3. Then, we can start looking for M. Once we see v_3, we should stop. We have now satisfied the only constraint for M. Note that M can have more vowels or consonants after v_3, and no constraints will be broken.
4. After seeing v_3, we can begin our search for E. We already established what substring E must be, so we can simply do a search on the rest of the string starting from the character after v_3. If S appears, then our search is finished. Otherwise, a spell with $S = E$ does not exist. In this case, we need to repeat the process and look for a new S and E.

If we need to look for a new S and E, where should we start? We can look for a new S starting at v_2 and ending and v_3, and follow the same logic as above. If we iterate through all the vowels and still do not find a valid spell, then we know that the string does not contain one, and can return `Nothing`.

Now let us look at the runtime of this solution. Let $\mathbf{N}$ be the length of the given string. We iterate through the entire string to look for a valid S, M, and E, at most $\mathbf{N}$ times in the worst case. Thus, the runtime is $O(\mathbf{N}^2)$.

Another simple solution is to use RegEx to search whether a spell satisfies the given constraints. The runtime of this solution is $O(\mathbf{N})$, but keep in mind that the RegEx construction could take up to $O(2^{\mathbf{C}})$ construction time and space, where $\mathbf{C}$ is the size of the regex.