# A Simple Fuzzy Search Algorith 2.0

***Abdullah Saquib*** *(Independent Researcher)*

There are two updates in this version. In the previous version, each matched word of the `search_string` were given equal weight in evaluatung the `score` value returned by the `compare_strings` function. The weight of each word was irrespective of the size of the word. For instance, consider `db_string="Harry Potter and the Chamber of Secrets"` and `search_string="The adventures of Rocky and Bullwinkle"`. The words 'of', 'and', 'the' of search_string are present in db_string. If they were given equal weights, the result would be around 3/7 or 0.43 as 3 out of 7 words of `search_string` are present in `db_string`. The result is independent of the size of the matched words. In order to solve this issue, we have introduced weigths for each word in the search_string. We have taken weights of each word to be proportional to the length of the word. Now, words like 'adventures', 'Bullwinkle', 'Rocky' holds higher weights than 'a', 'of' and 'the'.

To understand the origin of the second issue, consider computing `compare_strings` function with the above two strings as arguments. You will get 0.63 instead of the expected 0.43. The reason is `fuzzy_compare_words('a', 'adventure')` returns 1 instead of some small number like 1/9 or 0.11 (`'a'` matches with one out of the nine letters of `'adventure'`). We have fixed this issue by modifying the function `fuzzy_compare_words`. In this version, the value returned by the `fuzzy_compare_words` is marks divided by the length of the word with greater size. In this case, first letters of the two words `'a'` and `'adventure'` matches, the value of marks=1. The length of the `'adventure'` is 9 and of `'a'` is 1. The returned value will be 1/9 or 0.11.
In the previous version, the returned value was marks divided by the length of the word with smaller size. So, returned value would be 1/1 or 1. In previous version, we had in mind the idea that there should be a perfect match between words like 'call' , 'called', 'calling'. So, we returned marks divided by the smallest of the lengths of the two words.

With these two update the value returned by compare_strings(`"Harry Potter and the Chamber of Secrets", "search_string="The adventures of Rocky and Bullwinkle"`) is 0.33 about half the value returned by the previous version.