

Functionality Investigation Sheet

Feature: Main Search

Problem: Quickly find and access recipes among recipes in the database that correspond to a user's search request. Stand out among the competition with a fluid search engine.

Option 1: Naive Linear Algorithm

In this option, we are using a naive linear algorithm to search the user's search query in the recipes' titles, ingredients or descriptions. The advantage of this algorithm is its very simple code and that it runs without having to preprocess the search query. However, due to this lack of preprocessing the algorithm's time complexity is higher, as it includes unnecessary steps in its matching process.

Benefits

- simple code
- no preprocessing of search query
- lower space complexity

Disadvantages

- includes unnecessary steps in matching process
- runs through text one character at a time
- higher time complexity

Best case: $\Omega(m)$ (linear)

Worst case: $O((n-m+1)*m)$ (square)

of operations per second: 18.540 ops/s*

Option 2: KMP Algorithm

In this option, we are using the KMP algorithm to find the search query in the recipes' titles, ingredients or descriptions. This algorithm includes the preprocessing of the search query to find prefixes that are also suffixes to facilitate the search and skip unnecessary steps in the matching process. This causes the algorithm's time complexity to be lower, which clearly is its advantage. Its disadvantage, however, is that it uses extra space to compute and store the aforementioned preprocessed information.

Benefits

- skips unnecessary steps in matching process
- runs through text intelligently
- lower time complexity

Disadvantages

- additional memory needed to preprocess search query
- higher space complexity

Best case: $\Omega(m+m)$ (linear)

Worst case: $O((n-m+1)+m)$ (linear)

of operations per second: 27.200 ops/s*

Solution Retention:

Although both algorithms' best cases are linear, they differ greatly in its worst case scenario with the naive linear algorithm being of square complexity. In addition to that, in a test search, the KMP algorithm performs higher with 27.200 ops/s against 18.540 ops/s for the naive linear algorithm. Therefore, I have chosen the KMP algorithm over the naive linear algorithm.

* Find tomato in „Spread the dough to the size of the pan, spread the mustard on the dough, add the tuna. Cut the tomatoes into slices and place them on the fish, add a little crème fraîche to the whole pie and top with grated Gruyere. Bake for 30 minutes"