

# Raport z zadania wzorce

Konrad Jurczyński-Chu 318666

Bogumił Stoma 325233

## Podział zadań

Bogumił Stoma: algorytmy KMP, KR, raport

Konrad Jurczyński-Chu: algorytm naiwny, testy, wykresy

## Struktura projektu

### plik `pattern_search.py`

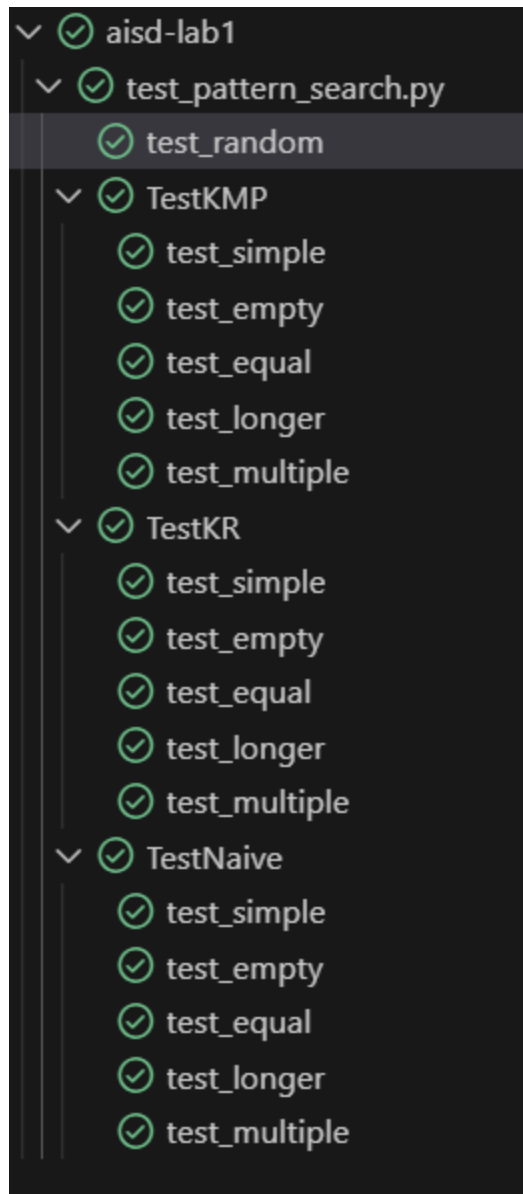
Biblioteka zawierająca implementacje algorytmów: naiwnego, KMP, KR w funkcjach o nazwach odpowiednio: `searchN`, `searchKMP`, `searchKR`.

Funkcje przyjmują argumenty `pattern: str`, `text: str`

### plik `test_pattern_search.py`

Plik zawierający testy sprawdzające poprawne działanie algorytmów, testy zawierają przypadki :

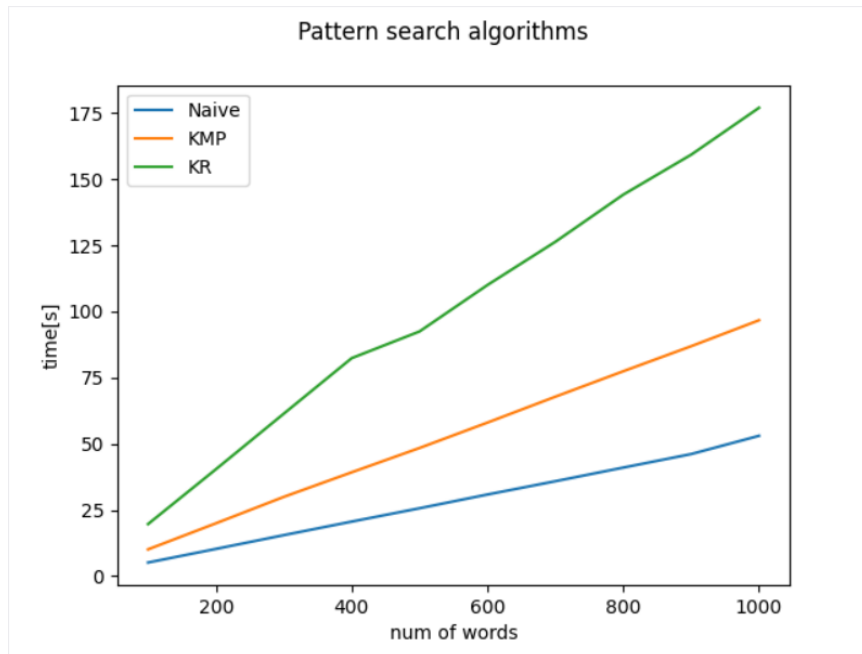
- gdy wzory lub tekst są puste
- gdy wzory lub tekst są równe
- gdy jeden jest dłuższy od drugiego
- szukanie wielu kluczy



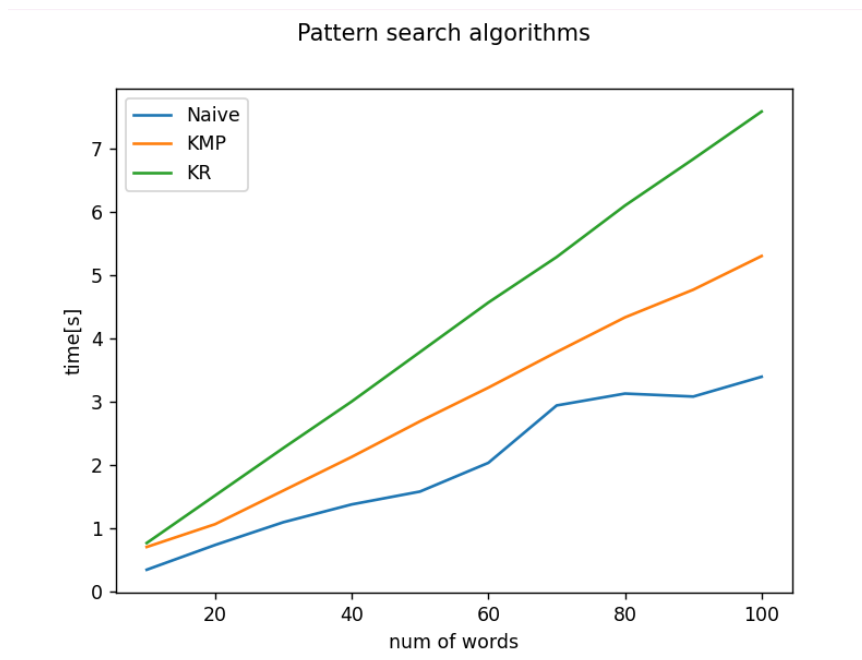
### **plik main.py**

Plik zawierający kod odpowiedzialny za tworzenie wykresów, dla każdego algorytmu mierzy jego czas działania dla  $n = 100, 200, \dots, 1000$

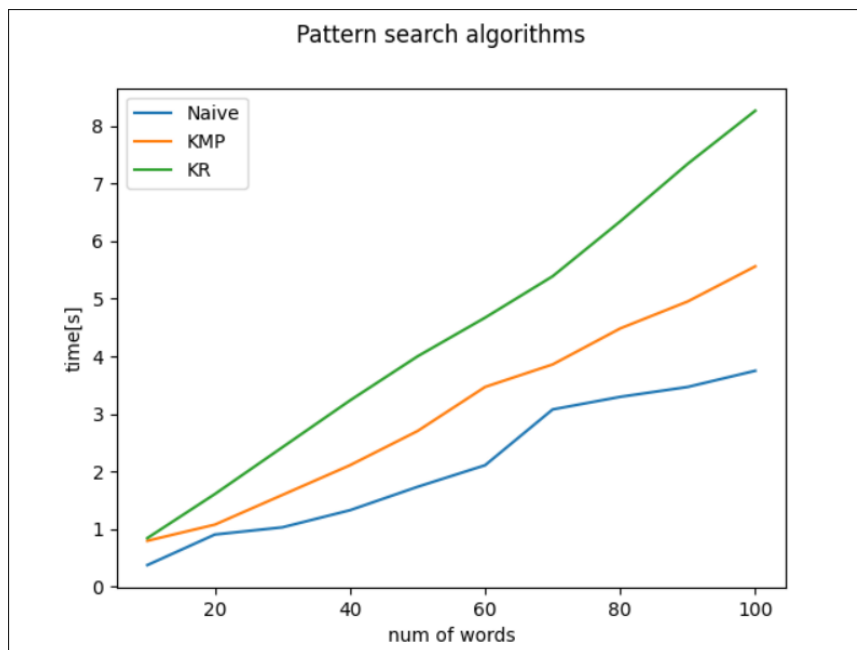
# Wykresy



Dodatkowo sprawdziliśmy działanie algorytmu KR dla popularnej funkcji haszującej:



Oraz dla samodzielnie wymyślonej, prostej funkcji haszującej:



## Wnioski

W teorii algorytm naiwny powinien być najwolniejszy, zaraz po nim algorytm KR i najszybszy KMP. Natomiast w praktyce okazuje się że to algorytm naiwny jest najszybszy, a KR najwolniejszy. Jest to prawdopodobnie spowodowane tym że długość wzorca  $m$  prawie zawsze wynosi mniej niż 10, dodatkowo zawsze szukamy słów, które rzadko się pojawiają kilkakrotnie.