

# Wyszukiwanie wzorca w tekście (1)

---

dr inż. Marcin Ciura

[mgc@agh.edu.pl](mailto:mgc@agh.edu.pl)

Wydział Informatyki, Akademia Górniczo-Hutnicza

# Plan wykładu

- Zasady wyzwań
- Definicje
- Algorytm naiwny
- Funkcja Z
- Algorytm Boyera-Moore'a
- Algorytm Knutha-Morrisa-Pratta
- Algorytm Karpa-Rabina

## Zasady wyzwania

---

Wyzwanie to taki problem, który się nie mieści na wykładzie, a jest ciekawy

Te osoby, które rozwiążą najwięcej wyzwań, dostaną nagrody

# Zasady rozwiązywania wyzwań

1. Nie muszę chodzić na zajęcia ścieżki algorytmicznej
2. Nawet nie muszę studiować informatyki :-)
3. Mogę rozmawiać z innymi osobami o tym, jak rozwiązać wyzwanie
4. Jeśli rozmowa z jakąś osobą pomoże mi rozwiązać wyzwanie, to napiszę w rozwiązaniu imię i nazwisko tej osoby
5. Jeśli rozmowa ze sztuczną inteligencją pomoże mi rozwiązać wyzwanie, to zamieszczę w rozwiązaniu treść tej rozmowy
6. Zapiszę rozwiązanie w prywatnym repozytorium na GitHubie i zaproszę użytkownika MarcinCiura do tego repozytorium
7. Zgłoszę, że rozwiązałem wyzwanie, do ostatniej niedzieli przed 7. wykładem

## Definicje

---

# Definicje

- **Alfabet** – skończony zbiór **symboli** czyli **znaków**
- **łańcuch znaków**, krócej **łańcuch** – każdy skończony ciąg znaków
- **Długość łańcucha** – liczba znaków w tym łańcuchu. Oznaczam długość łańcucha, otaczając ten łańcuch kreskami pionowymi. Na przykład  $|Ala\_ma\_kota| = 11$

# Definicje

- **łańcuch pusty** – łańcuch o długości zero
- **Prefiks** łańcucha **s** – każdy łańcuch **s[:i]**, gdzie  $0 \leq i \leq |s|$
- **Sufiks** łańcucha **s** – każdy łańcuch **s[i:]**, gdzie  $0 \leq i \leq |s|$
- **Prefiks/sufiks właściwy** – taki prefiks/sufiks, który nie jest łańcuchem pustym
- **Podłańcuch** łańcucha **s** – każdy łańcuch **s[i:j]**, gdzie  $0 \leq i \leq j \leq |s|$



## Wyszukiwanie wzorca w tekście

Dane są dwa łańcuchy: **wzorzec** i **tekst**. Nazywam 2 zmienne, które przechowują te 2 łańcuchy, odpowiednio: **pat** i **text**

Chcę znaleźć wszystkie wystąpienia wzorca w tekście, czyli wszystkie takie pozycje **i** w tekście, że podłańcuch tekstu, który zaczyna się na pozycji **i**, jest równy wzorcowi

Innymi słowy chcę znaleźć wszystkie takie pozycje **i**, że  
`slices.Equal(text[i:i+len(pat)], pat)`

Różne wystąpienia wzorca w tekście mogą się nakładać

# Algorytm naiwny

---

## Algorytm naiwny

Najpierw porównuję ze wzorcem `pat` wycinek `text[0:len(pat)]`

Potem porównuję ze wzorcem `pat` kolejne wycinki `text[i:i+len(pat)]`  
tak długo, dopóki `i + len(pat) <= len(text)`

Jeśli te 2 wycinki są równe, to podaję, że wzorzec `pat` występuje w tekście  
`text` od pozycji `i`

## Algorytm naiwny

Kiedy porównuję wycinek **s** ze wzorcem **pat**, kolejno porównuję odpowiednie znaki wycinka **s** i wzorca **pat** tak długo, aż:

- znajdę 2 różne znaki **s[j] != pat[j]**
- lub dojdę do końca wzorca **pat**

Te 2 przypadki oznaczają odpowiednio, że:

- wycinek **s** i wzorec **pat** są **różne**
- wycinek **s** i wzorec **pat** są **równe**

## Algorytm naiwny – przykład 1

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

porównań znaków: 0

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.  
dźwiedź

porównań znaków: 1

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

porównań znaków: 2



## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 3

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 4

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 5

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 6

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 7

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 8

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 9

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 10



## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 11

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 12

## Algorytm naiwny – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 13

## Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

porównań znaków: 14

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

porównań znaków: 15

## Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

porównań znaków: 16

## Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

porównań znaków: 17

## Algorytm naiwny – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 18



# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

porównań znaków: 19

## Algorytm naiwny – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 20

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź

dźwiędź

dźwiędź

porównań znaków: 21

# Algorytm naiwny – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

porównań znaków: 22

# Algorytm naiwny – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

porównań znaków: 23

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

porównań znaków: 24

# Algorytm naiwny – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 25

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

porównań znaków: 26



# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

porównań znaków: 27

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

porównań znaków: 28

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 29

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź dźwiędź

porównań znaków: 30

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 31

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 32

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 33

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 34



# Algorytm naiwny – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

porównań znaków: 35

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 36

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 37

# Algorytm naiwny – przykład 1

dźwiędź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

porównań znaków: 38

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 39

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 40

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 41

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 42



# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 43

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 44

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 45

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 46

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 47

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 48

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 49

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 50



# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 51

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 52

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 53

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 54

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 55

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 56

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 57

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 58



# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 59

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 60

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 61

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 62

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 63

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 64

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 65

# Algorytm naiwny – przykład 1

dźwiędź

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 66



porównań znaków: 66



## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

porównań znaków: 0

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB

porównań znaków: 1

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB

BBBBB

porównań znaków: 2

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB

BBBBB

BBBBB

porównań znaków: 3

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB

BBBBB

BBBBB

BBBBB

porównań znaków: 4

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB

BBBBB

BBBBB

BBBBB

BBBBB

porównań znaków: 5



## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB

BBBBB

BBBBB

BBBBB

BBBBB

BBBBB

porównań znaków: 6

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB BBBBB

BBBBB

BBBBB

BBBBB

BBBBB

BBBBB

porównań znaków: 7

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB BBBBB

BBBBB BBBBB

BBBBB

BBBBB

BBBBB

BBBBB

porównań znaków: 8

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB

BBBBB

BBBBB

porównań znaków: 9

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB

BBBBB

porównań znaków: 10

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB

porównań znaków: 11

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

BBBBB BBBBB

porównań znaków: 12

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

B B B B B B B B B B B B B B B B

B B B B B B B B B B B B B B B B

B B B B B B B B B B B B B B B B

B B B B B B B B B B B B B B B B

B B B B B B B B B B B B B B B B

B B B B B B B B B B B B B B B B

porównań znaków: 13



## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

porównań znaków: 14

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

porównań znaków: 15

## Algorytm naiwny – przykład 2

AAAAAAAAAAAAAAAAAAAA

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

B B B B B B B B B B B

porównań znaków: 16

porównań znaków: 16

## Wyzwanie 1

Udowodnię, że liczba porównań, które wykonuje algorytm naiwny, nigdy nie jest mniejsza niż  $|text| - |pat| + 1$



## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

porównań znaków: 0

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

A

porównań znaków: 1



## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

AA AAA

porównań znaków: 2

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

AAAAA

porównań znaków: 3

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

AAAAA

porównań znaków: 4

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

AAAAA

porównań znaków: 5

## Algorytm naiwny – przykład 3

AAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

porównań znaków: 6

## Algorytm naiwny – przykład 3

AAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

porównań znaków: 7

## Algorytm naiwny – przykład 3

AAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

porównań znaków: 8

## Algorytm naiwny – przykład 3

AAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

porównań znaków: 9



## Algorytm naiwny – przykład 3

AAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

porównań znaków: 10

## Algorytm naiwny – przykład 3

AAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 11

## Algorytm naiwny – przykład 3

AAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 12

## Algorytm naiwny – przykład 3

AAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 13

## Algorytm naiwny – przykład 3

AAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 14

## Algorytm naiwny – przykład 3

AAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 15

## Algorytm naiwny – przykład 3

AAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

A

porównań znaków: 16

## Algorytm naiwny – przykład 3

AAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 17



## Algorytm naiwny – przykład 3

AAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 18

## Algorytm naiwny – przykład 3

AAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 19

## Algorytm naiwny – przykład 3

AAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 20

## Algorytm naiwny – przykład 3

AAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

A AAAA

porównań znaków: 21

## Algorytm naiwny – przykład 3

AAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 22

## Algorytm naiwny – przykład 3

AAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 23

## Algorytm naiwny – przykład 3

AAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 24

## Algorytm naiwny – przykład 3

AAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 25



## Algorytm naiwny – przykład 3

AAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

A

porównań znaków: 26

## Algorytm naiwny – przykład 3

AAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AA~~AAA~~

porównań znaków: 27

## Algorytm naiwny – przykład 3

AAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 28

## Algorytm naiwny – przykład 3

AAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 29

## Algorytm naiwny – przykład 3

AAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 30

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 31

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 32

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 33



## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 34

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 35

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 36

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 37

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 38

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 39

## Algorytm naiwny – przykład 3

AAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 40

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 41



## Algorytm naiwny – przykład 3

AAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 42

## Algorytm naiwny – przykład 3

AAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 43

## Algorytm naiwny – przykład 3

AAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 44

## Algorytm naiwny – przykład 3

AAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

AAAAA

porównań znaków: 45

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

porównań znaków: 46

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

porównań znaków: 47

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAA

AAAAA

AAAAA

porównań znaków: 48

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

porównań znaków: 49



## Algorytm naiwny – przykład 3

AAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

AAAAA

porównań znaków: 50

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

porównań znaków: 51

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAA

AAAAA

porównań znaków: 52

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAA

AAAAA

porównań znaków: 53

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

porównań znaków: 54

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA

porównań znaków: 55

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 56

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAA

porównań znaków: 57



## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAA

porównań znaków: 58

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 59

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 60

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 61

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAA

AAAAA AAAAA AA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 62

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 63

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 64

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 65



## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 66

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 67

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 68

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 69

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 70

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 71

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 72

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 73



## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 74

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 75

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 76

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 77

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 78

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 79

## Algorytm naiwny – przykład 3

AAAAAAAAAAAAAAAAAAAAA

AAAAAAAAAAAAAAAAAAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA AAAAA

AAAAA AAAAA

AAAAA AAAAA

porównań znaków: 80

porównań znaków: 80



## Wyzwanie 2

Udowodnię, że liczba porównań, które wykonuje algorytm naiwny, nigdy nie jest większa niż  $|pat| (|text| - |pat| + 1)$

## Algorytm naiwny – kod

```
// Naive wywołuje funkcję `output(i)` dla każdego takiego  
// indeksu `i`, że `slices.Equal(text[i:i+len(pat)], pat)`  
func Naive(pat, text []byte, output func(int)) {  
    for i := 0; i+len(pat) <= len(text); i++ {  
        if slices.Equal(text[i:i+len(pat)], pat) {  
            output(i)  
        }  
    }  
}
```

Wywołanie funkcji

```
slices.Equal(text[i:i+len(pat)], pat)
```

można zastąpić wywołaniem funkcji

```
hasPrefix(text[i:], pat)
```

## Algorytm naiwny – kod

```
// hasPrefix zwraca `true`, jeśli  
// `slices.Equal(s[:len(pat)], pat)`  
func hasPrefix(s, pat []byte) bool {  
    for j := 0; j < len(pat); j++ {  
        if s[j] != pat[j] {  
            return false  
        }  
    }  
    return true  
}
```

## Algorytm naiwny – kod

```
// Naive wywołuje funkcję `output(i)` dla każdego takiego  
// indeksu `i`, że `slices.Equal(text[i:i+len(pat)], pat)`  
func Naive(pat, text []byte, output func(int)) {  
    for i := 0; i+len(pat) <= len(text); i++ {  
        if hasPrefix(text[i:], pat) {  
            output(i)  
        }  
    }  
}
```

Algorytm naiwny:

- korzysta z  $O(1)$  dodatkowej pamięci na zmienne  $i$  i  $j$
- w najlepszym przypadku wykonuje  $O(|\text{text}|)$  porównań znaków
- w najgorszym przypadku wykonuje  $O(|\text{text}||\text{pat}|)$  porównań znaków

**Jak przyspieszyć algorytm naiwny?**

---

## Jak przyspieszyć algorytm naiwny?

Tak można przyspieszyć algorytm naiwny:

- można w rozsądnym czasie utworzyć pewną strukturę danych  $D$  na podstawie wzorca **pat** lub tekstu **text**
- jeśli odpowiadające sobie znaki wycinka tekstu **text[i:]** i wzorca **pat** są różne, to w kolejnym kroku porównać ze wzorcem **pat** niekoniecznie następny wycinek tekstu **text[i+1:]**, tylko wycinek **text[i+k:]**
  - czyli **przesunąć** wzorzec **pat** o  $k$  pozycji
- liczba pozycji  $k$ , o które przesuniemy wzorzec, zależy od zawartości struktury danych  $D$



## Funkcja $Z$ (Gusfield)

### Definicja:

Dla danego łańcucha  $s$  i pozycji  $k > 0$

$Z_k(s)$  oznacza długość najdłuższego takiego prefiksu łańcucha  $s[k:]$ , który jest taki sam, jak pewien prefiks łańcucha  $s$

### Przykład:

$k$	0	1	2	3	4	5	6	7
$s$	i	n	d	i	a	n	i	n
$Z_k(s)$	0	0	1	0	0	2	0	

## Funkcja Z (Gusfield)

```
// lenOfCommonPrefix zwraca długość najdłuższego  
// wspólnego prefiksu łańcuchów `s` i `t`  
func lenOfCommonPrefix(s, t []byte) int {  
    k := 0  
    for ; k < min(len(s), len(t)); k++ {  
        if s[k] != t[k] {  
            return k  
        }  
    }  
    return k  
}
```

## Funkcja Z (Gusfield)

*// SimpleZFunction zwraca wycinek. `k`-ty element tego  
// wycinka jest równy długości najdłuższego takiego prefiksu  
// łańcucha `s[k:]`, który jest równy pewnemu prefiksowi  
// łańcucha `s`*

```
func SimpleZFunction(s []byte) []int {  
    z := make([]int, len(s))  
    for k := 1; k < len(s); k++ {  
        z[k] = lenOfCommonPrefix(s[k:], s)  
    }  
    return z  
}
```

# Funkcja Z (Gusfield)

```
// Preprocess zwraca wycinek. `k`-ty element tego wycinka
// jest równy długości najdłuższego takiego prefiksu
// łańcucha `s[k:]`, który jest równy pewnemu prefiksowi
// łańcucha `s`
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-l] >= r-k {
            z[k] = r - k + lenOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-l]
        }
        // bytes.HasPrefix(s, s[k:k+z[k]])
        // bytes.HasPrefix(s, s[l:r])
    }
    return z
}
```

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-1] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-1]
        }
    }
    return z
}
```

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-1] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-1]
        }
    }
    return z
}
```

$$n \begin{matrix} k \\ i \end{matrix} e n a p e \begin{matrix} k \\ i \end{matrix} n i e n i e$$
  
$$[0]$$

# Funkcja Z – przykład

```
func ZFunction(s []byte) []int {  
    z := make([]int, len(s))  
    l := 0  
    r := 0  
    for k := 1; k < len(s); k++ {  
        if k >= r {  
            z[k] = lenghtOfCommonPrefix(s[k:], s)  
            if z[k] > 0 {  
                l = k  
                r = k + z[k]  
            }  
        } else if z[k-1] >= r-k {  
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])  
            l = k  
            r = k + z[k]  
        } else {  
            z[k] = z[k-1]  
        }  
    }  
    return z  
}
```

n i e n a p e ł n i e n i e  
0 0  
[]

# Funkcja Z – przykład

```
func ZFunction(s []byte) []int {  
    z := make([]int, len(s))  
    l := 0  
    r := 0  
    for k := 1; k < len(s); k++ {  
        if k >= r {  
            z[k] = lenghtOfCommonPrefix(s[k:], s)  
            if z[k] > 0 {  
                l = k  
                r = k + z[k]  
            }  
        } else if z[k-1] >= r-k {  
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])  
            l = k  
            r = k + z[k]  
        } else {  
            z[k] = z[k-1]  
        }  
    }  
    return z  
}
```

n i e n a p e ł n i e n i e  
0 0 1  
[ ]



## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {  
    z := make([]int, len(s))  
    l := 0  
    r := 0  
    for k := 1; k < len(s); k++ {  
        if k >= r {  
            z[k] = lenghtOfCommonPrefix(s[k:], s)  
            if z[k] > 0 {  
                l = k  
                r = k + z[k]  
            }  
        } else if z[k-1] >= r-k {  
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])  
            l = k  
            r = k + z[k]  
        } else {  
            z[k] = z[k-1]  
        }  
    }  
    return z  
}
```

n i e n a p e l n i e n i e  
0 0 1 0  
[ ]

# Funkcja Z – przykład

```
func ZFunction(s []byte) []int {  
    z := make([]int, len(s))  
    l := 0  
    r := 0  
    for k := 1; k < len(s); k++ {  
        if k >= r {  
            z[k] = lenghtOfCommonPrefix(s[k:], s)  
            if z[k] > 0 {  
                l = k  
                r = k + z[k]  
            }  
        } else if z[k-1] >= r-k {  
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])  
            l = k  
            r = k + z[k]  
        } else {  
            z[k] = z[k-1]  
        }  
    }  
    return z  
}
```

n i e n a p e l n i e n i e  
0 0 1 0 0  
[ ]

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {  
    z := make([]int, len(s))  
    l := 0  
    r := 0  
    for k := 1; k < len(s); k++ {  
        if k >= r {  
            z[k] = lenghtOfCommonPrefix(s[k:], s)  
            if z[k] > 0 {  
                l = k  
                r = k + z[k]  
            }  
        } else if z[k-1] >= r-k {  
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])  
            l = k  
            r = k + z[k]  
        } else {  
            z[k] = z[k-1]  
        }  
    }  
    return z  
}
```

n i e n a p e l n i e n i e  
0 0 1 0 0 0  
[ ]

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-1] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-1]
        }
    }
    return z
}
```

n i e n a p e i n i e n i e

0 0 1 0 0 0 0

[ ]

k

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-1] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-1]
        }
    }
    return z
}
```

n	i	e	n	a	p	e	l	n	i	e	n	i	e
0	0	1	0	0	0	0	4						
							[					]	

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-1] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-1]
        }
    }
    return z
}
```

n i e n a p e l n i e n i e  
0 0 1 0 0 0 0 4 0  
[ ]

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-1] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-1]
        }
    }
    return z
}
```

n i e n a p e l n i e n i e  
0 0 1 0 0 0 0 4 0 0  
[ ]

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-1] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-1]
        }
    }
    return z
}
```

```

n i e n a p e l n i e n i e
0 0 1 0 0 0 0 4 0 0 3
                        [      ]
```



## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-1] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-1]
        }
    }
    return z
}
```

n i e n a p e l n i e n i e  
0 0 1 0 0 0 0 4 0 0 3 0  
[ ]

## Funkcja Z – przykład

```
func ZFunction(s []byte) []int {
    z := make([]int, len(s))
    l := 0
    r := 0
    for k := 1; k < len(s); k++ {
        if k >= r {
            z[k] = lenghtOfCommonPrefix(s[k:], s)
            if z[k] > 0 {
                l = k
                r = k + z[k]
            }
        } else if z[k-l] >= r-k {
            z[k] = r - k + lenghtOfCommonPrefix(s[r:], s[r-k:])
            l = k
            r = k + z[k]
        } else {
            z[k] = z[k-l]
        }
    }
    return z
}
```

$$\begin{bmatrix} n & i & e & n & a & p & e & l & n & i & e & n & i & e & k \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 3 & 0 & 0 \end{bmatrix}$$

## Funkcja $Z$ (Gusfield)

Gusfield dowodzi na stronach 9–10, że funkcja **ZFunction** zwraca poprawne wartości  $Z_k(s)$  i że ta funkcja działa w czasie liniowym względem długości łańcucha **s**

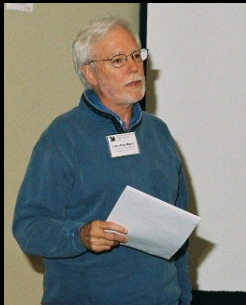
# Algorytm Boyera-Moore'a

---



Profesor informatyki, matematyki i filozofii na Uniwersytecie Teksasu w Austin. W latach 1974–1977 wraz z J Strotherem Moore'em opracował szybki algorytm wyszukiwania wzorca w tekście.

## J Strother Moore



Amerykański informatyk. Ma na pierwsze imię „J”. Był wspinaczem skałkowym.

## Algorytm Boyera-Moore'a – 3 pomysły

3 pomysły na to, jak przyspieszyć algorytm naiwny:

1. Porównywać łańcuchy wspak
2. Reguła niezgodności
3. Reguła dobrego sufiksu

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak

Porównuję znaki łańcuchów `text[i:i+len(pat)]` i `pat` od końca



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak

```
// backwardHasPrefix zwraca `true`, jeśli
// `slices.Equal(s[:len(pat)], pat)`
func backwardHasPrefix(s, pat []byte) bool {
    for j := len(pat) - 1; j >= 0; j-- {
        if s[j] != pat[j] {
            return false
        }
    }
    return true
}
```

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak

```
// BackwardNaive wywołuje `output(i)` dla każdego takiego `i`,  
// że `slices.Equal(text[i:i+len(pat)], pat)`  
func BackwardNaive(pat, text []byte, output func(int)) {  
    for i := 0; i+len(pat) <= len(text); i++ {  
        if backwardHasPrefix(text[i:], pat) {  
            output(i)  
        }  
    }  
}
```

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

porównań znaków: 0

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.  
dźwiedź

porównań znaków: 1

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

porównań znaków: 2

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

porównań znaków: 3

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwi**edź**

porównań znaków: 4



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

porównań znaków: 5

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

porównań znaków: 6

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 7

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 8

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 9

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 10

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 11

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 12



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 13

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 14

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 15

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 16

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 17

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 18

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 19

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 20



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź

dźwiedź dźwiędź

dźwiędź dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

porównań znaków: 21

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 22

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

porównań znaków: 23

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

porównań znaków: 24

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

porównań znaków: 25

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 26

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 27

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 28



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 29

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 30

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 31

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 32

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 33

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 34

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

porównań znaków: 35

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiedź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiedź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 36



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

porównań znaków: 37

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

porównań znaków: 38

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

porównań znaków: 39

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

porównań znaków: 40

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

porównań znaków: 41

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

porównań znaków: 42

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 43

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 44



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 45

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 46

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 47

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 48

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

porównań znaków: 49

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 50

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 51

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 52



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 53

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 54

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 55

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 56

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 57

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 58

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 59

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 60



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 61

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

porównań znaków: 62

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 63

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiedź

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź dźwiedź dźwiedź

porównań znaków: 64

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 65

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 66

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 67

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 68



## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 69

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

dźwiędź

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

porównań znaków: 70

## Algorytm Boyera-Moore'a – porównywać łańcuchy wspak – przykład

porównań znaków: 70

## Algorytm Boyera-Moore'a – reguła niezgodności

Jeśli znak tekstu jest różny niż znak wzorca, czyli  $\text{text}[i] \neq \text{pat}[j]$ ,  
to przesuwam wzorzec  $\text{pat}$  w prawo tak, żeby:

- ostatnie wystąpienie znaku  $\text{text}[i]$  we wzorcu  $\text{pat}$  odpowiadało  $i$ -temu znakowi tekstu
- lub – jeśli wzorzec  $\text{pat}$  wcale nie zawiera znaku  $\text{text}[i]$  – przesuwam wzorzec  $\text{pat}$  w prawo tak, żeby zaczynał się przy  $i+1$ -szym znaku tekstu

## Algorytm Boyera-Moore'a – reguła niezgodności

```
// findLastOccurrences zwraca mapę, która:  
// + odwzorowuje wszystkie takie znaki, które występują  
//   w łańcuchu `s` na `k+1`, gdzie `k` to indeks ostatniego  
//   wystąpienia danego znaku w `s`  
// + odwzorowuje wszystkie takie znaki, które nie występują  
//   w łańcuchu `s`, na 0
```

```
func findLastOccurrences(s []byte) map[byte]int {  
    lastOccurrences := map[byte]int{}  
    for k, c := range s {  
        lastOccurrences[c] = k + 1  
    }  
    return lastOccurrences  
}
```

## Algorytm Boyera-Moore'a – reguła niezgodności

*// BoyerMooreHorspoolHasPrefix zwraca parę (R, S). R ma wartość `true`,  
// jeśli `slices.Equal(text[:len(pat)], pat)`; S określa, o ile  
// pozycji w prawo należy przesunąć wzorzec `pat` w następnej iteracji  
// algorytmu Boyera-Moore'a*

```
func boyerMooreHorspoolHasPrefix(  
    pat, text []byte, lastOccurrences map[byte]int) (bool, int) {  
    for j := len(pat) - 1; j >= 0; j-- {  
        if text[j] != pat[j] {  
            return false, max(j+1-lastOccurrences[text[j]], 1)  
        }  
    }  
    return true, 1  
}
```

## Algorytm Boyera-Moore'a – reguła niezgodności

```
// BoyerMooreHorspool wywołuje `output(i)` dla każdego takiego `i`,  
// że `slices.Equal(text[i:i+len(pat)], pat)`
```

```
func BoyerMooreHorspool(pat, text []byte, output func(int)) {  
    lastOccurrences := findLastOccurrences(pat)  
    for i := 0; i+len(pat) <= len(text); /**/ {  
        found, shift := boyerMooreHorspoolHasPrefix(  
            pat, text[i:], lastOccurrences)  
        if found {  
            output(i)  
        }  
        i += shift  
    }  
}
```

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład



## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 0

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

$\max(7 - \text{lastOccurrences}['d'], 1) = 1$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 1

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 2

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 3

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwi**edź**

d   ż   w   i   e   \*

6 7 3 4 5 0

porównań znaków: 4

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 5

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

$\max(3 - \text{lastOccurrences}['n'], 1) = 3$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 6

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

$\max(7 - \text{lastOccurrences}['e'], 1) = 2$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 7



## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 8

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 9

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 10

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 11

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 12

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 13

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

1

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 14

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

$\max(7 - \text{lastOccurrences}['\ '], 1) = 7$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 15



## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź            dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

$\max(7 - \text{lastOccurrences}['\acute{z}], 1) = 7$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 16

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź            dźwiedź

dźwiedź            dźwiedź

dźwiedź

dźwiedź

dźwiedź

$\max(7 - \text{lastOccurrences}['e'], 1) = 2$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 17

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 18

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 19

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź    dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   \*

6 7 3 4 5 0      porównań znaków: 20

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź    dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   \*

6 7 3 4 5 0      porównań znaków: 21

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 22

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 23



## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

dźwiedź

1

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 24

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

$\max(7 - \text{lastOccurrences}['?'], 1) = 7$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 25

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

$\max(7 - \text{lastOccurrences}[' '], 1) = 7$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 26

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

dźwiedź

$\max(7 - \text{lastOccurrences}['w'], 1) = 4$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 27

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

6 7 3 4 5 0

porównań znaków: 28

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

6 7 3 4 5 0

porównań znaków: 29

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

6 7 3 4 5 0

porównań znaków: 30

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

6 7 3 4 5 0

porównań znaków: 31



## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiedź

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

porównań znaków: 32

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

6 7 3 4 5 0

porównań znaków: 33

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

1

d z w i e \*

6 7 3 4 5 0

porównań znaków: 34

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

dźwiędź                      dźwiędź                      dźwiędź  
To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.  
dźwiędź                      dźwiędź dźwiędź dźwiędź                      dźwiędź  
dźwiędź                      dźwiędź                      dźwiędź  
dźwiędź                      dźwiędź                      dźwiędź  
dźwiędź                      dźwiędź                      dźwiędź  
dźwiędź  
dźwiędź

$\max(7 - \text{lastOccurrences}['.'], 1) = 7$

d ź w i e \*

6 7 3 4 5 0      porównań znaków: 35

## Algorytm Boyera-Moore'a – reguła niezgodności – przykład

porównań znaków: 35

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu

Jeśli pewien podłańcuch tekstu  $\text{text}[i+1:i+k+1]$  i pewien sufiks wzorca  $\text{pat}[j+1:]$  są równe, a poprzedni znak tekstu jest różny od poprzedniego znaku wzorca, czyli  $\text{text}[i] \neq \text{pat}[j]$ , to przesuwam wzorzec  $\text{pat}$  w prawo tak, żeby:

- inny podłańcuch wzorca  $\text{pat}[1+1:1+k+1]$  był równy łańcuchowi  $\text{text}[i+1:i+k+1]$ , a poprzednie znaki tekstu i wzorca były różne:  $\text{text}[i] \neq \text{pat}[1]$
- lub pewien prefiks wzorca  $\text{pat}[:1]$  był równy pewnemu sufiksowi łańcucha  $\text{text}[i+1:i+k+1]$
- przy czym prefiks wzorca  $\text{pat}[:1]$  może być łańcuchem pustym – wtedy przesuwam wzorzec  $\text{pat}$  w prawo tak, żeby zaczynał się przy  $i+1$ -szym znaku tekstu

Wybieram ten podpunkt, w którym przesuwam wzorzec o mniej pozycji

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu

```
func simpleFindShiftOfSuffix(s []byte, i int) int {  
    for k := 1; k <= i; k++ {  
        if s[i-k] != s[i] &&  
            slices.Equal(s[i+1:], s[i-k+1:len(s)-k]) {  
            return k  
        }  
    }  
    for k := i + 1; k < len(s); k++ {  
        if slices.Equal(s[k:], s[:len(s)-k]) {  
            return k  
        }  
    }  
    return len(s)  
}
```

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

xxxxxxx

dźwiędź

dźwiędź

1



## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

xxxxxxz

dźwiedz

dźwiedz

71

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

xxxxxdź

dźwiedź

dźwiedź

571

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

xxxxedź

dźwiedź

dźwiedź

5571

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

xxxiedź

dźwiedź

dźwiedź

55571

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

xxwiedź

dźwiedź

dźwiedź

555571

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

xżwiedź

dżwiedź

      dżwiedź

5555571

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

XXXXXXXX

OWOCOWO

OWOCOWO

1

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

XXXXXXO

OWOCOWO

OWOCOWO

21



## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

XXXXXWO

OWOCOWO

OWOCOWO

621

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiedź

5555571

XXXXOWO

OWOCOWO

OWOCOWO

4621

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiedź

5555571

XXXCOWO

OWOCOWO

OWOCOWO

44621

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiedź

5555571

XXOCOWO

OWOCOWO

OWOCOWO

444621

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

XWOCOWO

OWOCOWO

OWOCOWO

4444621

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

OWOCOWO

4444621

xxxxx

eeeeee

eeeeee

5

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

OWOCOWO

4444621

xxxxe

eeeee

      eeeee

      45

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

OWOCOWO

4444621

xxxee

eeee

eeee

345



## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

owocowo

4444621

xxeee

eeeee

eeeee

2345

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

OWOCOWO

4444621

xeeee

eeee

eeee

12345

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu – przykład

dźwiędź

5555571

OWOCOWO

4444621

eeeeee

12345

## Algorytm Boyera-Moore'a – reguła dobrego sufiksu

```
func simpleComputeGoodSuffixes(s []byte) []int {  
    r := make([]int, len(s))  
    for i := range s {  
        r[i] = simpleFindShiftOfSuffix(s, i)  
    }  
    return r  
}
```

## Wyzwanie 3

Pesymistyczna złożoność czasowa funkcji `simpleComputeGoodSuffixes` jest większa niż  $O(|pat|)$

Gusfield podaje na stronach 20–21 i 31–33 takie algorytmy wypełniania tablicy dobrych sufiksów, które działają w czasie liniowym względem długości wzorca

Zapiszę co najmniej jeden z tych algorytmów w Go. Będę pamiętać o testach :-)

## Algorytm Boyera-Moore'a

```
// boyerMooreHasPrefix zwraca parę (R, S). R ma wartość `true`,  
// jeśli `slices.Equal(text[:len(pat)], pat)`; S określa,  
// o ile pozycji w prawo należy przesunąć wzorzec `pat`  
func boyerMooreHasPrefix(text, pat []byte,  
    lastOccurrences map[byte]int, goodSuffixes []int) (bool, int) {  
    for j := len(pat) - 1; j >= 0; j-- {  
        if text[j] != pat[j] {  
            return false, max(j+1 - lastOccurrences[text[j]],  
                                goodSuffixes[j])  
        }  
    }  
    return true, goodSuffixes[0]  
}
```

## Algorytm Boyera-Moore'a

```
// BoyerMoore wywołuje `output(i)` dla każdego takiego `i`,  
// że `slices.Equal(text[i:i+len(pat)], pat)`
```

```
func BoyerMoore(pat, text []byte, output func(int)) {  
    lastOccurrences := findLastOccurrences(pat)  
    goodSuffixes := simpleComputeGoodSuffixes(pat)  
    for i := 0; i+len(pat) <= len(text); /**/ {  
        found, shift := boyerMooreHasPrefix(text[i:], pat,  
            lastOccurrences, goodSuffixes)  
        if found {  
            output(i)  
        }  
        i += shift  
    }  
}
```

## Algorytm Boyera-Moore'a – przykład



## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 0

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.  
dźwiedź

7-lastOccurrences['d'] = 1      goodSuffixes[6] = 1  
d ż w i e \*      d ż w i e d ż  
6 7 3 4 5 0      5 5 5 5 5 7 1      porównań znaków: 1

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 2

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 3

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwi**edź**

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 4

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 5

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

3-lastOccurrences['n'] = 3      goodSuffixes[2] = 5

d ź w i e \*      d ź w i e d ź

6 7 3 4 5 0      5 5 5 5 5 7 1      porównań znaków: 6

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 7



## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 8

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 9

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 10

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 11

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 12

## Algorytm Boyera-Moore'a – przykład

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

goodSuffixes[0] = 5

d ź w i e \*

d ź w i e d ź

6 7 3 4 5 0

5 5 5 5 5 7 1

porównań znaków: 13

## Algorytm Boyera-Moore'a – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź

dźwiedź

7-lastOccurrences[' '] = 7      goodSuffixes[6] = 1

d ź w i e \*      d ź w i e d ź

6 7 3 4 5 0      5 5 5 5 5 7 1      porównań znaków: 14

## Algorytm Boyera-Moore'a – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 15



## Algorytm Boyera-Moore'a – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 16

## Algorytm Boyera-Moore'a – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

5-lastOccurrences[' '] = 5      goodSuffixes[4] = 5

d ź w i e \*      d ź w i e d ź

6 7 3 4 5 0      5 5 5 5 5 7 1      porównań znaków: 17

## Algorytm Boyera-Moore'a – przykład

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiedź dźwiedź

dźwiędź

d ż w i e \*

6 7 3 4 5 0

d ż w i e d ź

5 5 5 5 5 7 1

porównań znaków: 18

## Algorytm Boyera-Moore'a – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 19

## Algorytm Boyera-Moore'a – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 20

## Algorytm Boyera-Moore'a – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 21

## Algorytm Boyera-Moore'a – przykład

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź dźwiedź dźwiedź

dźwiedź dźwiedź

dźwiedź

d ź w i e \*

6 7 3 4 5 0

d ź w i e d ź

5 5 5 5 5 7 1

porównań znaków: 22

## Algorytm Boyera-Moore'a – przykład

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź

dźwiędź

d ż w i e \*

6 7 3 4 5 0

d ż w i e d ź

5 5 5 5 5 7 1

porównań znaków: 23



## Algorytm Boyera-Moore'a – przykład

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiedź dźwiedź

dźwiędź

goodSuffixes[0] = 5

d ż w i e \*

d ż w i e d ż

6 7 3 4 5 0

5 5 5 5 5 7 1

porównań znaków: 24

## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź

7-lastOccurrences['y'] = 7      goodSuffixes[6] = 1

d ź w i e \*      d ź w i e d ź

6 7 3 4 5 0      5 5 5 5 5 7 1      porównań znaków: 25

## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź

7-lastOccurrences[' '] = 7      goodSuffixes[6] = 1

d ź w i e \*      d ź w i e d ź

6 7 3 4 5 0      5 5 5 5 5 7 1      porównań znaków: 26

## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

d z w i e d ź

6 7 3 4 5 0

5 5 5 5 5 7 1

porównań znaków: 27

## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

d z w i e d ź

6 7 3 4 5 0

5 5 5 5 5 7 1

porównań znaków: 28

## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

d z w i e d ź

6 7 3 4 5 0

5 5 5 5 5 7 1

porównań znaków: 29

## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

d z w i e d ź

6 7 3 4 5 0

5 5 5 5 5 7 1

porównań znaków: 30

## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

6 7 3 4 5 0

d z w i e d ź

5 5 5 5 5 7 1

porównań znaków: 31



## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e \*

d z w i e d ź

6 7 3 4 5 0

5 5 5 5 5 7 1

porównań znaków: 32

## Algorytm Boyera-Moore'a – przykład

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

goodSuffixes[0] = 5

d z w i e \*

d z w i e d ź

6 7 3 4 5 0

5 5 5 5 5 7 1

porównań znaków: 33

## Algorytm Boyera-Moore'a – przykład

porównań znaków: 33

## Algorytm Boyera-Moore'a – najgorszy przypadek

Richard Cole udowodnił, że gdy tekst nie zawiera wzorca, algorytm Boyera-Moore'a wykonuje nie więcej niż  $3|text|$  porównań znaków

Gusfield przedstawia na stronach 39–45 dowód słabszego ograniczenia górnego na liczbę porównań znaków  $4|text|$

## Algorytm Boyera-Moore'a – najgorszy przypadek

Gdy tekst zawiera wzorec, algorytm Boyera-Moore'a może wykonywać aż  $O(|\text{text}||\text{pat}|)$  porównań znaków, ponieważ zapomina, które znaki już porównywał

Alberto Apostolico i Raffaele Giancarlo ulepszyli ten algorytm tak, żeby zapamiętywał długość sufiksu wzorca równego podłańcuchowi tekstu.

Algorytm Apostolico-Giancarlo wykonuje nie więcej niż  $\frac{3}{2}|\text{text}|$  porównań znaków

## Algorytm Boyera-Moore'a – najlepszy przypadek

Jeżeli zachodzą oba poniższe przypadki:

- kiedy porównujemy znaki tekstu ze znakami wzorca, zawsze się okazuje, że te znaki są różne
- reguła niezgodności zawsze przesuwa wzorzec o całą jego długość

to algorytm Boyera-Moore'a wykonuje tylko  $\lfloor \frac{|text|}{|pat|} \rfloor$  porównań znaków

## Algorytm Boyera-Moore'a – podsumowanie

### Algorytm Boyera-Moore'a:

- wymaga wstępnego przetwarzania wzorca, które da się zrobić w czasie  $O(|pat|)$
- korzysta z  $O(|pat|)$  dodatkowej pamięci na mapę `lastOccurrences`, na tablicę `goodSuffixes` i na kilka zmiennych
- jeśli tekst nie zawiera wzorca, to wykonuje nie więcej niż  $3|text|$  porównań znaków
- jeśli tekst zawiera wzorzec, to może wykonywać  $O(|text||pat|)$  porównań znaków
- w najlepszym przypadku wykonuje  $O(|text|/|pat|)$  porównań znaków

# Algorytm Knutha-Morrisa-Pratta

---





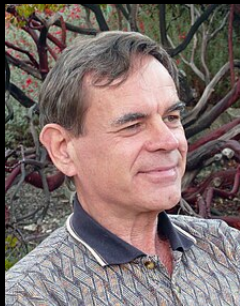
Amerykański informatyk i matematyk. Emerytowany profesor sztuki programowania na Uniwersytecie Stanforda. Laureat Nagrody Turinga w 1974 roku za wkład w analizę algorytmów, projektowanie języków programowania i serię książek **Sztuka programowania**. Autor systemu składu tekstu  $\text{T}_{\text{E}}\text{X}$ , wielu książek o matematyce i informatyce, książki o Biblii i utworu na organy **Fantasia apocalyptic**.

## James H. Morris (1941–)



Amerykański informatyk. Profesor Uniwersytetu Carnegie-Mellon. Jedną z osób, które wprowadziły **leniwą ewaluację** w językach programowania. Pracował nad komputerem Xerox Alto, pierwszym komputerem z interfejsem graficznym i myszką. W 1970 roku niezależnie od Donalda Knutha odkrył algorytm wyszukiwania wzorca, który zawsze działa w czasie liniowym względem długości tekstu.

## Vaughan Pratt (12.4.1944–)



Amerykański informatyk i geofizyk, z pochodzenia Australijczyk.  
Emerytowany profesor Uniwersytetu Stanforda. Doktorant Donalda Knutha.  
Pracował nad stacjami roboczymi Sun. Opracował logo firmy  
Sun Microsystems.

### Definicja:

Dla danego łańcucha  $s$

$p[j]$  to długość najdłuższego takiego właściwego sufiksu łańcucha  $s[:j+1]$ , który jest pewnym prefiksem łańcucha  $s$

Można łatwo stablicować wartości funkcji prefiksowej na podstawie wyniku funkcji **ZFunction**

## Algorytm KMP – funkcja prefiksowa

```
// KMPPrefixFunction zwraca wycinek. `j`-ty element tego wycinka  
// to wartość funkcji prefiksowej `p[j]`, czyli długość najdłuższego  
// takiego właściwego sufiksu łańcucha `s[:j+1]`, który jest pewnym  
// prefiksem łańcucha `s`  
func KMPPrefixFunction(s []byte) []int {  
    z := ZFunction(s)  
    p := make([]int, len(s)+1)  
    for j := len(s) - 1; j > 0; j-- {  
        p[j+z[j]] = z[j]  
    }  
    return p  
}
```

## Algorytm KMP – funkcja prefiksowa – przykład

dźwiędź

z: 0000020

p: 00000002

## Algorytm KMP – funkcja prefiksowa – przykład

dźwiędź

p: 00000002

owocowo

z: 0010301

p: 00010003

## Algorytm KMP – funkcja prefiksowa – przykład

dźwiędź

p: 00000002

owocowo

p: 00010003

iiii

z: 04321

p: 000004



## Algorytm KMP – funkcja prefiksowa – przykład

dźwiędź

p: 00000002

owocowo

p: 00010003

iiii

p: 000004

# Algorytm KMP

Porównuję kolejne znaki `text[i]` z `pat[j]`

W każdej iteracji głównej pętli zwiększam `i` o 1

- Dopóki `j > 0` i `text[i] != pat[j]`:  
przesuwam wzorec `pat` w prawo o `j-p[j]` pozycji  
nie zmieniam wartości `i`
- Jeśli `text[i] == pat[j]`, to:  
nie przesuwam wzorca `pat`  
zwiększam `j` o 1
- Jeśli `j == len(pat)`, czyli wzorec `pat` został znaleziony :-), to:  
przesuwam wzorec `pat` w prawo o `len(pat)-p[len(pat)]` pozycji



## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 0

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.  
dźwiedź

d ż w i e d ź

0 0 0 0 0 0 0 2

porównań znaków: 0

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 1

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 1

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 2



## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 2

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 3

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 3

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 4

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 4

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 5

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 5

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 6



## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ź

0 0 0 0 0 0 0 2

porównań znaków: 6

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ź

0 0 0 0 0 0 0 2

porównań znaków: 7

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 8

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 9

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ź

0 0 0 0 0 0 0 2

porównań znaków: 10

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 11

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ż

0 0 0 0 0 0 0 2

porównań znaków: 12

## Algorytm KMP – przykład 1

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d ż w i e d ź

0 0 0 0 0 0 0 2

porównań znaków: 13



## Algorytm KMP – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d z w i e d ź

0 0 0 0 0 0 0 2      $j-p[j] = 7-2 = 5$

porównań znaków: 13

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2       $j-p[j] = 2-0 = 2$

porównań znaków: 14

## Algorytm KMP – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź      dźwiedź

dźwiędź      dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 14

## Algorytm KMP – przykład 1

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź      dźwiędź

dźwiędź      dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 15

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 15

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 16

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 16

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 17



## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 17

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 18

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 18

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 19

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 19

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 20

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 20

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 21



# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 21

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 22

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 22

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 23

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 23

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 24

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 24

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 25



# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 26

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 27

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 28

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 29

# Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 30

## Algorytm KMP – przykład 1

dźwiedź

To\_niedźwiedź\_czy\_może\_dźwiedź?\_Chyba\_nie\_dźwiedź.

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź      dźwiedź

dźwiedź

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 31

## Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź      dźwiędź dźwiędź

dźwiędź      dźwiędź dźwiędź

dźwiędź      dźwiędź dźwiędź

dźwiędź      dźwiędź dźwiędź

dźwiędź      dźwiędź

dźwiędź      dźwiędź

dźwiędź      dźwiędź

dźwiędź

d   z   w   i   e   d   ź

0 0 0 0 0 0 0 2       $j-p[j] = 7-2 = 5$

porównań znaków: 31

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2     $j-p[j] = 2-0 = 2$

porównań znaków: 32



## Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 32

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź  
dźwiędź    dźwiędź dźwiędź    dźwiędź  
dźwiędź    dźwiędź dźwiędź  
dźwiędź    dźwiędź dźwiędź  
dźwiędź    dźwiędź  
dźwiędź    dźwiędź  
dźwiędź    dźwiędź  
            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 33

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 33

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 34

## Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 34

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 35

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź                    dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 35

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 36



# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 36

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 37

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 37

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 38

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 38

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 39

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 39

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 40



# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 40

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 41

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 41

## Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 42

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 42

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

            dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 43

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 43

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź    dźwiędź            dźwiędź

            dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 44



# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 44

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 45

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 46

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 47

## Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 48

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 49

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 50

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź dźwiędź

dźwiędź    dźwiędź            dźwiędź

            dźwiędź            dźwiędź

d   z   w   i   e   d   z

0 0 0 0 0 0 0 2

porównań znaków: 51



# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź    dźwiędź dźwiędź

dźwiędź    dźwiędź    dźwiędź

dźwiędź    dźwiędź

d   z   w   i   e   d   ź

0 0 0 0 0 0 0 2     $j - p[j] = 7 - 2 = 5$

porównań znaków: 51

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź

d z w i e d ź

0 0 0 0 0 0 0 2  $j-p[j] = 2-0 = 2$

porównań znaków: 52

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź d

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź

d z w i e d ź

0 0 0 0 0 0 0 2

porównań znaków: 52

# Algorytm KMP – przykład 1

dźwięd                      dźwięd                      dźwięd

To\_niedźwięd\_czy\_może\_dźwięd?\_Chyba\_nie\_dźwięd.

dźwięd    dźwięd dźwięd    dźwięd dźwięd    dźw

dźwięd            dźwięd dźwięd    dźwięd dźwięd    d

dźwięd            dźwięd dźwięd    dźwięd dźwięd

dźwięd            dźwięd dźwięd    dźwięd dźwięd

dźwięd            dźwięd                      dźwięd dźwięd

dźwięd            dźwięd                      dźwięd dźwięd

dźwięd            dźwięd                      dźwięd

dźwięd            dźwięd                      dźwięd

d   ż   w   i   e   d   ż

0 0 0 0 0 0 0 2

porównań znaków: 53

# Algorytm KMP – przykład 1

dźwiędź

dźwiędź

dźwiędź

To\_niedźwiędź\_czy\_może\_dźwiędź?\_Chyba\_nie\_dźwiędź.

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź d

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź dźwiędź

dźwiędź dźwiędź

d z w i e d ź

0 0 0 0 0 0 0 2

porównań znaków: 53

porównań znaków: 53



## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

o w o c o w o

0 0 0 1 0 0 0 3

porównań znaków: 0



## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 0

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

owocowo

o w o c o w o

0 0 0 1 0 0 0 3

porównań znaków: 1

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 1

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 2

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3       $j - p[j] = 1 - 0 = 1$

porównań znaków: 3

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 3

0 W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 3

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 4

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 4



## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 5

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 5

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 6

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 6

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 7

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 8

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 9

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3      $j - p[j] = 3 - 1 = 2$

porównań znaków: 10



## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3      $j - p[j] = 1 - 0 = 1$

porównań znaków: 11

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O   O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 11

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

OWOCOWO OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 12

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O O W O C O W O

O W O C O W O O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 12

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O O W O C O W O

O W O C O W O O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 13

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O O W O C O W O

O W O C O W O O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 14

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O O W O C O W O

O W O C O W O O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 15

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O O W O C O W O

O W O C O W O O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 16



## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O   O W O C O W O

O W O C O W O   O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 17

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O O W O C O W O

O W O C O W O O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 18

## Algorytm KMP – przykład 2

to\_i\_owo\_owocowo

O W O C O W O O W O C O W O

O W O C O W O O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 19

## Algorytm KMP – przykład 2

OWOCOWO

to\_i\_owo\_owocowo

OWOCOWO OWOCOWO

OWOCOWO OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

o w o c o w o

0 0 0 1 0 0 0 3      $j-p[j] = 7-3 = 4$

porównań znaków: 19

## Algorytm KMP – przykład 2

OWOCOWO

to\_i\_owo\_owocowo

OWOCOWO OWOCOWO

OWOCOWO OWOCOWO

OWOCOWO OWO

OWOCOWO

OWOCOWO

OWOCOWO

OWOCOWO

O W O C O W O

0 0 0 1 0 0 0 3

porównań znaków: 19

porównań znaków: 19

## Algorytm KMP

```
func KMP(pat, text []byte, output func(int)) {  
    // len(pat) > 0  
    p := KMPPrefixFunction(pat)  
    j := 0  
    for i := 0; i < len(text); i++ {  
        for j > 0 && text[i] != pat[j] {  
            j = p[j]  
        }  
        if text[i] == pat[j] {  
            j++  
        }  
        if j == len(pat) {  
            output(i - len(pat) + 1)  
            j = p[j]  
        }  
    }  
}
```

## Algorytm KMP – złożoność czasowa

- Funkcja `KMPPrefixFunction` wykonuje  $O(|pat|)$  działań
- Funkcja `KMP` wykonuje w swojej głównej pętli co najwyżej  $2|text|$  porównań znaków. Gusfield dowodzi tego faktu na stronie 26
- Zatem funkcja `KMP` wykonuje  $O(|text| + |pat|)$  działań



## Algorytm KMP – złożoność pamięciowa

- Funkcja `KMPPrefixFunction` zwraca tablicę, która zawiera  $|pat| + 1$  liczb całkowitych
- Zatem funkcja `KMP` wymaga  $O(|pat|)$  dodatkowej pamięci

Algorytm Knutha-Morrisa-Pratta:

- wymaga wstępnego przetwarzania wzorca, które da się zrobić w czasie  $O(|pat|)$
- korzysta z  $O(|pat|)$  dodatkowej pamięci na tablicę  $p$  i na kilka zmiennych
- wykonuje w głównej pętli nie więcej niż  $2|text|$  porównań znaków
- wraz ze wstępnym przetwarzaniem działa w czasie  $O(|text| + |pat|)$

# Algorytm Karpa-Rabina

---

## Richard Karp (3.1.1935–)



Amerykański informatyk. Laureat Nagrody Turinga w 1985 roku za rozwój teorii problemów NP-zupełnych. Współtwórca m.in. algorytmu Edmondsa-Karpa, który znajduje maksymalny przepływ w sieci, algorytmu Hopcrofta-Karpa, który znajduje maksymalne skojarzenie w grafie dwudzielnym, i algorytmu Karpa-Rabina.

## Michael Rabin (1.9.1931–)



Izraelski matematyk i informatyk. Urodził się we Wrocławiu. Laureat Nagrody Turinga w 1976 roku za wprowadzenie pojęcia niedeterministycznych automatów skończonych. W 1987 roku wraz z Richardem M. Karpem opublikował szybki algorytm, który wyszukuje wzorzec w tekście, korzystając z haszowania.

- Haszuję wzorzec `pat`
- Haszuję kolejne łańcuchy `text[i:i+len(pat)]`
- Jeśli wynik haszowania łańcucha `text[i:i+len(pat)]` jest równy wynikowi haszowania wzorca `pat`, to porównuję łańcuch `text[i:i+len(pat)]` ze wzorcem `pat`. Z powodu kolizji funkcji haszującej łańcuch `text[i:i+len(pat)]` może być różny od wzorca `pat`. Ale jeśli te 2 łańcuchy są równe, to podaję, że wzorzec `pat` występuje w tekście `text` od pozycji `i`

Algorytm Karpa-Rabina jest szybki dzięki temu, że potrafi haszować łańcuch `text[i+1:i+1+len(pat)]` w czasie  $O(1)$  na podstawie wyniku haszowania łańcucha `text[i:i+len(pat)]`

*// hashByteModN zwraca liczbę całkowitą z przedziału  $[0, n)$*

```
func hashByteModN(b byte, h, n uint64) uint64 {  
    // Nie ma przepełnienia, jeśli  
    //  $h \ll 8 + \text{uint64}(b) < 1 \ll 64$   
    //  $h \ll 8 < 1 \ll 64 - \text{uint64}(b)$   
    //  $h \ll 8 < 1 \ll 64 - 1 \ll 8 + 1$   
    //  $h \ll 8 \leq 1 \ll 64 - 1 \ll 8$   
    //  $h \ll 8 \gg 8 \leq 1 \ll 64 \gg 8 - 1 \ll 8 \gg 8$   
    //  $h \leq 1 \ll 56 - 1$   
    //  $h < 1 \ll 56$   
    return (h<<8 + uint64(b)) % n  
}
```



```
// hashBytesModN zwraca liczbę całkowitą z przedziału [0, n)  
func hashBytesModN(bs []byte, n uint64) uint64 {  
    h := uint64(0)  
    for _, b := range bs {  
        // h < n  
        h = hashByteModN(b, h, n)  
    }  
    return h  
}
```

## Algorytm Karpa-Rabina

```
// twoToPower8PModN zwraca  $2^{(8*p)} \% n$   
func twoToPower8PModN(p int, n uint64) uint64 {  
    r := uint64(1)  
    for i := 0; i < p; i++ {  
        //  $r == 1 \ll (8*i) \% n$   
        //  $r == 2^{(8*i)} \% n$   
        // Nie ma przepełnienia, jeśli  
        //  $r \ll 8 < 1 \ll 64$   
        //  $r \ll 8 \gg 8 < 1 \ll 64 \gg 8$   
        //  $r < 1 \ll 56$   
        r = (r << 8) % n  
    }  
    return r  
}
```

## Dygresja: operacja % a operacja mod

Wynik operacji % ma taki sam znak, jak dzielna:

$$7 \% 5 == 2$$

$$7 \% -5 == 2$$

$$-7 \% 5 == -2$$

$$-7 \% -5 == -2$$

Potrzebuję operacji mod, która zawsze zwraca dodatni wynik:

$$7 \bmod 5 == 2$$

$$7 \bmod -5 == 2$$

$$-7 \bmod 5 == 3$$

$$-7 \bmod -5 == 3$$

## Algorytm Karpa-Rabina

```
// unhashByteModN zwraca liczbę całkowitą z przedziału [0, n)
func unhashByteModN(b byte, h, n, power uint64) uint64 {
    // Nie ma przepełnienia, jeśli
    //  $h - \text{power} * \text{uint64}(b) > -(1 \ll 64)$ 
    //  $0 - \text{power} * (1 \ll 8 - 1) > -(1 \ll 64)$ 
    //  $\text{power} * (1 \ll 8 - 1) < 1 \ll 64$ 
    //  $\text{power} \ll 8 - \text{power} < 1 \ll 64$ 
    //  $\text{power} \ll 8 < 1 \ll 64$ 
    //  $\text{power} \ll 8 \gg 8 < 1 \ll 64 \gg 8$ 
    //  $\text{power} < 1 \ll 56$ 
    r := h - power*uint64(b)
    if int64(r) < 0 {
        r += n * (-r/n + 1)
    }
    return r
}
```

## Algorytm Karpa-Rabina

```
func KarpRabin(pat, text []byte, output func(int)) {  
    // len(pat) <= len(text)  
    h := hashBytesModN(text[:len(pat)], N)  
    ph := hashBytesModN(pat, N)  
    power := twoToPower8PModN(len(pat), N)  
    for i := 0; ; i++ {  
        // h == hashBytesModN(text[i:i+len(pat)], N)  
        if h == ph && slices.Equal(pat, text[i:i+len(pat)]) {  
            output(i)  
        }  
        if i+len(pat) >= len(text) {  
            break  
        }  
        h = hashByteModN(text[i+len(pat)], h, N)  
        h = unhashByteModN(text[i], h, N, power)  
    }  
}
```

## Algorytm Karpa-Rabina – jak dobrać wartość $N$ ?

- Nie ma przepiętnienia, jeśli wyniki funkcji `HashByteModN`, `twoToPower8PModN` i `unhashByteModN` są mniejsze niż  $1 \ll 56$
- Wyniki funkcji `HashByteModN`, `twoToPower8PModN` i `unhashByteModN` są równe co najwyżej  $N$
- Jak dobrać wartość  $N$ ?

## Algorytm Karpa-Rabina – jak dobrać wartość $N$ ?

- Jeśli liczba  $N$  jest duża, to prawdopodobieństwo kolizji funkcji haszującej jest niskie
- Kolizje funkcji haszującej spowalniają działanie funkcji KarpRabin
- Gusfield dowodzi na stronach 79–80, że prawdopodobieństwo kolizji funkcji haszującej jest ograniczone z góry, jeśli  $N$  jest liczbą pierwszą

## Algorytm Karpa-Rabina – jak dobrać wartość $N$ ?

```
// Największa liczba pierwsza mniejsza niż  $1 \ll 56$   
// https://t5k.org/lists/2small/0bit.html  
const N uint64 = 1<<56 - 5
```





`To_niedzwiedz_czy_moze_dzwiedz?_Chyba_nie_dzwiedz.`

`N = 100 :-)`

`hash(" ") = 0`

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$N = 100$  :-)

$\text{hash}("") = 0$

$\text{hash}("d") = (0 \ll 8 + 100) \% 100 = 0$

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$N = 100$  :-)

$\text{hash}("") = 0$

$\text{hash}("d") = (0 \ll 8 + 100) \% 100 = 0$

$\text{hash}("dz") = (0 \ll 8 + 122) \% 100 = 22$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$N = 100$  :-)

$\text{hash}("") = 0$

$\text{hash}("d") = (0 \ll 8 + 100) \% 100 = 0$

$\text{hash}("dz") = (0 \ll 8 + 122) \% 100 = 22$

$\text{hash}("dzw") = (22 \ll 8 + 119) \% 100 = 51$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$N = 100$  :-)

$\text{hash}("") = 0$

$\text{hash}("d") = (0 \ll 8 + 100) \% 100 = 0$

$\text{hash}("dz") = (0 \ll 8 + 122) \% 100 = 22$

$\text{hash}("dzw") = (22 \ll 8 + 119) \% 100 = 51$

$\text{hash}("dzwi") = (51 \ll 8 + 105) \% 100 = 61$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$N = 100$  :-)

$\text{hash}("") = 0$

$\text{hash}("d") = (0 \ll 8 + 100) \% 100 = 0$

$\text{hash}("dz") = (0 \ll 8 + 122) \% 100 = 22$

$\text{hash}("dzw") = (22 \ll 8 + 119) \% 100 = 51$

$\text{hash}("dzwi") = (51 \ll 8 + 105) \% 100 = 61$

$\text{hash}("dzwie") = (61 \ll 8 + 101) \% 100 = 17$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$N = 100$  :-)

$\text{hash}("") = 0$

$\text{hash}("d") = (0 \ll 8 + 100) \% 100 = 0$

$\text{hash}("dz") = (0 \ll 8 + 122) \% 100 = 22$

$\text{hash}("dzw") = (22 \ll 8 + 119) \% 100 = 51$

$\text{hash}("dzwi") = (51 \ll 8 + 105) \% 100 = 61$

$\text{hash}("dzwie") = (61 \ll 8 + 101) \% 100 = 17$

$\text{hash}("dzwied") = (17 \ll 8 + 100) \% 100 = 52$



## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$N = 100$  :-)

$\text{hash}("") = 0$

$\text{hash}("d") = (0 \ll 8 + 100) \% 100 = 0$

$\text{hash}("dz") = (0 \ll 8 + 122) \% 100 = 22$

$\text{hash}("dzw") = (22 \ll 8 + 119) \% 100 = 51$

$\text{hash}("dzwi") = (51 \ll 8 + 105) \% 100 = 61$

$\text{hash}("dzwie") = (61 \ll 8 + 101) \% 100 = 17$

$\text{hash}("dzwied") = (17 \ll 8 + 100) \% 100 = 52$

$\text{hash}("dzwiedz") = (52 \ll 8 + 122) \% 100 = 34$

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

hash("dzwiedz") = 34

hash("") = 0

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

hash("dzwiedz") = 34

hash("") = 0

hash("T") =  $(0 \ll 8 + 84) \% 100 = 84$

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$\text{hash}(\text{"dzwiedz"}) = 34$

$\text{hash}("") = 0$

$\text{hash}(\text{"T"}) = (0 \ll 8 + 84) \% 100 = 84$

$\text{hash}(\text{"To"}) = (84 \ll 8 + 111) \% 100 = 15$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$\text{hash}(\text{"dzwiedz"}) = 34$

$\text{hash}("") = 0$

$\text{hash}(\text{"T"}) = (0 \ll 8 + 84) \% 100 = 84$

$\text{hash}(\text{"To"}) = (84 \ll 8 + 111) \% 100 = 15$

$\text{hash}(\text{"To_"}) = (15 \ll 8 + 32) \% 100 = 72$

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$\text{hash}(\text{"dzwiedz"}) = 34$

$\text{hash}("") = 0$

$\text{hash}(\text{"T"}) = (0 \ll 8 + 84) \% 100 = 84$

$\text{hash}(\text{"To"}) = (84 \ll 8 + 111) \% 100 = 15$

$\text{hash}(\text{"To_"}) = (15 \ll 8 + 32) \% 100 = 72$

$\text{hash}(\text{"To_n"}) = (72 \ll 8 + 110) \% 100 = 42$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$\text{hash}(\text{"dzwiedz"}) = 34$

$\text{hash}("") = 0$

$\text{hash}(\text{"T"}) = (0 \ll 8 + 84) \% 100 = 84$

$\text{hash}(\text{"To"}) = (84 \ll 8 + 111) \% 100 = 15$

$\text{hash}(\text{"To_"}) = (15 \ll 8 + 32) \% 100 = 72$

$\text{hash}(\text{"To_n"}) = (72 \ll 8 + 110) \% 100 = 42$

$\text{hash}(\text{"To_ni"}) = (42 \ll 8 + 105) \% 100 = 57$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$\text{hash}(\text{"dzwiedz"}) = 34$

$\text{hash}(\text{""}) = 0$

$\text{hash}(\text{"T"}) = (0 \ll 8 + 84) \% 100 = 84$

$\text{hash}(\text{"To"}) = (84 \ll 8 + 111) \% 100 = 15$

$\text{hash}(\text{"To_"}) = (15 \ll 8 + 32) \% 100 = 72$

$\text{hash}(\text{"To_n"}) = (72 \ll 8 + 110) \% 100 = 42$

$\text{hash}(\text{"To_ni"}) = (42 \ll 8 + 105) \% 100 = 57$

$\text{hash}(\text{"To_nie"}) = (57 \ll 8 + 101) \% 100 = 93$



## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

$\text{hash}(\text{"dzwiedz"}) = 34$

$\text{hash}("") = 0$

$\text{hash}(\text{"T"}) = (0 \ll 8 + 84) \% 100 = 84$

$\text{hash}(\text{"To"}) = (84 \ll 8 + 111) \% 100 = 15$

$\text{hash}(\text{"To_"}) = (15 \ll 8 + 32) \% 100 = 72$

$\text{hash}(\text{"To_n"}) = (72 \ll 8 + 110) \% 100 = 42$

$\text{hash}(\text{"To_ni"}) = (42 \ll 8 + 105) \% 100 = 57$

$\text{hash}(\text{"To_nie"}) = (57 \ll 8 + 101) \% 100 = 93$

$\text{hash}(\text{"To_nied"}) = (93 \ll 8 + 100) \% 100 = 8$

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

hash("dzwiedz") = 34

hash("To\_nied") = 8

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

hash("dzwiedz") = 34

hash("To\_nied") = 8

power =  $256^7 \% 100 = 36$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 8|

| 34|

$\text{hash}(\text{"To\_niedz"}) = (8 \ll 8 + 122) \% 100 = 70$

$\text{hash}(\text{"o\_niedz"}) = (70 - 36 \cdot 84) \bmod 100 = 46$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 46|

| 34|

$\text{hash}(\text{"o\_niedz"}_w) = (46 \ll 8 + 119) \% 100 = 95$

$\text{hash}(\text{"\_niedz"}_w) = (95 - 36 \cdot 111) \bmod 100 = 99$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 99|

| 34|

$\text{hash}(\text{"_niedzwi"}) = (99 \ll 8 + 105) \% 100 = 49$

$\text{hash}(\text{"niedzwi"}) = (49 - 36 \cdot 32) \bmod 100 = 97$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 97|

| 34|

$\text{hash}(\text{"niedzwie"}) = (97 \ll 8 + 101) \% 100 = 33$

$\text{hash}(\text{"iedzwie"}) = (33 - 36 \cdot 110) \bmod 100 = 73$

## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 73|

| 34|

$\text{hash}(\text{"iedzwiedz"}) = (73 \ll 8 + 100) \% 100 = 88$

$\text{hash}(\text{"edzwiedz"}) = (88 - 36 \cdot 105) \bmod 100 = 8$



## Algorytm Karpa-Rabina – przykład

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 8|

| 34|

$\text{hash}(\text{"edzwiedz"}) = (8 \ll 8 + 122) \% 100 = 70$

$\text{hash}(\text{"dzwiedz"}) = (70 - 36 \cdot 101) \bmod 100 = 34$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 34|

| 34|

$\text{hash}(\text{"dzwiedz\_"}) = (34 \ll 8 + 32) \% 100 = 36$

$\text{hash}(\text{"zwiedz\_"}) = (36 - 36 * 100) \bmod 100 = 36$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 36|

| 34|

$\text{hash}(\text{"zwiedz\_c"}) = (36 \ll 8 + 99) \% 100 = 15$

$\text{hash}(\text{"wiedz\_c"}) = (15 - 36 * 122) \bmod 100 = 23$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 23|

| 34|

$\text{hash}(\text{"wiedz\_cz"}) = (23 \ll 8 + 122) \% 100 = 10$

$\text{hash}(\text{"iedz\_cz"}) = (10 - 36 * 119) \bmod 100 = 26$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 26 |

| 34 |

$\text{hash}(\text{"iedz\_czy"}) = (26 \ll 8 + 121) \% 100 = 77$

$\text{hash}(\text{"edz\_czy"}) = (77 - 36 \cdot 105) \bmod 100 = 97$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 97|

| 34|

$\text{hash}(\text{"edz\_czy\_"}) = (97 \ll 8 + 32) \% 100 = 64$

$\text{hash}(\text{"dz\_czy\_"}) = (64 - 36 \cdot 101) \bmod 100 = 28$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 28|

| 34|

$\text{hash}(\text{"dz\_czy\_m"}) = (28 \ll 8 + 109) \% 100 = 77$

$\text{hash}(\text{"z\_czy\_m"}) = (77 - 36 \cdot 100) \bmod 100 = 77$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 77|

| 34|

$\text{hash}(\text{"z\_czy\_mo"}) = (77 \ll 8 + 111) \% 100 = 23$

$\text{hash}(\text{"\_czy\_mo"}) = (23 - 36 \cdot 122) \bmod 100 = 31$



## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 31 |

| 34 |

$\text{hash}(\text{"\_czy\_moz"}) = (31 \ll 8 + 122) \% 100 = 58$

$\text{hash}(\text{"czy\_moz"}) = (58 - 36 \cdot 32) \bmod 100 = 6$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 6|

| 34|

$\text{hash}(\text{"czy\_moze"}) = (6 \ll 8 + 101) \% 100 = 37$

$\text{hash}(\text{"zy\_moze"}) = (37 - 36 \cdot 99) \bmod 100 = 73$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 73|

| 34|

$\text{hash}(\text{"zy\_moze\_"}) = (73 \ll 8 + 32) \% 100 = 20$

$\text{hash}(\text{"y\_moze\_"}) = (20 - 36 * 122) \bmod 100 = 28$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 28|

| 34|

$\text{hash}(\text{"y\_moze\_d"}) = (28 \ll 8 + 100) \% 100 = 68$

$\text{hash}(\text{"\_moze\_d"}) = (68 - 36 * 121) \bmod 100 = 12$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 12|

| 34|

$\text{hash}(\text{"_moze\_dz"}) = (12 \ll 8 + 122) \% 100 = 94$

$\text{hash}(\text{"moze\_dz"}) = (94 - 36 \cdot 32) \bmod 100 = 42$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 42|

| 34|

$\text{hash}(\text{"moze\_dzw"}) = (42 \ll 8 + 119) \% 100 = 71$

$\text{hash}(\text{"oze\_dzw"}) = (71 - 36 \cdot 109) \bmod 100 = 47$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 47|

| 34|

$\text{hash}(\text{"oze\_dzwi"}) = (47 \ll 8 + 105) \% 100 = 37$

$\text{hash}(\text{"ze\_dzwi"}) = (37 - 36 \cdot 111) \bmod 100 = 41$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 41|

| 34|

$\text{hash}(\text{"ze\_dzwie"}) = (41 \ll 8 + 101) \% 100 = 97$

$\text{hash}(\text{"e\_dzwie"}) = (97 - 36 \cdot 122) \bmod 100 = 5$



## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 5|

| 34|

$\text{hash}(\text{"e\_dzwiedz"}) = (5 \ll 8 + 100) \% 100 = 80$

$\text{hash}(\text{"_dzwiedz"}) = (80 - 36 \cdot 101) \bmod 100 = 44$

## Algorytm Karpa-Rabina – przykład

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 44|

| 34|

$\text{hash}(\text{"_dzwiedz"}) = (44 \ll 8 + 122) \% 100 = 86$

$\text{hash}(\text{"dzwiedz"}) = (86 - 36 \cdot 32) \bmod 100 = 34$

## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 34|

| 34|

$\text{hash}(\text{"dzwiedz?"}) = (34 \ll 8 + 63) \% 100 = 67$

$\text{hash}(\text{"zwiedz?"}) = (67 - 36 \cdot 100) \bmod 100 = 67$

## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 67|

| 34|

$\text{hash}(\text{"zwiedz?_"}) = (67 \ll 8 + 32) \% 100 = 84$

$\text{hash}(\text{"wiedz?_"}) = (84 - 36 \cdot 122) \bmod 100 = 92$

## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 92|

| 34|

$\text{hash}(\text{"wiedz?_C"}) = (92 \ll 8 + 67) \% 100 = 19$

$\text{hash}(\text{"iedz?_C"}) = (19 - 36 * 119) \bmod 100 = 35$

## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 35|

| 34|

$\text{hash}(\text{"iedz?_Ch"}) = (35 \ll 8 + 104) \% 100 = 64$

$\text{hash}(\text{"edz?_Ch"}) = (64 - 36 \cdot 105) \bmod 100 = 84$

## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 84|

| 34|

$\text{hash}(\text{"edz?_Chy"}) = (84 \ll 8 + 121) \% 100 = 25$

$\text{hash}(\text{"dz?_Chy"}) = (25 - 36 \cdot 101) \bmod 100 = 89$

## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 89|

| 34|

$\text{hash}(\text{"dz?_Chyb"}) = (89 \ll 8 + 98) \% 100 = 82$

$\text{hash}(\text{"z?_Chyb"}) = (82 - 36 \cdot 100) \bmod 100 = 82$



## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 82|

| 34|

$\text{hash}(\text{"z?_Chyba"}) = (82 \ll 8 + 97) \% 100 = 89$

$\text{hash}(\text{"?_Chyba"}) = (89 - 36 \cdot 122) \bmod 100 = 97$

## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 97 |

| 34 |

$\text{hash}("?\_Chyba\_") = (97 \ll 8 + 32) \% 100 = 64$

$\text{hash}("_Chyba\_") = (64 - 36 * 63) \bmod 100 = 96$

## Algorytm Karpa-Rabina – przykład

dzwiedz

dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 96|

| 34|

$\text{hash}(\text{"_Chyba\_n"}) = (96 \ll 8 + 110) \% 100 = 86$

$\text{hash}(\text{"Chyba\_n"}) = (86 - 36 \cdot 32) \bmod 100 = 34$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz   Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    34 |

|    34 |

$\text{hash}(\text{"Chyba\_ni"}) = (34 \ll 8 + 105) \% 100 = 9$

$\text{hash}(\text{"hyba\_ni"}) = (9 - 36 * 67) \bmod 100 = 97$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz   Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    97 |

|    34 |

$\text{hash}(\text{"hyba\_nie"}) = (97 \ll 8 + 101) \% 100 = 33$

$\text{hash}(\text{"yba\_nie"}) = (33 - 36 \cdot 104) \bmod 100 = 89$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz    Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    89|

|    34|

$\text{hash}(\text{"yba\_nie\_"}) = (89 \ll 8 + 32) \% 100 = 16$

$\text{hash}(\text{"ba\_nie\_"}) = (16 - 36 \cdot 121) \bmod 100 = 60$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz    Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    60 |

|    34 |

$\text{hash}(\text{"ba\_nie\_d"}) = (60 \ll 8 + 100) \% 100 = 60$

$\text{hash}(\text{"a\_nie\_d"}) = (60 - 36 \cdot 98) \bmod 100 = 32$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz   Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    32|

|    34|

$\text{hash}(\text{"a\_nie\_dz"}) = (32 \ll 8 + 122) \% 100 = 14$

$\text{hash}(\text{"\_nie\_dz"}) = (14 - 36 \cdot 97) \bmod 100 = 22$



## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz   Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    22|

|    34|

$\text{hash}(\text{"_nie\_dzw"}) = (22 \ll 8 + 119) \% 100 = 51$

$\text{hash}(\text{"nie\_dzw"}) = (51 - 36 \cdot 32) \bmod 100 = 99$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz   Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    99 |

|    34 |

$\text{hash}(\text{"nie\_dzwi"}) = (99 \ll 8 + 105) \% 100 = 49$

$\text{hash}(\text{"ie\_dzwi"}) = (49 - 36 * 110) \bmod 100 = 89$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz   Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|   89|

|   34|

$\text{hash}(\text{"ie\_dzwie"}) = (89 \ll 8 + 101) \% 100 = 85$

$\text{hash}(\text{"e\_dzwie"}) = (85 - 36 \cdot 105) \bmod 100 = 5$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz   Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    5 |

|    34 |

$\text{hash}(\text{"e\_dzwiedz"}) = (5 \ll 8 + 100) \% 100 = 80$

$\text{hash}(\text{"\_dzwiedz"}) = (80 - 36 \cdot 101) \bmod 100 = 44$

## Algorytm Karpa-Rabina – przykład

dzwiedz                      dzwiedz   Chyba n

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

|    44 |

|    34 |

$\text{hash}(\text{"_dzwiedz"}) = (44 \ll 8 + 122) \% 100 = 86$

$\text{hash}(\text{"dzwiedz"}) = (86 - 36 \cdot 32) \bmod 100 = 34$

## Algorytm Karpa-Rabina – przykład

dzwiedz	dzwiedz	Chyba n	dzwiedz
To_niedzwiedz_czy_moze_dzwiedz?_Chyba_nie_dzwiedz.			
			34
			34

$\text{hash}(\text{"dzwiedz."}) = (34 \ll 8 + 46) \% 100 = 50$

$\text{hash}(\text{"zwiedz."}) = (50 - 36 * 100) \bmod 100 = 50$

## Algorytm Karpa-Rabina – przykład

dzwiedz                dzwiedz Chyba n    dzwiedz

To\_niedzwiedz\_czy\_moze\_dzwiedz?\_Chyba\_nie\_dzwiedz.

| 50|

| 34|





## Algorytm Karpa-Rabina – podsumowanie

### Algorytm Karpa-Rabina:

- korzysta z haszowania
- wymaga wstępnego przetwarzania, które da się zrobić w  $O(|\text{pat}|)$  czasu
- korzysta z  $O(1)$  pamięci na kilka zmiennych
- wykonuje w głównej pętli  $O(|\text{text}|)$  operacji, jeśli prawdopodobieństwo kolizji funkcji haszującej jest ograniczone z góry
- wraz ze wstępnym przetwarzaniem działa w czasie  $O(|\text{text}| + |\text{pat}|)$
- może zostać uogólniony tak, żeby naraz wyszukiwał wiele różnych wzorców, które mają tę samą długość

## Podsumowanie

---

- Algorytm naiwny
- Funkcja Z
- Algorytm Boyera-Moore'a
- Algorytm Knutha-Morrisa-Pratta
- Algorytm Karpa-Rabina

Algorytm naiwny:

- korzysta z  $O(1)$  dodatkowej pamięci na zmienne  $i$  i  $j$
- w najlepszym przypadku wykonuje  $O(|\text{text}|)$  porównań znaków
- w najgorszym przypadku wykonuje  $O(|\text{text}||\text{pat}|)$  porównań znaków

## Algorytm Boyera-Moore'a – podsumowanie

### Algorytm Boyera-Moore'a:

- wymaga wstępnego przetwarzania wzorca, które da się zrobić w czasie  $O(|pat|)$
- korzysta z  $O(|pat|)$  dodatkowej pamięci na mapę `lastOccurrences`, na tablicę `goodSuffixes` i na kilka zmiennych
- jeśli tekst nie zawiera wzorca, to wykonuje nie więcej niż  $3|text|$  porównań znaków
- jeśli tekst zawiera wzorzec, to może wykonywać  $O(|text||pat|)$  porównań znaków
- w najlepszym przypadku wykonuje  $O(|text|/|pat|)$  porównań znaków

Algorytm Knutha-Morrisa-Pratta:

- wymaga wstępnego przetwarzania wzorca, które da się zrobić w czasie  $O(|pat|)$
- korzysta z  $O(|pat|)$  dodatkowej pamięci na tablicę  $p$  i na kilka zmiennych
- wykonuje w głównej pętli nie więcej niż  $2|text|$  porównań znaków
- wraz ze wstępnym przetwarzaniem działa w czasie  $O(|text| + |pat|)$

## Algorytm Karpa-Rabina – podsumowanie

### Algorytm Karpa-Rabina:

- korzysta z haszowania
- wymaga wstępnego przetwarzania, które da się zrobić w  $O(|\text{pat}|)$  czasu
- korzysta z  $O(1)$  pamięci na kilka zmiennych
- wykonuje w głównej pętli  $O(|\text{text}|)$  operacji, jeśli prawdopodobieństwo kolizji funkcji haszującej jest ograniczone z góry
- wraz ze wstępnym przetwarzaniem działa w czasie  $O(|\text{text}| + |\text{pat}|)$
- może zostać uogólniony tak, żeby naraz wyszukiwał wiele różnych wzorców, które mają tę samą długość

## Pomysły, uwagi, pytania, sugestie

Proszę wysyłać podpisane pomysły, uwagi, pytania, sugestie na temat wykładów lub laboratoriów na adres [mgc@agh.edu.pl](mailto:mgc@agh.edu.pl)

lub wpisywać anonimowe pomysły, uwagi, pytania, sugestie pod adresem <https://tiny.cc/algorytmy-tekstowe>



## **Do zobaczenia na następnym wykładzie**

**Jego tematem będzie znowu wyszukiwanie wzorca w tekście**

---

## Źródła zdjęć

<https://www.cs.utexas.edu/users/boyer/>

[https://en.wikipedia.org/wiki/J\\_Strother\\_Moore](https://en.wikipedia.org/wiki/J_Strother_Moore)

<https://www-cs-faculty.stanford.edu/~knuth/>

<http://www.cs.cmu.edu/~jhm/short%20biography.htm>

[https://en.wikipedia.org/wiki/Vaughan\\_Pratt](https://en.wikipedia.org/wiki/Vaughan_Pratt)

[https://en.wikipedia.org/wiki/Richard\\_M.\\_Karp](https://en.wikipedia.org/wiki/Richard_M._Karp)

[https://en.wikipedia.org/wiki/Michael\\_O.\\_Rabin](https://en.wikipedia.org/wiki/Michael_O._Rabin)