

# Pracownia Technik Obliczeniowych

## Wyrażenia regularne

Paweł Daniluk

Wydział Fizyki

Wiosna 2017



# Wyrażenia regularne

## W teorii

Sposób definiowania języków regularnych, które mogą być rozpoznawalne przez automaty skończone.

## W praktyce

Wyrażenie składające się ze symboli  $\emptyset$ ,  $\epsilon$ ,  $+$ ,  $*$ ,  $)$ ,  $($  oraz znaków alfabetu.

- 1  $\epsilon$  – język zawierający słowo puste
- 2  $\emptyset$  – pusty język
- 3  $a$  – język zawierający słowo  $a$  (gdzie  $a$  jest literą należącą do alfabetu)
- 4  $w + v$  – suma języków
- 5  $w^*$  – domknięcie Kleenego (język zawierający wszystkie słowa, które można utworzyć sklejając słowa języka  $w$ )
- 6  $wv$  – konkatenacja (język zawierający wszystkie słowa utworzone przez sklejenie pewnego słowa  $w$  z pewnym słowem  $v$ )

# Przykłady

$$(0 + 1) * 0$$

$$(a + b) * baba(a + b)*$$

# Składnia stosowana w praktyce

informatyka teoretyczna	praktyka	komentarz
$\emptyset, \epsilon$	brak	w praktyce niepotrzebny
$(, )$	$(, )$	
$+$	$ $	
$0 + 2 + 4 + h + a + l$	$[024ha1]$	zakres znaków
$0 + \dots + 5 + a + \dots + h$	$[0-5a-h]$	zakres znaków
$a + \dots + z + A + \dots + Z$	$.$	dowolny znak z alfabetu
$*$	$*$	0 lub więcej wystąpień
$e + \epsilon$	$e?$	wyrażenie $e$ występuje 0 lub 1 raz
$ee^*$	$e^+$	wyrażenie $e$ występuje 1 lub więcej razy
brak	$\wedge$	początek łańcucha (lub wiersza)
brak	$\$$	koniec łańcucha (lub wiersza)
$eeee$	$e\{4\}$	określona liczba powtórzeń (tutaj 4)
$eeee(e + \epsilon)(e + \epsilon)(e + \epsilon)$	$e\{4,7\}$	zakres liczby powtórzeń (tutaj od 4 do 7)

## Escaping

W wielu kontekstach znaki  $( ) | \{ }$  i inne mają specjalne znaczenie. Wówczas trzeba je poprzedzić znakiem  $\backslash$ .

grep

```
man grep
```

## Wyszukiwanie

Podczas wyszukiwania (klawisz /) można posługiwać się wyrażeniami regularnymi.

## Substytucje

Polecenie `s/<wzorzec>/<nowy>/` służy do zastępowania wystąpień wzorca nowym napisem. Można je stosować do całego tekstu, pojedynczej linii lub do zaznaczenia.

## Kilka miłych udogodnień

- `\s`, `\S` – spacja, nie-spacja
- `\d`, `\D` – cyfra, nie-cyfra
- `\w`, `\W` – znak słowny, znak nie-słowny
- `\1`, `\2`, ... – odwołania do elementów dopasowanych do poprzednich nawiasów
- `\<`, `\>` – początek/koniec słowa

sed

```
man sed
```

# Zadanie 0

Podaj wyrażenia regularne opisujące:

- kod pocztowy
- numer telefonu
- liczbę rzeczywistą
- deklarację funkcji w Pythonie (do dwukropka)



# Zadanie 1

Z pliku PDB wybierz:

- wszystkie atomy
- wszystkie heteroatomy
- atomy i heteroatomy (w kolejności w jakiej występują w pliku)
- węgle CA
- atomy o jednej ze współrzędnych w przedziale  $100.00 \div 199.999$
- atomy o równej części całkowitej współrzędnych X i Y

## Zadanie 2

Otwórz w vim'ie program napisany w Pythonie.

- znajdź przypisania postaci `a = a <op> <wyr>`
- w przypisaniach zamień lewą stronę z prawą
- użyj wyrażeń regularnych, aby automatycznie wygenerować metodę `__init__` przypisującą argumenty na atrybuty obiektu o tych samych nazwach