Regulární výrazy II

Práce s daty, 2. března 2023

Tři části regulárního výrazu

1. Vyhledávací (přiřazovací) maska
2. Substituční (nahrazovací) předpis
3. Modifikátory regulárního výrazu

Jednořádkový zápis:

/maska/substituce/modifikátory

Např. obalení slov uvozovkami:

```
/[a-z]+/"$&"/gmi
```

Modifikátory regulárního výrazu

Modifikátor = písmeno na konci RE (za zvoleným oddělovačem):

- `g` = *global* – hledá všechny shody v zadaném textu
- `m` = *multiline* – `^` a `$` označují začátek a konec každého řádku
- `s` = *singleline* (*dotall*) – metaznak tečka vyhovuje i konci řádku
- `i` = *insensitive* – nerozlišuje se velikost písmen
- `U` = *ungreedy* (*lazy*) – líný způsob vyhledávání v textu
- `u` = *unicode* – podpora národních abeced

Modifikátory lze kombinovat, např. `/gmiU`

Líný versus lačný způsob vyhledávání

Text: `<b>tučně</b> normálně <b>tučně</b>`

RE: `<b>.*<\/b>`

Greedy (lačný, nenasytný) – *as much as possible*

- Hledá co nejdelší řetězec vyhovující zadání --> vyhoví celý řetězec

Lazy (líný, pohodlný) – *as much as needed*

- Hledá co nejkratší řetězec --> vyhoví jenom `<b>tučně</b>`
- Líný může být i jenom kvantifikátor: `a+?`, `a*?`, `a{3,8}?`

Unicode rozšíření v regulárních výrazech

Přidává třídu metaznaků `\p{…}`

- `\p{L}` - písmena: `Ll` - malá, `Lu` - velká, `Lo` -- ostatní, …
- `\p{N}` - čísla: `Nd` - desítková, `Nl` - písmenková (Ⅷ), `No` - ostatní (¼), …
- `\p{S}` - symboly: `Sc` - měny (€, ¥), `Sm` - matematické, `So` - ostatní, …
- a dále interpunkce, oddělovače, kontrolní znaky, …

V závislosti na použitém nástroji může být rozšířen význam `\w`

Více informací např. na https://javascript.info/regexp-unicode

Substituce

Nalezená shoda bude nahrazena uvedeným předpisem:

- `/Ing/Mgr/` – titul *Ing.* bude nahrazen titulem *Mgr.*
- `/Bc\. ?//` – titul *Bc.* bude smazán (tj. předpis může být prázdný)

V substituci se lze odkazovat na nalezenou shodu či skupinu:

- `/(Ing|Mgr)/Dipl. $1/` – *Ing.* → *Dipl. Ing.* a *Mgr.* → *Dipl. Mgr.*
- `/(.*)\t(.*)/$2\t$1/` – prohození pořadí hodnot
- Starší způsob odkazování používá zpětné lomítko, např. \1, \2, …
  - Tímto způsobem se lze odkazovat i v rámci vyhledávací části RE!

Cvičení I

- Zapište RE, které budou validovat
  1. Jméno a příjmení včetně diakritiky
  2. Dvakrát to stejné slovo za sebou

- Zapište RE včetně substituce, který…
  1. Vymaže značky z HTML dokumentu
     - A) všechny, B) jen ty bez dalších vnořených značek
  2. Přeformátuje datum z tvaru 20230302
     - A) do podoby 02. 03. 2023 B) do podoby 2. 3. 2023

Cvičení I – řešení

- Jméno a příjmení s diakritikou – `\p{Lu}\p{Ll}+ \p{Lu}\p{Ll}+`
- Stejné slovo dvakrát za sebou – `\s(.+)\s+\1`
- Výmaz všech značek HTML – `<(.+)>/$2/gmU`
- Výmaz značek bez dalších vnořených značek
  - `<(.+)>(.*)<\/\1>/$2/gmU`
- Přeformátované datum
  - S nulami – `^(\d{4})(\d{2})(\d{2})$/$3. $2. $1/gm`
  - Bez nul – `^(\d{4})(0(\d)|(1[0-2]))(0(\d)|([1-3]\d))$ /$6$7. $3$4. $1/gm`

Konstrukce RE začínající znaky `(?`

Pomocí vnořených částí RE ve tvaru `(?  …  )` se tvoří

- Vnořené modifikátory
  - `(?isx-m)` – lze je zapínat a vypínat v průběhu RE
- Rozhlédnutí – *lookarounds*
  - `(?=  …  )` – *lookahead* (koukáme dopředu); `(?!  …  )` – negace podmínky
  - `(?<=  …  )` – *lookbehind* (koukáme zpět); `(?<!  …  )` – negace podmínky
- Skupiny, jejichž obsahuje si nepotřebujeme pamatovat
  - `(?:  …  )` – v angličtině *non-capturing groups*
- Testování podmínky – *conditionals*
  - `(?(A)B)` nebo `(?(A)B|C)` – pokud platí A, pak B (jinak C)

Příklady použití *lookarounds*

- `(?=rum)` – vyžaduje, aby od stávající pozice následoval *rum*
- `(?!rum)` – vyžaduje, aby od stávající pozice nenásledoval *rum*
- `(?<=rum)` – vyžaduje, aby to, co bezprostředně předchází stávající pozici, byl *rum*
- `(?<!rum)` – vyžaduje, aby to, co bezprostředně předchází stávající pozici, nebyl *rum*
- `^(?=.*A).*$` – vybere celý řádek, který obsahuje písmeno *A*
- `(?i)(?![áé])\p{L}` – vybere každé písmeno kromě *á*, *é* a *Á*, *É*
  - Ale pozor, modifikátor pro *unicode* musí být povolen až na konci RE!

Cvičení II

1. Otestujte, zda zvolené heslo vyhovuje kritériím:
   - alespoň 8 znaků dlouhé
   - alespoň jedno číslo
   - alespoň jedno velké písmeno

2. Jmenný seznam učitelů s jejich tituly rozdělte do tří sloupců
   - Tituly před jménem
   - Samotné jméno bez titulů
   - Tituly za jménem

Cvičení II – řešení

- Otestování hesla

```
^(?=\w{8,}$)(?=[^A-Z]*[A-Z])(?=\D*\d).*
```

- Rozdělení jmenného seznamu s tituly do tří sloupců

```
((?:Bc\. |Ing\. |Mgr\. |prof\. |doc\. |RNDr\. )*)
(?<=\. |^)([^,\n]+)(?:, (.+))?
$1\t$2\t$3
gm
```

Kde se RE dají používat

- Programovací jazyky
  - Java, JavaScript, Python, PHP, .NET, C#, Perl, …
- SQL databáze
  - MariaDB, MySQL, PostgreSQL, Oracle, …
- Analyzátory dat
  - R, Matlab, …
- Editory
  - SciTe, EditPad Lite, LibreOffice
- a na mnoha dalších místech, kde se pracuje s daty