

# /PCRE ir PHP/

## Perl Compatible Regular Expressions ir PHP

Emilis Dambauskas  
emilis@gildija.lt  
2004-04-20

# /(Kas•yra•PCRE\?)|(Kodėl•verta•jį•išmokti\?)/

## → Pavadinimai:

- Reguliariosios išraiškos
- Įprastinės išraiškos (Įpraiškos)
- Regular Expressions (RegExps)
- Šablonai
- Patterns

→ Galingas įrankis teksto apdorojimui

→ Standartinis PHP modulis

→ Plačiai paplitę  
(grep, Apache, JavaScript, redaktoriuose...)

# /Sveikas, •pasauli!/ ---

→ \$pattern

→ /Sveikas,•pasauli!/  
→ {Sveikas,•pasauli!}

→ /as,•pa/

→ /s/

→ /sveikas,•pasauli!/  
→ /•Sveikas,•pasauli•!/  

---

→ \$string

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli!”  

---



# /(Metasimboliai | Metacharacters)/

## Metasimboliai:

→ /,•./

→ /^./

→ /.\$/

→ /\../

→ /\n/

→ /\s/

→ /\w\\,•\d\d/

. ^ \$ \ \* + ? { } [ ] ( ) |

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli.”

→ “Sveikas,↵  
pasauli!”

→ “Sveikas,•pasauli!”

→ “Sveikas\\,•36•pasauli!”

# / (Kvantifikatoriai | Quantifiers) • 1 /

Kvantifikatoriai:

? \* + {...}

→ /Sveikas,•pasauli!\_?/

→ “Sveikas,•pasauli”

“Sveikas,•pasauli!”

“Sveikas,•pasauli!!!!”

→ /Sveikas,•pasauli!\_+/

→ “Sveikas,•pasauli”

“Sveikas,•pasauli!”

“Sveikas,•pasauli!!!!”

→ /Sveikas,•pasauli!\_\*/

→ “Sveikas,•pasauli”

“Sveikas,•pasauli!”

“Sveikas,•pasauli!!!!”

→ /k\_.\*u/

→ “Sveikas,•pasauli!!!!”

# / (Kvantifikatoriai | Quantifiers) • 2/

→ /Sveikas{2,3},•pasauli!/

→ “Sveikas,•pasauli!”

“Sveikass,•pasauli!”

“Sveikasss,•pasauli!”

“Sveikasssss,•pasauli!”

→ /Sveikas{2,},•pasauli!/

→ “Sveikas,•pasauli!”

“Sveikass,•pasauli!”

“Sveikasss,•pasauli!”

“Sveikasssss,•pasauli!”

...

→ /Sveikas{,2},•pasauli!/

→ “Sveika,•pasauli!”

“Sveikas,•pasauli!”

“Sveikass,•pasauli!”

“Sveikasss,•pasauli!”

# / (Ženklių•klasės) | (Character•class) /

Ženklių klasės:

→ `/[ks]a/`

→ `/[a-z]+/`

→ `/[-a-zA-Z]+/`

→ `/[^a-z]+/`

→ `/[a-z^]+/`

→ `/[A-Z0-9]{2,}/`

[...], [...-...], [^...]

→ “Sveikas,•pasauli!”

→ “Sveikas,•pasauli!”

→ “Svei-kas,-•pasauli!”

→ “Sveikas,•pasauli!”

→ “Sveikas,^•pasauli!”

→ “Sveikas,•R2D2!”

# /Grupavimas, • (Atgalinės nuorodos | Backreferences) /

→ /(ne)?sveikas•pasaulis/

→ “sveikas•pasaulis”  
“nesveikas•pasaulis”

/(sveikas|nesveikas)•pasaulis/

“gražus•pasaulis”

## Atgalinės nuorodos

→ /Paksui•(taip|ne)!•LDP•\1!/

→ “Paksui•taip!•LDP•taip!”  
“Paksui•ne!•LDP•ne!”

/(Paksui•(taip|ne))!•LDP•\2!/

“Paksui•taip!•LDP•ne!”

→ Nesumaišykite su aštuntainiais kodais!

^44/

→ “Noriu•\$\$\$”



/Ar•[Jj]ūs•vis•dar•čia\?/



# / (Modifikatoriai | Modifiers) • 1 /

→ /sveikas,•pasauli/i

→ “Sveikas,•pasauli!”  
“SvElkAs,•PaSAuLi!”

→ /^pasauli/

→ “sveikas,•¶  
pasauli”

→ /^pasauli/m

→ “sveikas,•¶  
pasauli”

→ /sveikas.\* /

→ “sveikas,•¶  
pasauli”

→ /sveikas.\* /s

→ “sveikas,•¶  
pasauli”

# / (Modifikatoriai | Modifiers) • 2 /

→ /sveikas,pasauli!/x

→ “sveikas,pasauli!”

→ /.../S

→ Optimizuoja šablona

→ /.../u

→ UTF-8 suderinamumas

...

→ Plačiau:

PHP manual:

Regular Expression Functions  
(Perl-Compatible):  
Pattern Modifiers

# /Pastabos•1/

→ Naujos eilutės simbolis skiriasi skirtingose OS:

- \n — UNIX style
- \r\n — Windows style
- \n\r — Macintosh style

/[\r\n]+/

```
/preg_.*\(\)\.1/
```

`preg_grep($pattern, $input)`

Tikrina masyvo `$input` elementus

`preg_match($pattern, $string, [$matches])`

Tikrina eilutę, rezultatus iškerpa į masyvą `$matches`:

```
$pattern = '/((sveikas),\.\.)(pasauli)/i';  
$string = 'Sveikas,\.\.pasauli!';  
$matches == array(0 => 'Sveikas,\.\.pasauli',  
1 => 'Sveikas,\.\.',  
2 => 'Sveikas',  
3 => 'pasauli');
```

```
/preg_.*\(\)\.2/
```

`preg_match_all($pattern, $string, $matches)`

Suranda ne tik pirmą atitinkančią eilutę, bet ir visas kitas eilutes. Gražina dvimatį masyvą:

```
$matches = array(0=> array('pattern1', 'pattern2'),  
                 1=> array('p1sub1', 'p2sub1'),  
                 ...);
```

`preg_replace($pattern, $replacement, $string)`

Pakeičia atitikmenis:

```
$string = ('/dog/', 'cat', 'I have a dog.');
```

```
$string == 'I have a cat.';
```

# /Keletas • pavyzdžių/

→ Patikrinti el. pašto adresą:

```
/[a-z0-9_.-]+@[a-z0-9_]+\.[a-z]{2,4}/i
```

→ Sukurti nuorodas iš URL'ų:

```
$pattern =
```

```
'|(http://([a-z0-9_]+\.[a-z]{2,4}(/[a-z0-9._~/?;])?)|)i';
```

```
$replacement = '<a href="\1">\1</a>';
```

```
$string =
```

```
preg_replace($pattern, $replacement, $string);
```

→ Išskaidyti tekstą į žodžius:

```
$words = preg_split('/^[^a-z]+/i', $text);
```

# /Pastabos•2/

→ Neužmirškite standartinių PHP funkcijų:

`explode() / implode()`

`str_replace()`

`strstr()`



# /Nuorodos/

## → Žinios:

- <http://www.php.net/manual/en/ref.pcre.php>
- <http://lietuvoje.lt/~moxliukas/pcre.html>
- <http://zez.org/article/articleprint/11/>
- Mastering Regular Expressions  
<http://regex.info/>

## → Naudinga:

- Visual RegExp  
<http://laurent.riesterer.free.fr/regexp/>
- <http://regexlib.com>

