Programování v PHP

Katedra softwarového inženýrství
Fakulta informačních technologií
České vysoké učení technické v Praze

Regulární výrazy
BI-PHP

Regulární výrazy - testování, vyhledávání a úpravy řetězců dle masky (reg. vyrazu)
(Zde se pracuje s regulárními výrazy ve stylu jazyka Perl. Lze používat i styl POSIX)

Zakladni metaznaky regularnich vyrazu:

```
/^abc$/       ^ - zacatek retezce, $ - konec
/^ab*c$/      * - opakovani predchoziho znaku nebo skupiny znaku 0,1,2,...
/^ab+c$/      + - opakovani 1,2,3,...
/^ab?c$/      ? - opakovani 0,1 (nepovinne)
/^ab{3,5}c$/  {m,n} - opakovani m,m+1,m+2,...,n
/^ab{3,}c$/   {m,} - opakovani m, m+1, m+2, ...
/^ab{3}c$/    {m} - opakovani m-krat
/^a(xyz)*c$/  (...) - skupina znaku pro opakovani, podvyraz
/^a(po|ut|st|ct|pa)c$/  (...|...|...) - kterakoli z moznosti
/^a[kxd]c$/   [...] - jeden z uvedenych znaku
/^a[^kxd]c$/  [^...] - jeden znak, ruzny od uvedenych
/^a[0-9]c$/   [...-...] - jeden znak ze zadaneho intervalu
/^a.c$/       . (tecka) - jeden libovolny znak
```

Nektere zkratky:

```
[0-9]              \d
[^0-9]             \D
whitespace         \s (mezera, tabelator, novy radek, ...)
not whitespace     \S
[A-Za-z0-9_]       \w
[^A-Za-z0-9_]      \W
word boundary      \b (napr. Java pro versus JavaScript)
not word noundary  \B
```

Priklady:

```
/^[0-9]{3}[ ]?[0-9]{2}$/      PSC s nepovinnou mezerou
/^\d{3}[ ]?\d{2}/             totez (se zkratkami)
/^[-+]?[1-9][0-9]*(,[0-9]+)?$/     cislo bez vedoucich nul s nepovinnym
                                   znamenkem a desetinnou casti
/^ab+c$/i /.../img  (i nerozlisovat mala/velka, m - mutiline, g - global)
/[\s]*,[\s]*/      carka, obklopena libovolnym poctem mezer (i nulovym)
/(['"])([^'"]*)\1/ retezec (neobsahujici ' nebo ") obklopeny znaky ' nebo "
                   (\1 - retezec, odpovidajici 1. podvyrazu (zavorce))
(\w+\.)*\w+@(\w+\.)+[A-Za-z]{2,3}  mailova adresa (zjednodusena)
Java(?=Script)     positivni predbezny prepoklad (shoda jen, je-li
                   nasledovano slovem Script; to se ale nepamatuje jako
                   podvyraz)
Java(?!Script)     negativni predbezny prepoklad (Java neni nasledovano
                   Script, ktere se nepamatuje)
```

Funkce pro regularni vyrazy, pouzivane v tomto programu (detaily viz help):

```
preg_match - zjistuje shodu daneho retezce (2. param) s regularnim
             vyrazem (1. param) a vraci true/false. Muze vracet i pole pro
             shody s ruznymi podvyrazy
preg_match_all - nalezeni vsech shod (vraci pole) - vcetne podvyrazu
preg_replace - nahradi v retezci vsechny shody s reg.vyrazem zadanou
               hodnotou
preg_split - rozdeli retezec na podretezce (vraci pole) dle zadaneho
             regularniho vyrazu
```

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Jednoduchy program pro praci s regularnimi vyrazy</title>
    </head>
    <body>

        <?php
            $retez = $valRetez = "";
            $re = $valRe = "";
            $replSplit = $valReplSplit = "";
            $vystup = "";

           if (!empty($_GET["btnTest"]) || !empty($_GET["btnReplace"]) ||
               !empty($_GET["btnSplit"])){ // stisknuto tlacitko?
                $retez = $_GET["txtRetez"];
                $re = $_GET["txtVyraz"];
                $replSplit = $_GET["txtReplSplit"];

                // Zapnuto vkladani backslashes? (novejsi verze PHP by nemely
                // pouzivat magic quotes - automaticke vkladani escape char)
                /*
                if (get_magic_quotes_gpc()){
                    // stripslashes zrusi pridane znaky \ (backslash)
                    $retez = stripslashes($retez);
                    $re = stripslashes($re);
                    $replSplit = stripslashes($replSplit);
                }
                */

                // str_replace pouzit k nahrazeni pripadnych uvozovek
                // v retezci entitou &quot; - aby po vlozeni do tagu
                // input (jako value) nebyla uvozovka povazovana za
                // konec hodnoty u atributu value!! Napr. value="/[;",]/"
                $valRetez = str_replace('"','&quot;',$retez);
                $valRe = str_replace('"','&quot;',$re);
                $valReplSplit = str_replace('"','&quot;',$replSplit);
            }
        ?>

        <form action="re_test.php" method="get">
            Retezec:<br>
            <input type="text" name="txtRetez" size="80"
                   value="<?php echo $valRetez;?>"><br>
            Reg. exp.:<br>
            <input type="text" name="txtVyraz" size="80"
                   value="<?php echo $valRe;?>"><br>
            Replace/Split:<br>
            <input type="text" name="txtReplSplit" size="80"
                   value="<?php echo $valReplSplit;?>"><br>

            <input type="submit" name="btnTest" value="Test">
            <input type="submit" name="btnReplace" value="Replace">
            <input type="submit" name="btnSplit" value="Split"><br><br>
        </form>
```

```
        <?php
            if (!empty($_GET["btnTest"])){
                // testuje shodu
                // $vysledek = preg_match($re, $retez, $shody);

                // nalezne vsechny shody
                $vysledek = preg_match_all($re, $retez, $shody);

                if ($vysledek)
                    echo "<b>Retezec odpovida regularnimu vyrazu</b><br>";
                else
                    echo "<b>Retezec NEodpovida regularnimu vyrazu</b><br>";

                $vystup = "";
                foreach ($shody[0] as $shoda){ // pole vsech shod s celym re
                    $vystup .= $shoda."\r\n";
                }
            }
            elseif (!empty($_GET["btnReplace"])){
                // nahrazeni v retezci dle reg. vyrazu
                $vystup = preg_replace($re,$replSplit,$retez);

                // Priklad 1 - nahrazeni mezery, obklopene libovolnym
                // poctem mezer carkou a mezerou
                // $text = "Prvni   ,   Druhy  , Treti";
                // echo "<br>".preg_replace("/[\s]*,[\s]*/",", ",$text)."<br>";

                // Priklad 2 - nahradit uvozovky nebo apostrofy, obklopujici
                // retezce (ktere neobsahujici apostrof ani uvozovku)
                // hranatymi zavorkami
                // $text = "\"Prvni\",'Druhy','Treti'";
                // echo "$text<br>";
                // \1 - prvni nalezeny podvzor = cast v zavorkach
                // (pro regularni vyraz)
                // $2 - druhy nalezeny podvzor (pro nahrazovani)
                // echo "<br>".preg_replace("/(['\"])([^'\"]*)\\1/",
                //                         "[$2]",$text)."<br>";
            }
            elseif (!empty($_GET["btnSplit"])){
                // Priklad - rozdeleni slov s libovolnym poctem mezer
                // jako oddelovacem
                // $text = "Prvni   Druhy   Treti";
                // rozdeli retezec podle regularniho vyrazu
                // $casti = preg_split("/[\s]+/",$text);
                // echo "<br>".implode(",",$casti)."<br><br>";

                // rozdeleni retezec podle regularniho vyrazu
                $casti = preg_split($replSplit,$retez);

                $vystup = "";
                foreach ($casti as $cast){
                    $vystup .= $cast."\r\n";
                }
            }
        ?>
        <textarea name="taShody" rows="10" cols="80">
                                <?php echo $vystup?></textarea>
    </body>
</html>
```