# Лабораторная работа №4. Регулярные выражения в PHP.

**Цель работы:** изучение основ регулярных выражений и их использования в языке программирования PHP.

**Порядок выполнения работы**

1. Изучить тему 3.11 лекционного материала.

2. Выполнить задание по лабораторной работе (во всех вариантах использование регулярных выражений является ОБЯЗАТЕЛЬНЫМ!).

3. Представить для проверки результат выполнения работы в виде файлов и демонстрации выполнения на компьютере

**Рекомендации по выполнению заданий**

* Основное требование при выполнении данной работы – универсальность и ориентированность на производительность.
* Не допускается выполнение данной лабораторной работы в виде, когда программа реагирует лишь на некоторые частные случаи вхождения искомых элементов в текст. Следует помнить, что отдельные части искомых элементов могут быть разделены переносами строк, пробелами и иными символами, которые автор документа мог использовать для форматирования текста.
* Производительность скриптов следует анализировать, разрабатывая несколько альтернативных вариантов и выбирая наиболее быстродействующий.
* В некоторых случаях алгоритмически более простым является выполнение задачи в несколько этапов с предварительным приведением данных к формату, позволяющему использовать более простые регулярные выражения, нежели в случае анализа исходного текста без предобработки.
* Ещё одним важным показателем качества программы является использование оперативной памяти. Поскольку вложенные регулярные выражения приводят к рекурсивным вызовам и геометрической прогрессии занимаемого анализируемыми данными объёма оперативной памяти, следует предусмотреть особое поведение программы для случаев, когда объём входных данных приближается к объёму доступной программе оперативной памяти.
* В предложенных выше вариантах заданий нет строгого требования к анализу текста в различных кодировках, однако при выполнении задания рекомендуется предусмотреть представление входных данных в кодировке UTF8.
* Во всех вариантах заданий результатом работы программы должна являться корректная HTML-страница, содержащая как исходный, так и преобразованный согласно заданию текст.

**Задания по вариантам**

**Вариант 1:** Напишите регулярное выражение которое будет определять корректно адреса электронной почты в формате, например, name@server.сom, name.surname@subserver.server.сom и т.д.. Поместите его в функцию, принимающую аргументом строку с электронной почтой. Возвратите соответствующие значения результата проверки. Данные принимать через форму (Имя, E-mail) . Корректные адреса вместе с именем записывать в текстовый файл.

**Вариант 2:** Написать программу оставления отзывов. Форма: Имя, Сообщение. Отправленные сообщения хранить в файле и выводить в блоке выше формы. Все ссылки в отправляемом сообщении типа http://mysite.by, https://www.mysite.by/price и т.д. должны заменяться на текст #Внешние ссылки запрещены#. Ссылки на сайт БГУИР типа http://bsuir.by, https://www.bsuir.by/ru/kafedry-bguir и т.д. остаются в тексте. Вся проверка в одном регулярном выражении.

**Вариант 3:** Написать программу оставления отзывов. Форма: Имя, Сообщение. Отправленные сообщения хранить в файле и выводить в блоке выше формы. Все упоминания e-mail в отправляемом сообщении типа mymail@mail.com, name.surname@subserver.server.сom и т.д. должны заменяться на текст #Cтоп e-mail#. Упоминание почты БГУИР mymail@bsuir.by, mymail.poit@bsuir.by и т.д. остаются в тексте. Вся проверка в одном регулярном выражении.

**Вариант 4:** в произвольном тексте все целые числа вывести синим цветом, все дроби вывести красным цветом и округлить до десятых, все слова, начинающиеся с большой буквы, вывести зелёным цветом. Текст вводить через форму.

**Вариант 5:** в произвольном тексте все URL'ы вывести красным цветом и привести к виду <a href="URL">URL</a>. Если до преобразования присутствовала человекочитаемая часть URL'а, выводить URL в виде <a href="URL">URL; человекочитаемая\_часть</a>. Текст загружать из файла.

**Вариант 7:** в произвольном тексте все номера телефонов (предусмотреть не менее пяти вариантов записи номера) вывести зелёным цветом. При этом номера сотовых телефонов (начинаются с "+КОД-") подчеркнуть. Текст вводить через форму.

**Вариант 8:** в произвольном тексте все даты (в формате DD.MM.YYYY и MM/DD/YYYY, причём день и месяц могут быть однозначными, а год – двузначным) вывести красным цветом, при этом увеличить год на единицу. Формат MM/DD/YYYY привести к DD.MM.YYYY также с использованием регулярного выражение. Текст получать из файла.

**Вариант 9:** в произвольном тексте все слова, состоящие из английских букв, вывести синим цветом, все слова, состоящие из русских букв, вывести красным цветом, все числа вывести зелёным цветом. Текст вводить через форму.

**Вариант 10:** в произвольном тексте последовательности из двух и более пробельных символов заменить на один пробел, каждое предложение оформить в виде отдельного абзаца, все аббревиатуры (например ОАО, АСУ и т.п.) подчеркнуть, все числа вывести синим цветом. Текст получать из файла.

**Вариант 11:** в произвольном HTML-документе все заголовки вывести синим, все наклонные фрагменты текста вывести зелёным, все жирные фрагменты текста вывести красным.

Текст получать из файла.

**Вариант 12:** В произвольном тесте (длинный текст) выполнить. При вводе слова в форму поиска необходимо найти все упоминания этого слова в тексте и выделить (подсветить) цветом, жирным или другим настраивающим способом. В случае, если указываются 2 слова, то каждое должно искаться индивидуально, если словосочетание указывается в кавычках, то ищется как единое словосочетание. Помимо грубого поиска так же должен выполнятся поиск слова с разными окончаниями: Родина, Родины, Родиной.. Искомое слово получить через веб-форму.

Текст получать из файла.

**Вариант 13**: Дан длинный текст, в нём встречаются слова длиннее 7 символов! Если слово длиннее 7 символов, то необходимо: оставить первые 6 символа и добавить звёздочку. Остальные символы вырезаются. Шаблон: "я стану крутым программистом после БГУИРа" . Результат: " я стану крутым програм\* после БГУИРа ". Текст вводить через форму.

**Вариант 14**: в произвольном тексте все e-mail адреса вывести красным цветом и привести к виду <a href="mailto:EMAIL">EMAIL</a>. Также список e-mail адресов сохранить в файл. Текст вводить через форму.

**Вариант 15**: в произвольном тексте все слова, начинающиеся с большой буквы, но не стоящие в начале предложения, вывести красным цветом, а все такие слова, стоящие в начале предложения подчеркнуть. Набор цифр более 3 подряд вывести зеленым цветом. Текст вводить через форму.

**Расширенное задание**

Написать шаблонизатор (программу, управляющую сборкой готовых HTML-страниц из отдельных шаблонов). Шаблонизатор должен уметь обрабатывать следующие инструкции:

{FILE="path\_to\_file"} – чтение и подстановка указанного файла;

{CONFIG="value"} – чтение и подстановка значения из конфигурационного файла;

{VAR="variable\_name"} – подстановка значения из массива $VARS, формируемого в процессе работы приложения;

{DB="value"} – подстановка значения из предопределённой таблицы в БД, хранящей текстовые надписи, настройки приложения и тому подобную информацию;

{IF "var\_1"</>/==/!=/<=/>="var2"} PART1 {ELSE} PART2 {ENDIF} – анализ условия и удаление из шаблона той части, которая не соответствует условию; условия могут быть вложенными; часть {ELSE} может отсутствовать.

**Полезные ссылки:**

https://www.php.net/manual/ru/book.pcre.php

https://www.php.net/manual/ru/reference.pcre.pattern.syntax.php

Аврамец Дмитрий Владимирович (ПОИТ)

dva@bsuir.by