Regular Expressions

Set Theory Jargon

An **alphabet** is a set of characters.

A **(formal) language** is a set of strings of characters over an alphabet;

**Regular expressions** can be used to specify some sets of character strings, and formal languages that can be represented by a regular expression are called **regular languages**.

Computer Programming and Regular Expressions

Regular expressions are used for **lexical analysis**.

* They are used by programming languages and programmers to define **sets of strings**.

Validating User Input

Regular expressions are commonly used by web programmers to **validate user input** in web forms.

The programmer:

|  |  |
| --- | --- |
|  | Constructs a regular expression for invalid strings (or characters). |
|  | Writes code to check user input against the regular expression before it is submitted to the backend. |

Regular expressions can be constructed by combining **literals** and **meta characters** to refer to sets of strings.

|  |  |  |
| --- | --- | --- |
|  | Regular Expression | Meaning |
| String Literal | ”hello” | The exact sequence of characters: hello |
| Character Sets | ”[abcde5]” | Any character from the set  { a, b, c, d, 5} |
| Character Ranges | ”[a-g]” | Any character from the set  { a, b, c, d, e, f, g } |
| Negation | ”[^1-5]” | Any character not in the set  { 1, 2, 3, 4, 5 } |
| Starts With | ”^[1-5]” | Starts with a character contained in the set  { 1, 2, 3, 4, 5 } |
| Ends With | ”[a-c]$” | Ends with a character contained in the set  { a, b, c } |
| Optional | ”behaviou?r” | The exact sequence of characters: ”behaviour” – where ’u’ is optional. |
| Sequences | ”[0-9]+” | A sequence of 1 or more integers. |
|  | ”[0-9]\*” | A sequence of zero or more integers. |
|  | ”[0-9]{5}” | A sequence of 5 integers |

Complex Regular Expressions

The basic regular expressions can be combined to refer to specific sequences of characters.

|  |  |
| --- | --- |
| ”^[0-9]{2}$” | Start with (^) and end with ($) two integers. |
| ”^[0-9]{2}[ ]?[0-9]{2}$” | * Starts with (^) two integers * Ends with ($)two integers * Has an optional (?)space character in-between, i.e. 12 44 |
| ”^[a-z][0-9]+.[0-9]{2}$” | * Starts with one letter * Followed by at least one (+) integer * Followed by a ‘.’ * Ends with two integers, i.e. a133.55 |

Regular Expression and Javascript

JavaScript has 3 **library functions** for testing strings against regular expressions and/or returning matches. Here are simple examples of their use.

|  |  |
| --- | --- |
| test() | var pattern = /var/;  var str = "var x = 55;";    if(pattern.test(str)){  document.write("Pattern matched using pattern.test(str) <br/>" );  } |
| exec() | **var** pattern = /var/;  **var** matches = pattern.exec(str);  **if**(matches){  document.write("Matches found using pattern.exec(str): " + matches + "<br/>");  } |
| match() | **var** patternStr = "var";  matches = str.match( patternStr );  **if**(matches){  document.write("Matches found using str.match(patternStr): " + matches + "<br/>");  } |