Regular Expressions

Regular Expressions

- Used to match text rather than filenames
- Similar to glob expressions, but much more powerful
- Uses:
  - Search files for text
  - Edit text (patterns for search and replace)
  - Validate input in programs, webforms, etc
  - Match lines to control AWK scripts
  - Specify elements of the syntax of a programming language
- Most regular expressions used with bash must be quoted

Regular Expression Syntax (1)

- `*` matches 0 or more of the preceding expression
- Examples

Regex	Matches
a*	a, aa, aaa, aaaa, etc, and the empty string
abc*	ab, abc, abcc, abccc, etc
ab*c	ac, abc, abbc, abbbc, etc
(ab)*	ab, abab, ababab, etc, and the empty string (Parentheses are used for grouping)

Regular Expression Syntax (2)

- `+` matches 1 or more of the preceding expression
- Examples

Regex	Matches
a+	a, aa, aaa, aaaa, etc
abc+	abc, abcc, abccc, etc
ab+c	abc, abbc, abbbc, etc
(ab)+	ab, abab, ababab, etc

Regular Expression Syntax (3)

- `.` matches any single character (except newline)
- `[ ]` is the same as in glob expressions
- `?` makes the previous expression optional (0 or 1 occurrences)
- Examples:

Regex	Matches
`[a-z].*`	Strings beginning with a letter
`[0-9]+`	Integers
`[^0-9]*`	Strings that don't contain a digit
`.?.?.?`	Strings of length 0, 1, 2, or 3

Regular Expression Syntax (4)

- `{n}` matches n of the previous expression
- `{m,n}` matches m through n of the previous expression
- `{n,}` matches n or more of the previous expression
- `{,n}` matches n or fewer of the previous expression
- r_1`|`r_2 matches like r_1 or r_2

Regex	Matches
`.{5}`	strings of length 5
`(.{3})*`	strings whose length is a multiple of 3
`foo|bar`	matches foo or bar
`image\.(png|jpg)`	matches image.png or image.jpg

Using Regular Expressions with Egrep

- `egrep regexp` displays lines from stdin containing a match for regexp
- `egrep regexp file ...` displays lines from the files matching a match for the regexp
- Egrep Options

`-i`	ignore case
`-v`	display lines that don't match
`-c`	count the lines that match
`-l`	display the names of files with a match
`-q`	print nothing - set exit status to 0 if there's a match otherwise 1
`-x`	Only display lines that are an exact match (instead of containing a match)

Egrep Examples

- `egrep foobar infile` prints lines from infile containing foobar
- `egrep '^......$' infile` prints lines from infile of length 6
- `egrep '^ ' infile` prints lines from infile that start with a space
- `egrep '^[0-9]+$'` prints nonempty lines from stdin containing only digits