

A *regular expression* is a sequence of characters that specifies a set of strings, which are said to *match* the regular expression.

For example, in one flavor of regular expression syntax:

`gli..ering` → set of strings that begin with "gli",
followed by any two characters,
followed by "ering"

grep

vi/emacs/other text editors

most command shells (e.g., csh, bash, Windows shell)

many programming languages

Unfortunately, this does not imply that all use the same syntax rules for REs.

For historical reasons, there are many variations (flavors) of RE syntax.

For the sake of sanity, we will restrict ourselves to the grep flavor.

Most characters simply stand for themselves.

Metacharacters have special meaning:

period (.)

matches any single character

a.c is matched by aac, abc, a)c, etc.

b..t is matched by beet, best, boot, bart, etc.

asterisk (*)

matches zero or more occurrences of the preceding RE

ab*c is matched by ac, abc, abbc, abbbc, etc.

.* is matched by all strings

plus (+)

matches one or more occurrences of the preceding RE

ab+c is matched by abc, abbc, abbbc, etc., but not by ac

```
$ grep -E gli..er MobyDick.txt
```

a fine frosty night; how Orion glitters; what northern lights! Let them glittering teeth resembling ivory saws; others were tufted with knots of footfall in the passage, and saw a glimmer of light come into the room glittering in the clear, cold air. Huge hills and mountains of casks on glittering expression--all this sufficiently proclaimed him an inheritor looked celestial; seemed some plumed and glittering god uprising from suddenly relieved hull rolled away from it, to far down her glittering the wife sat frozen at the window, with tearless eyes, glitteringly glittering fiddle-bows of whale ivory, were presiding over the hilarious to glimmer into sight. Glancing upwards, he cried: "See! see!" and once At the first faintest glimmering of the dawn, his iron voice was heard leeward; and Ahab heading the onset. A pale, death-glimmer lit up glittering mouth yawned beneath the boat like an open-doored marble methodic intervals, the whale's glittering spout was regularly announced the moment, intolerably glittered and glared like a glacier; and

Note the use of the `-E` switch in the example here. This specifies to `grep` to use certain extensions to the basic RE syntax; rather than fuss about the difference, we will simply invoke `grep` with this switch in all cases.

```
$ grep -n -E fe+d MobyDick.txt
```

```
278:      Tho' stuffed with hoops and armed with ribs of whale."  
746:I stuffed a shirt or two into my old carpet-bag, tucked it under my arm,  
1267:Whether that mattress was stuffed with corn-cobs or broken crockery,  
1381:he puffed out great clouds of tobacco smoke. The next moment the light  
1822:But Faith, like a jackal, feeds among the tombs, and even from these  
2644:How I snuffed that Tartar air!--how I spurned that turnpike earth!--that  
2903:Hosea's brindled cow feeding on fish remnants, and marching along the  
4929:own. Yet now, federated along one keel, what a set these Isolatoes were!  
. . .
```

Note the use of the `-n` switch in the example here. This specifies to `grep` to report line numbers along with the matching lines.

```
$ grep -E travel+er MobyDick.txt
```

```
the great New England traveller, and Mungo Park, the Scotch one; of all  
palsied universe lies before us a leper; and like wilful travellers in  
more travellers than in any other part.  
. . .
```

question mark (?)

matches zero or one occurrence of the preceding RE

ab?c is matched by ac and abc, but not by abbc

b.?t is matched by bt, bat, bet, bxt, etc.

logical or (|)

matches the RE before | or the RE after |

abc|def is matched by abc and def and nothing else

```
$ grep -E fee?d MobyDick.txt
```

```
    Tho' stuffed with hoops and armed with ribs of whale."  
I stuffed a shirt or two into my old carpet-bag, tucked it under my arm,  
Whether that mattress was stuffed with corn-cobs or broken crockery,  
he puffed out great clouds of tobacco smoke. The next moment the light  
. . .
```

```
$ grep -E 'equal|same' MobyDick.txt
```

```
and some other articles of the same nature in their boats, in order to  
"And pray, sir, what in the world is equal to it?" --EDMUND BURKE'S  
to have indirectly hit upon new clues to that same mystic North-West  
nearly the same feelings towards the ocean with me.  
. . .
```

Note the use of single-quotes around the RE in the second example; this is absolutely necessary in the Unix shell because the '|' character has special meaning to the shell and that takes priority; the same applies in the Windows shell except that double-quotes are used.

caret (^)

used outside brackets, matches only at the beginning of a line

`^D.*` is matched by any line beginning with D

see slide 10 for semantics if inside brackets...

dollar sign (\$)

matches only at the end of a line

`.*d$` is matched by any line ending with a d


```
$ grep -E ^equal MobyDick.txt
```

```
equalled by the realities of the whalemens.  
equally desolate Salisbury Plain in England; if casually encountering  
equal to that of the brain. Under all these circumstances, would it be  
equally doubted the story of Hercules and the whale, and Arion and the  
. . .
```

```
$ grep -E equal$ MobyDick.txt
```

```
twenty pounds; so that the whole rope will bear a strain nearly equal
```

The first example does not work properly in the Windows shell unless you put double-quotes around the RE.

backslash (\)

escapes other metacharacters

now\ . is matched by "now."

square brackets []

specify a set of characters as a set; any character in the set will match

[aeiou] is matched by any vowel

[a-z] is matched by any lower-case letter

^ specifies the complement (negation) of the set

[^aeiou] is matched by any character but 'a', 'e', 'i', 'o' and 'u'

parentheses ()

forms a group of characters to be treated as a unit

a(bc)+ is matched by abc, abcbc, abcbcbc, etc.

braces { }

specifies the number of repetitions of an RE

[a-z]{3} is matched by any three lower-case letters

```
$ grep -E 'equal(ly)?$' MobyDick.txt
```

```
twenty pounds; so that the whole rope will bear a strain nearly equal  
even now beholding him; aye, and into the eye that is even now equally
```

```
$ grep -E '^f[aeiou]t' MobyDick.txt
```

```
fathoms down, and 'the weeds were wrapped about his head,' and all the  
father was a High Chief, a King; his uncle a High Priest; and on the  
future investigators, who may complete what I have here but begun. If  
. . .  
fetch another for a considerable time. That is to say, he would then  
fathoms of rope; as, after deep sounding, he floats up again, and shows  
. . .  
fitted to sustain the weight of an almost solid mass of brick and  
fatal cork, forth flew the fiend, and shrivelled up his home. Now, for  
. . .
```

```
$ grep -E '^f[aeiou]+t' MobyDick.txt
```

```
foot of it. But I got a dreaming and sprawling about one night, and
footfall in the passage, and saw a glimmer of light come into the room
fathoms down, and 'the weeds were wrapped about his head,' and all the
feet high; consisting of the long, huge slabs of limber black bone taken
features of the leviathan, most naturalists have recognised him for one.
future investigators, who may complete what I have here but begun. If
faithfully narrated here, as they will not fail to elucidate several
fitted to sustain the weight of an almost solid mass of brick and
. . .
```

```
$ grep -E 'br(ing){2}' MobyDick.txt
```

```
myself involuntarily pausing before coffin warehouses, and bringing up
justified his bringing his harpoon into breakfast with him, and using it
bringing in good interest.
. . .
```

word boundaries (`\<` and `\>`)

specifies to only match entire words (in a loose sense)

`\<fat\>` is matched by "fat" but not "father" or "fathom"

```
$ grep -E '\<fat\>' MobyDick.txt
```

```
nothing certain. They grow exceeding fat, insomuch that an incredible
DUTCH SAILOR. Grand snoozing to-night, maty; fat night for that. I
exceeding richness. He is the great prize ox of the sea, too fat to be
. . .
```

Of course, `grep` doesn't "understand" English. Word boundaries are indicated by the beginnings and ends of alphanumeric sequences of characters.