grep_tutorial

This is a reference booklet for grep and regular expression. For explanation of various usages in detail please refer more elaborate guide. grep: Global Regular Expression Print. GNU grep is combination of basic regular expressions, extended regular expressions, fixed strings and Perl-style regular expressions. Default behavior of grep is to return the filename and the line of the test that contains the searched string. *Literals* are the normal text characters, whereas *metacharacters* have special meanings. Backtic (``) enclosed portion is interpreted. Double quotes ("") allow usage of environment variable as a part of search pattern.

There are two ways to search with grep i.e. searching for fixed string and searching for patterns. Concatenation is processed before alternation. Strings are concatenated by simply placing/being next to each other inside regular expression.

grep -E has advantage of accomplishing the task in fewer characters. If significant use of backreferences is required, grep -E is ideal.

grep -F, any search pattern for grep -F cannot contain any metacharacters, escapes, wildcards, or alternations.

Syntax usage of grep is as follows: grep [options] [regularexpression] [filename]

Metacharacter	Name	Matches
Single Character Match		
.	Dot	Any one character
[…]	Character class	Any one member of the character listed in brackets
[^…]	Negates character class	Any character not listed in bracket (any one)
\char	Escape character	Use the character after escape (\) literally (not interpreted).
Position Match		
^	Caret	Start of a line
$	Dollar	End of a line
\<	Backslash (less-than)	Start of a word
\>	Backslash (greater-than)	End of a word
Quantifiers		
?	Question mark	Optional match (any single character)
*	Astrisk	Any number of occurrence including zero, wild card
+	Plus	One or more of preceding expression (repetitive match)
{N}	Exactly match	Match exactly N times
{N,}	Match at least	Match at least N times
{min,max}	Specified range	Match minimum and maximum times i.e. {3,4}
|	Alternation	Match either of the expression given
-	Dash	Range
(…)	Parenthesis	Used to limit scope of alternation (sub pattern)
\1, \2, \3, …	Backreferences	Matches text previously matched within parenthesis

\b	Word boundary	Matches characters or words marked by the end of the word, i.e. space, period
\B	Backslash	Used for matching \ backslash same as \\
\w	Word character	Used for matching any word character, i.e. letter, number or underscore
\W	Non-word character	Used for matching anything considered not-word i.e. other than letter, number and underscore
\`	Start of buffer	Start of buffer sent to grep
\'	End of buffer	Matches the end of buffer sent to grep

POSIX definition

[:alpha:]		Any alphabetical character
[:digit:]		Any numerical character
[:alnum:]		Any alphabetical or numerical character
[:blank:]		Space or tab character
[:xdigit:]		Hexadecimal character
[:punct:]		Any punctuation symbol
[:print:]		Any printable character (not control characters)
[:space:]		Any white space character
[:graph:]		Excludes whitespace character
[:upper:]		Any uppercase letter
[:lower:]		Any lowercase letter
[:cntrl:]		Control character

Basic regular expression — **grep or grep -G**

-e	-e pattern	Recognizes pattern as regular expression argument i.e. grep -e -style (matches -style)
-f	-f file	Takes patterns from file. The pattern file must list one pattern per line.
-i	-i (ignore case)	Case insensitive search
-v	-v (invert match)	Returns lines that do not match pattern
-w	-w (word boundary match)	Matches exact word with boundary.
-x	-x (line match)	Matches entire line 'Hello, World!'
-c	-c (counts)	Counts the number of matching lines
-l	grep -l "error" *.txt	Prints files containing the pattern, stops at first match
-L	grep -L "error" *.txt	Prints files that do not contain the pattern, stops at first match
-m num	grep -m 10 "error" *.txt	Stops reading file after num lines are matched i.e. only 10 lines that contain regular expression
-o	grep -o pattern filename	Prints only the text that matches
-q	quite	Suppresses output
-s	silent, no messages	Silently discards any error messages resulting from permission errors or non-existent files
-b	byte offset	Displays byte offset of each matching text instead of line number
-H	with filename	Includes the name of the file before each line printed (default when more than one file is input)
-h	no filename	when more than one filename is given it suppresses printing the filename before each output

--label=LABEL	adds label	It will prefix the line with LABLE
-n	line number	Includes the line number of each line displayed.
-T	initial tab	Inserts a tab before each matching line
-u	Unix byte offsets	Computes the byte offset as if it were running under Unix system
-z	null	Prints ASCII NUL (a zero byte) after each filename
-A num	after context = num	Prints num (number of lines) after match
-B num	before context = num	Prints num (number of lines) before match
-C num, -num		Prints num (number of lines before and after match
-R or -r	recursive	Searches files underneath directory submitted as an input file i.e. grep -R pattern path
Extended Regular Expressions	**egrep or grep -E**	
?		Any character preceding ? may or may not appear in the target string.
+		Unlimited number of repetitions while looking for matching string, i.e. grep -E 'regex1+' filename (will look for regex1, regex11, regex111 etc.
{n,m}		Determines how many times a pattern needs to be repeated before matching. i.e. grep -E 'regex{4,6}' filename
|		| is or, allows to combine several patterns into one expression i.e. grep -E 'regex1|regex2' filename
( )		Used to group particular strings of text for various roles i.e. backreferences, alternation, or simply readability
[{]		[ ] Used for matching the character without invoking the special meaning
Fixed strings / Fast grep	**fgrep or grep -F**	
-c	Count	Counts the number of lines contain one or more instances of patter in a file i.e. fgrep -c 'regex' filename
-e		Used for searching more than one pattern or when the pattern begins with hyphen
-f	Outputs results to file	Outputs the results of the search into a file instead of printing it to the terminal
-h		When pattern is searched on more than one file, -h prevents fgrep from displaying filenames before the matched output.
-i	ignores case (capitalization)	-i option ignores capitalization in the pattern when matching it.
-l		Displays the files containing the pattern but not the matching lines.
-n	number of the line	Prints out the line number before the line that matches the pattern.
-v	reverse match	Matches any lines that do not contain the given pattern
Perl Style grep	**grep -P**	**Perl-Compatible Regular Expression (PCRE)**
PCRE-specific escapes		
\a		Matches the alarm character
\cX		Matches ctrl+X, where X is any letter

\e	Matches escape character
\f	Matches from feed character
\n	Matches newline character
\r	Matches carriage return
\t	Matches tab character
\d	Matches any decimal digit
\D	Matches any non-decimal character
\s	Matches any whitespace character
\S	Matches any non-whitespace character
\w	Matches any word character
\W	Matches any non-word character
\b	Matches when at word boundary
\B	Matches when at not a word boundary
\A	Matches when at start of subject
\Z	Matches when at end of subject or before newline
\z	Matches when at end of subject
\G	Matches at first matching position

SPACE FOR NOTES:

For more information please visit: **http://bioinformatictools.blogspot.in/**