

# Table of Contents



```
#!/usr/bin/perl -w

###! /usr/local/bin/perl -w
# match_or :    Time-stamp: <2014-07-17 13:47:48>

# code for the or (-o) option for match
# match -o str1 str2  matches lines that have str1 OR str2
# NYI: Either converting the "match" shell script to perl or converting
#      this to an inline that can be used in the shell script would be good.
#
# Note: (06/10/16) This can be done with
#      grep 'str1|str2'

my @patterns;
my $case_insensitive_flag = 0;
my $verbose_flag = 0;

sub msg {
    my $message = shift;
    $message .= "\n" unless $message =~ /\n$/; # ensure trailing "\n"
    print($message);
}

sub usage {
    print "match_or
    code for the or (-o) option for match
    usage: cat file | match_or str1 str2
    usage: match_or str1 str2 <infile
           match_or str1 str2  matches lines that have str1 OR str2
           match_or -i str1 str2    case insensitive match
    eg: cal 11 2004 | match_or 10 17  gives
           7  8  9 10 11 12 13
           14 15 16 17 18 19 20
           match_or -v str1 str2    verbose flag displays the search pattern
    eg: cal 11 2004 | match_or -v 10 17  gives
           10|17
           7  8  9 10 11 12 13
           14 15 16 17 18 19 20
";
}

if ($#ARGV == -1) {
    usage();
    exit;
}
```

```
OPT:
while ($#ARGV >= 0) {
  $_ = $ARGV[0];
  F00: {
    (/^-help$/ || /^-h$/) && (&usage, exit, last F00);
    /^-i$/ && ($case_insensitive_flag = 1, last F00);
    /^-v$/ && ($verbose_flag = 1, last F00);
    # The default /^-o$/ && (shift(@ARGV), @patterns = @ARGV, last F00);
    /^-/ && (print("$myname: \"$_\" is an unknown option.\n"),
             print("enter \"$myname -h\" for options\n"),
             exit, last OPT);
    last OPT;
  }
  shift(@ARGV);
}

# start
@patterns = @ARGV;    # the default
my $j;
foreach my $f (@patterns) {
  if ($j) {
    $j .= '|' . $f;
  } else {
    $j = $f;    # first time
  }
}

if ($verbose_flag) {
  msg $j;
}

my @lines = <STDIN>;
my @res;
if ($case_insensitive_flag) {
  @res = grep (/ $j/i, @lines);
} else {
  @res = grep (/ $j/, @lines);
}

foreach my $f (@res) {
  msg "$f";
}

### old version
### Note: using grep might be more efficient (do we care?)
## @patterns = @ARGV;    # the default
## my @lines = <STDIN>;
## foreach my $f (@lines) {
```

```
## PAT:
##   foreach my $j (@patterns) {
##   if ($f =~ /$j/) {
##       msg("$f");
##       last PAT;
##   }
##   }
## }

# eg: If infile is
# project "MyAdmin"
# 04-Feb-03.14:12:16 by ClearCase admin account (vobadmin.ccusers@titan)
# "This is a MyAdministrator project"
# owner: vobadmin
# group: ccusers
# folder: RootFolder@/vobs/mn_t
# integration stream: MyAdmin_integration@/vobs/mn_t
# modifiable components:
# default rebase promotion level: SANE

# Then "cat infile |match_or -o owner: group: stream:" prints:
# owner: vobadmin
# group: ccusers
# integration stream: MyAdmin_integration@/vobs/mn_t

__END__
```

From:

<https://wiki.linux-ottawa.org/> - **Linux-Ottawa (OCLUG) Wiki**

Permanent link:

[https://wiki.linux-ottawa.org/doku.php?id=wiki:match\\_or](https://wiki.linux-ottawa.org/doku.php?id=wiki:match_or)

Last update: **2017/12/16 15:23**

