

Re: converting to FLOATING_POINT..

Source: <http://coding.derkeiler.com/Archive/Perl/perl.beginners/2005-08/msg00426.html>

- *From:* charley@xxxxxxxxxxxxx (Chris Charley)
 - *Date:* Wed, 17 Aug 2005 23:38:11 -0400
-

Chris Charley wrote

The solution above **assumes** that you would know beforehand the widths you want for each column (determined by the largest number in each column you want to format).

The solution below allows you to determine the greatest width required programmatically using the `max()` function from the `List::Util` module. Then, the max widths for each column are used in the `sprintf` function, (which uses a '*' as a placeholder, if thats an accurate term for the asterisk :-). This would provide a better solution (in place of 'hard coding' the widths in `sprintf`).

Well, after a short motorcycle ride and some thinking, I realize I was too hasty and misstated three lines critical to a correct solution :-(

Sorry. The changes are given below with an explanation. Felt it was necessary to 'set the record straight'.

Chris

```
#!/usr/bin/perl
use strict;
use warnings;
use List::Util qw/ max /;

my $mdout_file = "mdout.txt";

my $mdout_xtemp_file = "temp.txt";
```

Re: converting to FLOATING_POINT..

```
open IN, $mdout_file or die;

my @time;
my @temp;
while (<IN>){

    if (/TEMP/) {

        my $time = (substr($_, 30, 14));
        $time =~ s/\s//g;
        my $temp = (substr($_, 53, 10));
        $temp =~ s/\s//g;

        push @time, $time*2; # incorrect
        push @temp, $temp; # incorrect

    }

    push @time, sprintf("%.1f", $time*2);
    push @temp, sprintf("%.2f", $temp);
}
```

The format should be applied at this point so that that when the `max_length` is calculated below, the width of each `$time` and `$temp` variable will be as it will be when printed out.

```
    }
}
close IN or die $!;

my $maxlen_time = max map{ length } @time;
my $maxlen_temp = max map{ length } @temp;

open OUT, ">$mdout_xtemp_file" or die;

for my $i (0..$#time) {
    print OUT sprintf("%.1f %.2f\n",
```

Re: converting to FLOATING_POINT..

should be: print OUT sprintf("%*s %*s\n",

Here, no need to reformat as a float the 2 values. That has been done above.

```
                $maxlen_time, $time[$i], $maxlen_temp, $temp[$i]);  
    }  
    close OUT or die $!;
```