

Scripting Languages - Assignments for the 2nd lab

Contenido

Introduction.....	2
Assignment 1	2
Assignment 2	3
Assignment 3	4
Assignment 4	5
Assignment 5	6
Assignment 6	7

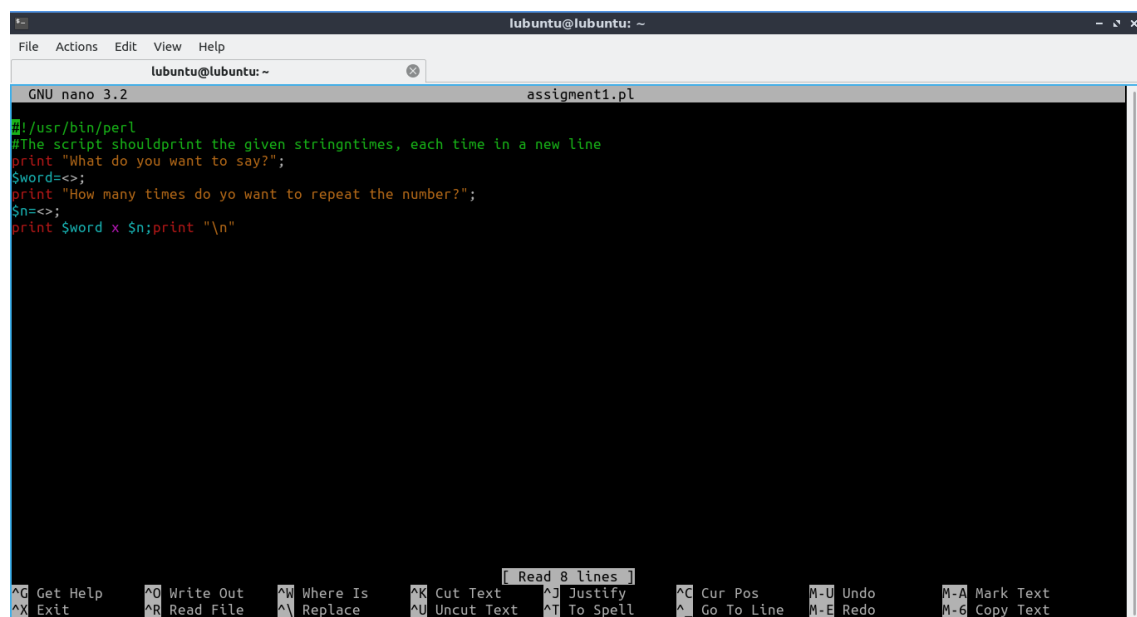
Introduction

I would like to start by commenting on how I have developed this practice. I have used a lubuntu image on a pendrive, to be able to carry out the exercises in a more optimal way. This operative system had inside perl language.

Next, I will include a series of screenshots to demonstrate the realization of these. I've comment the code of the program, but in addition, I will comment here for concrete some important parts of my program codes.

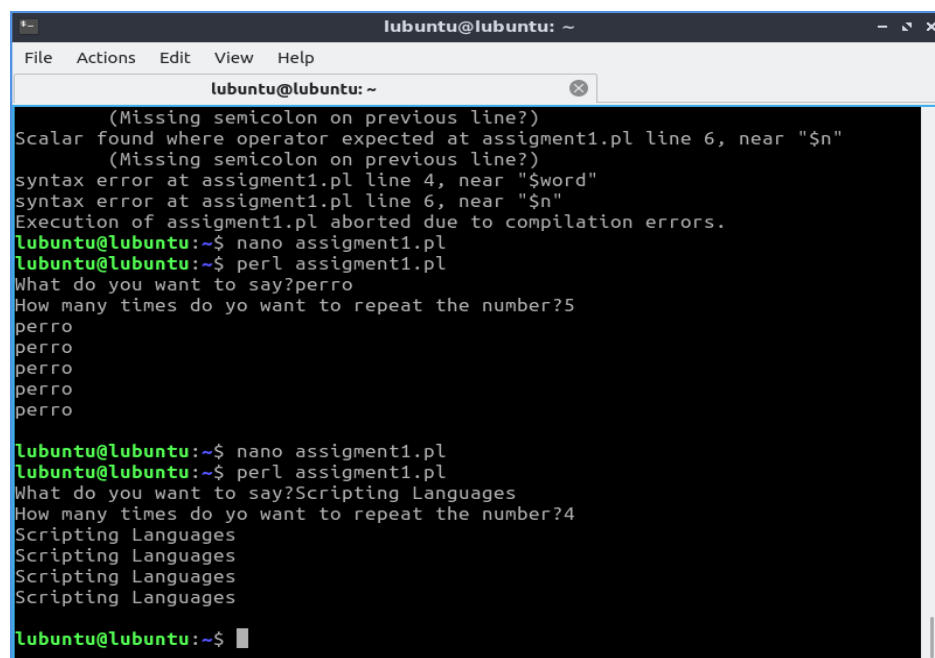
Assignment 1

This is the code:



```
#!/usr/bin/perl
#The script should print the given string times, each time in a new line
print "What do you want to say?";
$word=<>;
print "How many times do you want to repeat the number?";
$n=<>;
print $word x $n;print "\n"
```

This is the assignment 1 result:



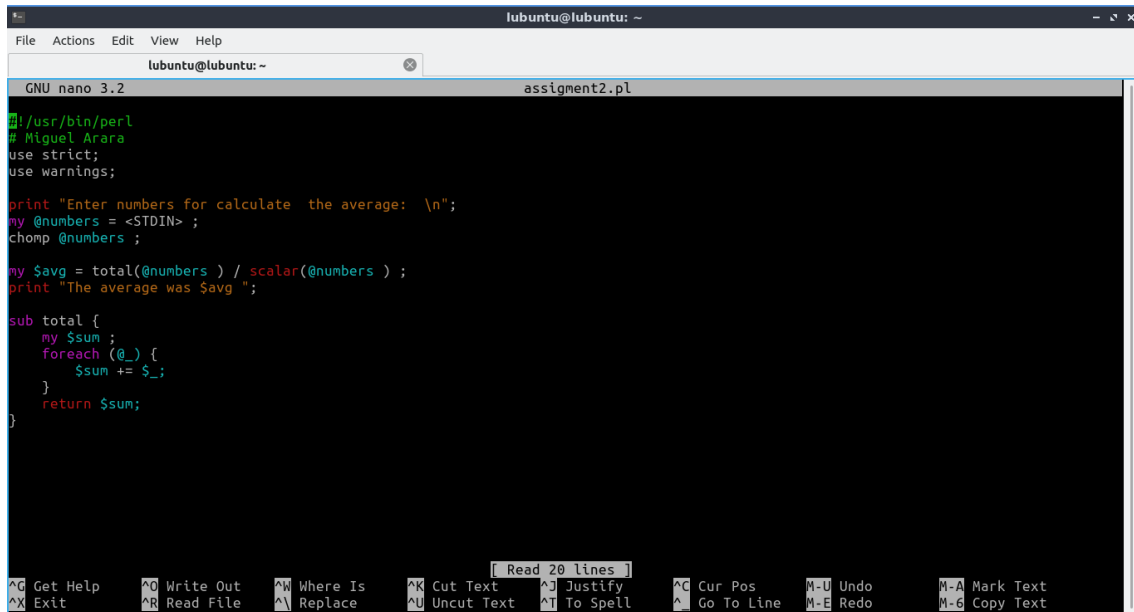
```
(Missing semicolon on previous line?)
Scalar found where operator expected at assignment1.pl line 6, near "$n"
(Missing semicolon on previous line?)
syntax error at assignment1.pl line 4, near "$word"
syntax error at assignment1.pl line 6, near "$n"
Execution of assignment1.pl aborted due to compilation errors.
lubuntu@lubuntu:~$ nano assignment1.pl
lubuntu@lubuntu:~$ perl assignment1.pl
What do you want to say?perro
How many times do you want to repeat the number?5
perro
perro
perro
perro
perro

lubuntu@lubuntu:~$ nano assignment1.pl
lubuntu@lubuntu:~$ perl assignment1.pl
What do you want to say?Scripting Languages
How many times do you want to repeat the number?4
Scripting Languages
Scripting Languages
Scripting Languages
Scripting Languages

lubuntu@lubuntu:~$
```

Assignment 2

This is the code:



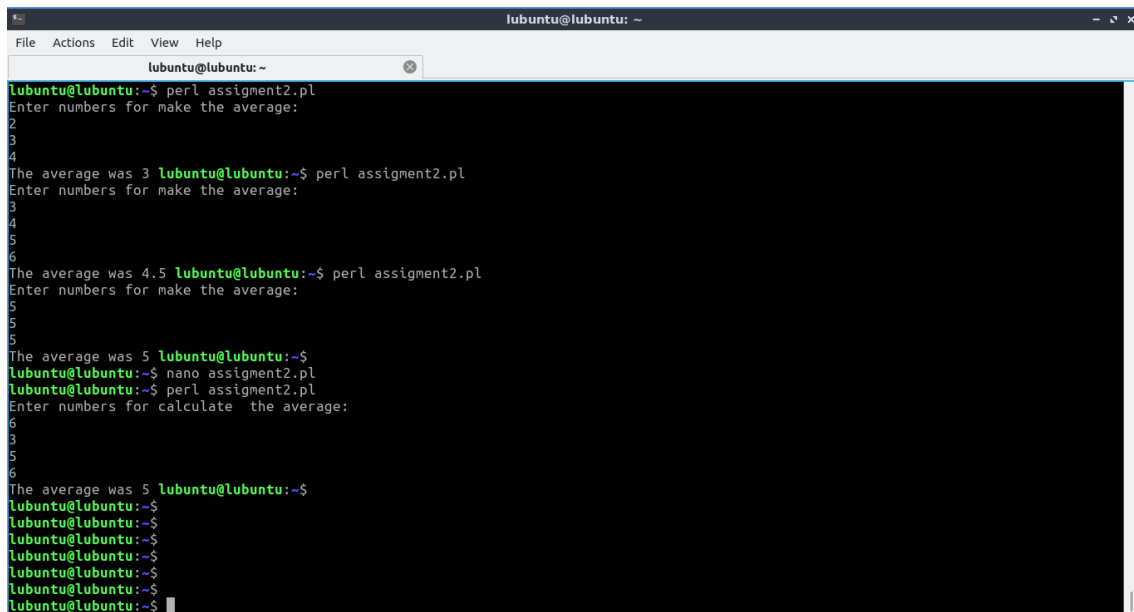
```
#!/usr/bin/perl
# Miguel Arara
use strict;
use warnings;

print "Enter numbers for calculate the average: \n";
my @numbers = <STDIN> ;
chomp @numbers ;

my $avg = total(@numbers ) / scalar(@numbers ) ;
print "The average was $avg ";

sub total {
    my $sum ;
    foreach (@_) {
        $sum += $_;
    }
    return $sum;
}
```

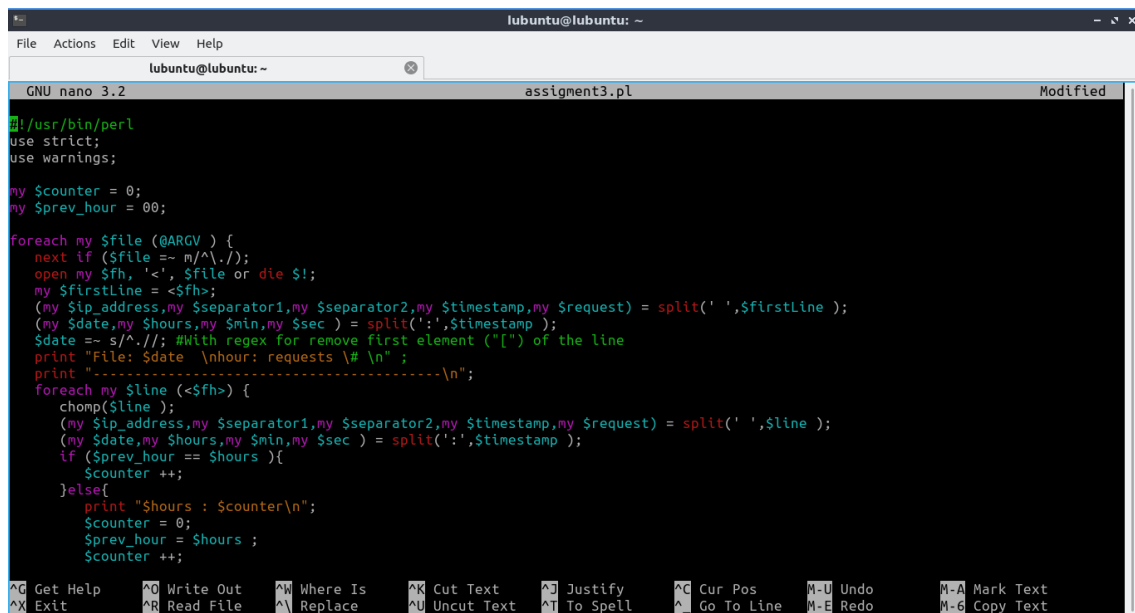
This is the assignment 2 result:



```
lubuntu@lubuntu:~$ perl assignment2.pl
Enter numbers for make the average:
2
3
4
The average was 3
lubuntu@lubuntu:~$ perl assignment2.pl
Enter numbers for make the average:
3
4
5
6
The average was 4.5
lubuntu@lubuntu:~$ perl assignment2.pl
Enter numbers for make the average:
5
5
5
The average was 5
lubuntu@lubuntu:~$ nano assignment2.pl
lubuntu@lubuntu:~$ perl assignment2.pl
Enter numbers for calculate the average:
6
3
5
6
The average was 5
lubuntu@lubuntu:~$
lubuntu@lubuntu:~$
lubuntu@lubuntu:~$
lubuntu@lubuntu:~$
lubuntu@lubuntu:~$
lubuntu@lubuntu:~$
lubuntu@lubuntu:~$
```

Assignment 3

This is the code (In the image doesn't appear the last 3 lines, but in the program .pl yes):

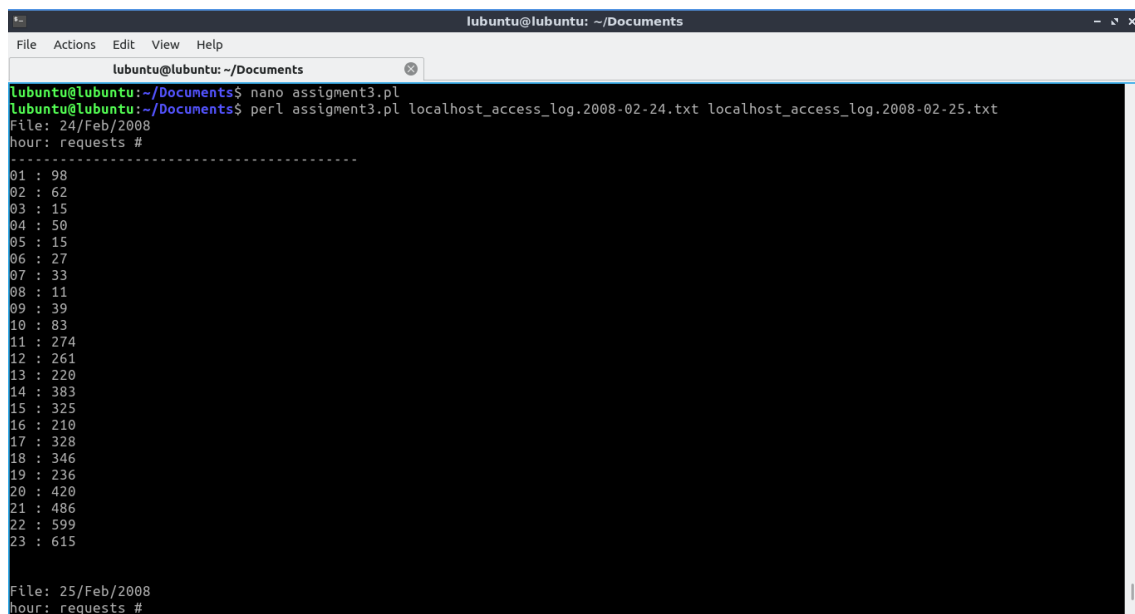


```
#!/usr/bin/perl
use strict;
use warnings;

my $counter = 0;
my $prev_hour = 00;

foreach my $file (@ARGV) {
    next if ($file =~ m/^\./);
    open my $fh, '<', $file or die $!;
    my $firstline = <$fh>;
    (my $ip_address, my $separator1, my $separator2, my $timestamp, my $request) = split(' ', $firstline);
    (my $date, my $hours, my $min, my $sec) = split(':', $timestamp);
    $date =~ s/^\./; #With regex for remove first element ("") of the line
    print "File: $date \nhour: requests \# \n";
    print "-----\n";
    foreach my $line (<$fh>) {
        chomp($line);
        (my $ip_address, my $separator1, my $separator2, my $timestamp, my $request) = split(' ', $line);
        (my $date, my $hours, my $min, my $sec) = split(':', $timestamp);
        if ($prev_hour == $hours) {
            $counter++;
        } else {
            print "$hours : $counter\n";
            $counter = 0;
            $prev_hour = $hours;
            $counter++;
        }
    }
}
```

This is the assignment 3 result:



```
lubuntu@lubuntu: ~/Documents
lubuntu@lubuntu:~/Documents$ nano assignment3.pl
lubuntu@lubuntu:~/Documents$ perl assignment3.pl localhost_access_log.2008-02-24.txt localhost_access_log.2008-02-25.txt
File: 24/Feb/2008
hour: requests #
-----
01 : 98
02 : 62
03 : 15
04 : 50
05 : 15
06 : 27
07 : 33
08 : 11
09 : 39
10 : 83
11 : 274
12 : 261
13 : 220
14 : 383
15 : 325
16 : 210
17 : 328
18 : 346
19 : 236
20 : 420
21 : 486
22 : 599
23 : 615

File: 25/Feb/2008
hour: requests #
```

```

lubuntu@lubuntu: ~/Documents
File Actions Edit View Help
lubuntu@lubuntu: ~/Documents
23 : 615

File: 25/Feb/2008
hour: requests #
-----
00 : 593
01 : 377
02 : 68
03 : 48
05 : 5
06 : 26
07 : 28
08 : 41
09 : 131
10 : 316
11 : 549
12 : 469
13 : 519
14 : 478
15 : 552
16 : 297
17 : 337
18 : 296
19 : 232
20 : 239
21 : 200
22 : 210
23 : 238

lubuntu@lubuntu:~/Documents$

```

******from here I have made the decision not to put the code of the entire program of assignments. It was too large. The code is commented...***

Assignment 4

For start, these are the code lines if user doesn't put any file in command line argument:

```

my $file = $ARGV[0];
unless ($file) {
    print "Enter filename:\n";
    $file = <STDIN>;
    chomp $file;
}

```

Probably, I could have divided this program, but I made it in a simple program

This is my table lab_submmited.csv (attached it in the .zip) :

JMBAG	Surname	Name	Laboratory	Submitted
36438919	Bagaric	Magdalena	2011-03-14 08:00 11:00 A209	14/03/2011 8:45
36433049	Bajac	Darko	2011-03-14 08:00 11:00 A209	14/03/2011 8:48
36436684	Lombarovic	Mladen	2011-03-14 11:00 14:00 A210	14/03/2011 12:08
36325839	Matosic	Luka	2011-03-14 11:00 14:00 A210	15/03/2011 11:49

This is the result:

```

lubuntu@lubuntu:~$ nano assignment4.pl
lubuntu@lubuntu:~$ perl assignment4.pl lab_submitted.csv
syntax error at assignment4.pl line 45, near ")" or"
syntax error at assignment4.pl line 49, near "}"
Execution of assignment4.pl aborted due to compilation errors.
lubuntu@lubuntu:~$ nano assignment4.pl
lubuntu@lubuntu:~$ perl assignment4.pl lab_submitted.csv
36436684 Lombarovic Mladen - PROBLEM 2011-03-14 11:00 -->2011-03-14 12:08:26
36325839 Matosic Luka - PROBLEM 2011-03-14 11:00 -->2011-03-15 11:49:26
lubuntu@lubuntu:~$ nano assignment4.pl
lubuntu@lubuntu:~$ nano assignment4.pl
lubuntu@lubuntu:~$ perl assignment4.pl lab_submitted.csv
36436684 Lombarovic Mladen - PROBLEM 2011-03-14 11:00 -->2011-03-14 12:08:26
36325839 Matosic Luka - PROBLEM 2011-03-14 11:00 -->2011-03-15 11:49:26
lubuntu@lubuntu:~$

```

Assignment 5

This is my table results.csv. I know that probably, is not the best organization for a table of Excel, but it was easier for processing the document in Perl:

component weights	0.15	0.2	0.3	0.1	0.1	0.1	0.05			
Student Result	36438919	Bagaric	Magdalena	91.5	90.75	88.25	100	87.5	87.5	90
	36439578	Bajer	Goran	64.4	72	64.75	72.5	76.25	50	62
	36436684	Lombarovic	Mladen	14.7	75.7	-	-	5	20	-
	36436690	Karlovac	Tomislav	67.8	90.4	-	80.5	73.4	33.3	50

Check the next image. When I don't put the file in line command argument , the program says me that enter a filename. Then with two subroutine, one for extract the components weights, and the other for make the ranking.

```

lubuntu@lubuntu:~$ perl assignment5.pl
Ranking List
-----
Enter filename:
results.csv
Enter filename:
^C
lubuntu@lubuntu:~$ nano assignment5.pl
lubuntu@lubuntu:~$ perl assignment5.pl
Enter filename:
results.csv
Ranking List
-----
Enter filename:
^C
lubuntu@lubuntu:~$ nano assignment5.pl
lubuntu@lubuntu:~$ perl assignment5.pl
Enter filename:
results.csv
Ranking List
-----
1. Magdalena Bagaric (36438919) 90.35
2. Goran Bajer (36439578) 66.46
3. Mladen Lombarovic (36436684) 19.845
4. Tomislav Karlovac (36436690) 49.47
lubuntu@lubuntu:~$

```

Assignment 6

This is a capture of my .txt file:

```
*lindsay_story.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda

One night Lindsay was lounging around at home by herself, watching television,
when she received a text message on her phone: "Linzy i c u." The message was from a number she wasn't familiar
with. She knew since she was sitting in bed upstairs with the shades drawn,
that no one could see her. But she instinctively looked around anyway, and even peeked out the window.
Nobody was around.

Lindsay knew it wasn't a friend texting her. None of her friends texted like that, and they all knew how to spell.
None of them would have misspelled the word "angel." So Lindsay text back,
"You're no friend of mine if you can't spell my name right and you can't even spell the word ANGEL correctly."
The response was, "doesnt matter. r u havin fun watchin ur tv?" This gave her chills. Who could be watching her?
There was nobody around unless this person was in the house. So Lindsay ran around checking the doors and windows and the other rooms
, even the garage. She was definitely the only one home.
```

This is the result of assignment 6, passing the number that I want for a prefix length, 4 and the document:

```
lubuntu@lubuntu: ~
File Actions Edit View Help
lubuntu@lubuntu: ~
lubuntu@lubuntu:~$ perl assignment6.pl 4 lindsay_story.txt
ANGE : 1
But : 1
Lind : 4
Linz : 1
Nobo : 1
None : 2
One : 1
She : 2
So : 2
The : 2
Ther : 1
This : 1
Who : 1
Your : 1
a : 3
all : 1
and : 5
ange : 1
anyw : 1
arou : 5
at : 1
back : 1
be : 1
bed : 1
by : 1
c : 1
cant : 2
chec : 1
chil : 1
corr : 1
coul : 2
```

This is the result passing number 5 as prefix length and then I don't put any file in the line command argument and then, the program says me enter the filename:

```
lubuntu@lubuntu: ~  
File Actions Edit View Help  
lubuntu@lubuntu: ~  
lubuntu@lubuntu:~$ perl assignment6.pl 5  
Enter filename:  
lindsay_story.txt  
ANGEL : 1  
But : 1  
Linds : 4  
Linzy : 1  
Nobod : 1  
None : 2  
One : 1  
She : 2  
So : 2  
The : 2  
There : 1  
This : 1  
Who : 1  
Youre : 1  
a : 3  
all : 1  
and : 5  
angel : 1  
anywa : 1  
aroun : 5  
at : 1  
back : 1  
be : 1  
bed : 1  
by : 1  
c : 1  
cant : 2  
check : 1  
chill : 1
```