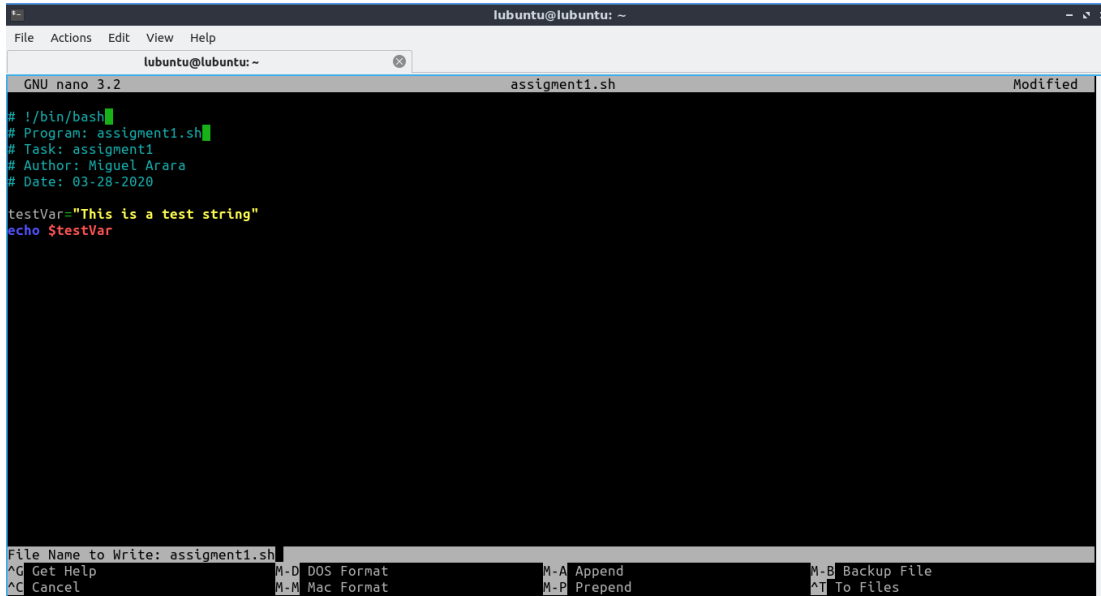


## Scripting Languages / Assignments for the 1<sup>st</sup> lab

### Assignment 1

- Created the assignment1.sh



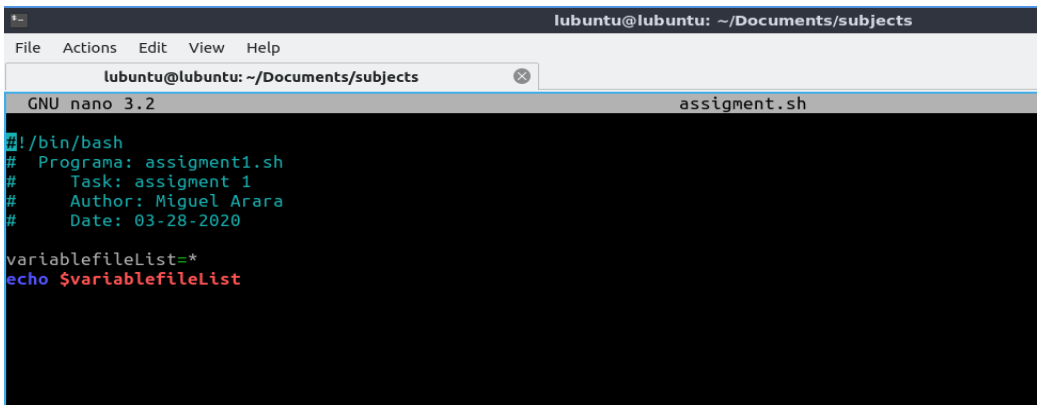
```
#!/bin/bash
# Program: assignment1.sh
# Task: assignment1
# Author: Miguel Arara
# Date: 03-28-2020

testVar="This is a test string"
echo $testVar
```

- Display the value of the variable testVar.

```
lubuntu@lubuntu:~/Documents$ bash assignment1.sh
This is a test string
lubuntu@lubuntu:~/Documents$
```

- Store the list of all filenames contained in the current directory into the variable fileList. Use the filename expansion. Display the result



```
#!/bin/bash
# Programa: assignment1.sh
# Task: assignment 1
# Author: Miguel Arara
# Date: 03-28-2020

variablefileList=*
echo $variablefileList
```

Executed the script, print all files in the current director.

```
lubuntu@lubuntu:~/Documents/subjects$ nano
lubuntu@lubuntu:~/Documents/subjects$ nano assignment.sh
lubuntu@lubuntu:~/Documents/subjects$ pwd
/home/lubuntu/Documents/subjects
lubuntu@lubuntu:~/Documents/subjects$ ls
Artificial_Intelligence  Scripting_Languages  assignment.sh  assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$ bash assignment.sh
Artificial_Intelligence Scripting_Languages assignment.sh assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$
```

- Concatenate the sentence stored in the variable `testVar` three times. Each sentence should end by an added full stop and a blank. Assign the result to the variable `testConcat`.

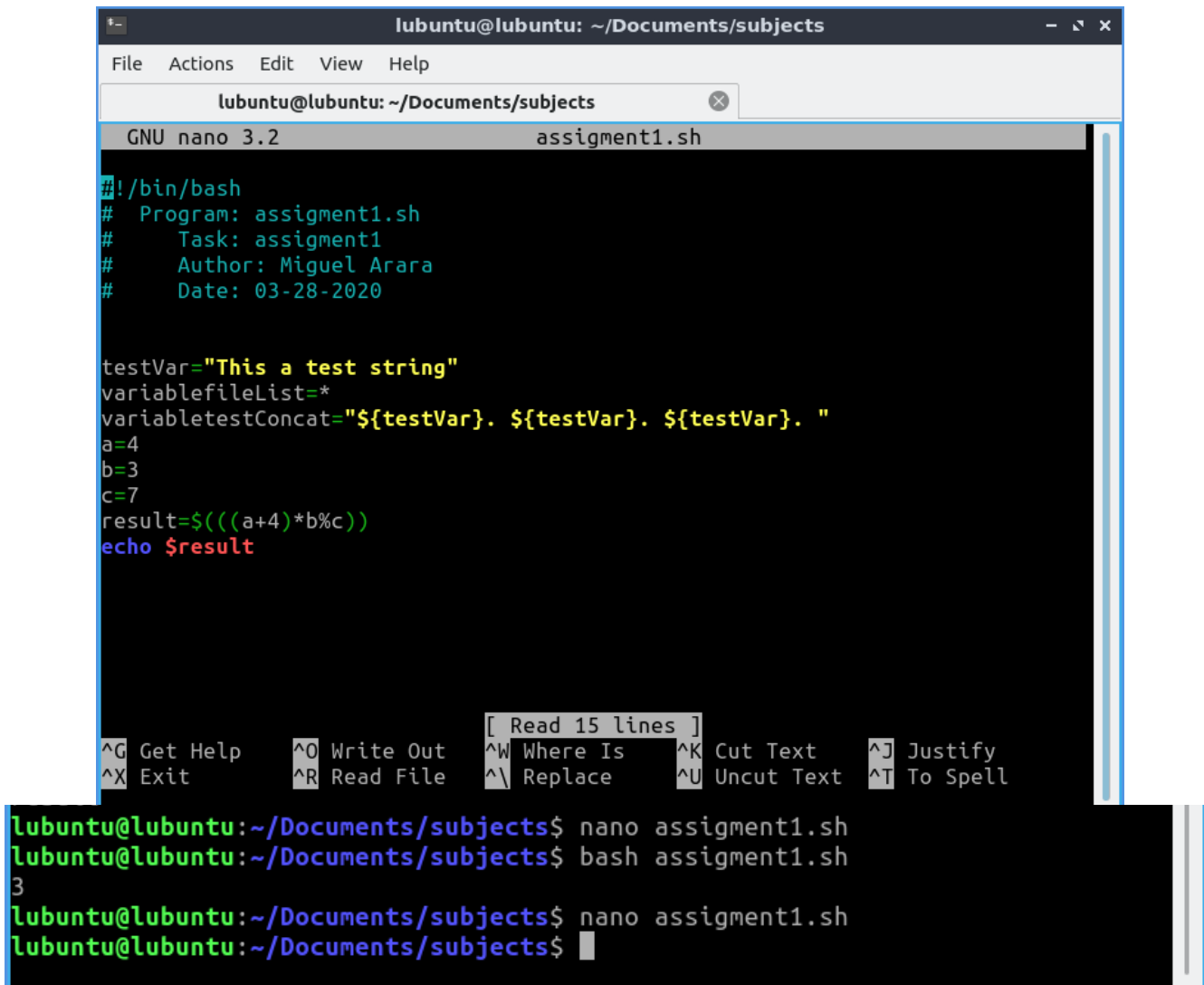
```
GNU nano 3.2 assignment1.sh
#!/bin/bash
# Program: assignment1.sh
# Task: assignment1
# Author: Miguel Arara
# Date: 03-28-2020

testVar="This a test string"
variablefileList=*
variabletestConcat="${testVar}. ${testVar}. ${testVar}. "
echo "$variabletestConcat"
```

The result of `variabletestConcat`>

```
lubuntu@lubuntu:~/Documents/subjects$ bash assignment1.sh
This a test string. This a test string. This a test string.
lubuntu@lubuntu:~/Documents/subjects$
```

- Set the variable `a` to value 4, variable `b` to 3, variable `c` to 7. Set variable `d` to the value obtained by evaluating the expression `(a+ 4)*b%c`. Use the arithmetic expansion.



```
lubuntu@lubuntu: ~/Documents/subjects
File Actions Edit View Help
lubuntu@lubuntu: ~/Documents/subjects
GNU nano 3.2 assignment1.sh
#!/bin/bash
# Program: assignment1.sh
# Task: assignment1
# Author: Miguel Arara
# Date: 03-28-2020

testVar="This a test string"
variablefileList=*
variabletestConcat="${testVar}. ${testVar}. ${testVar}. "
a=4
b=3
c=7
result=$((a+4)*b*c))
echo $result

[ Read 15 lines ]
^G Get Help      ^O Write Out     ^W Where Is     ^K Cut Text     ^J Justify
^X Exit          ^R Read File     ^\ Replace      ^U Uncut Text   ^T To Spell

lubuntu@lubuntu:~/Documents/subjects$ nano assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$ bash assignment1.sh
3
lubuntu@lubuntu:~/Documents/subjects$ nano assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$
```

The result is 3.

- Set the variable `number_of_words` to the number of words in all.txt files in the current directory. Use the command substitution and the `wc` command.

I have two .txt files in the current directory with 4 and 2 words in each one. In total 6.

```
lubuntu@lubuntu: ~/Documents/subjects
File Actions Edit View Help
lubuntu@lubuntu: ~/Documents/subjects
lubuntu@lubuntu:~/Documents/subjects$ nano assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$ bash assignment1.sh
This a test string. This a test string. This a test string.
lubuntu@lubuntu:~/Documents/subjects$ nano assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$ bash assignment1.sh
This a test string. This a test string. This a test string.
lubuntu@lubuntu:~/Documents/subjects$ nano assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$ bash assignment1.sh
result
lubuntu@lubuntu:~/Documents/subjects$ nano assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$ bash assignment1.sh
3
lubuntu@lubuntu:~/Documents/subjects$ nano assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$ wc assignment1.sh
 15  29 266 assignment1.sh
lubuntu@lubuntu:~/Documents/subjects$ nano texto1.txt
lubuntu@lubuntu:~/Documents/subjects$ wc texto1.txt
  4  3 21 texto1.txt
lubuntu@lubuntu:~/Documents/subjects$ nano texto2.txt
lubuntu@lubuntu:~/Documents/subjects$ wc texto2.txt
  2  2 15 texto2.txt
lubuntu@lubuntu:~/Documents/subjects$ wc *.txt
  4  3 21 texto1.txt
  2  2 15 texto2.txt
  6  5 36 total
lubuntu@lubuntu:~/Documents/subjects$
```

- Print the content of your home directory by using the tilde expansion

```
lubuntu@lubuntu:~/Documents/subjects$ ~
bash: /home/lubuntu: Is a directory
lubuntu@lubuntu:~/Documents/subjects$ cd ~
lubuntu@lubuntu:~$ ~
bash: /home/lubuntu: Is a directory
lubuntu@lubuntu:~$
```

- Use a cut command to display (only) the information about user name, home directory and login shell of each user. The information is to be read from the/etc/passwd file.

## Assignment 2

Command grep, regular expressions, command find. Loops.

- Write a grep command to find all lines in the file food.txt containing names of fruits(banana, apple, strawberry, grape, watermelon). The search should be case-insensitive.
  - I have created fruits array, and I have used in grep.

```
lubuntu@lubuntu: ~/Documents/subjects/Scripting
File Actions Edit View Help
lubuntu@lubuntu: ~/Documents/subjects/Scripting
lubuntu@lubuntu:~/Documents/subjects/Scripting$ $fruits
banana,apple,strawberry,grape,watermelon: command not found
lubuntu@lubuntu:~/Documents/subjects/Scripting$ echo $fruits
banana,apple,strawberry,grape,watermelon
lubuntu@lubuntu:~/Documents/subjects/Scripting$ grep -i $fruits food.txt
ham,cheesse,coca cola,jam,croissant,bread,banana,apple,strawberry,grape,watermelon
lubuntu@lubuntu:~/Documents/subjects/Scripting$
lubuntu@lubuntu:~/Documents/subjects/Scripting$
lubuntu@lubuntu:~/Documents/subjects/Scripting$
lubuntu@lubuntu:~/Documents/subjects/Scripting$
lubuntu@lubuntu:~/Documents/subjects/Scripting$
```

- Modify the command so that it prints only the lines not containing any of the listed words. Redirect the output to a file named non-fruit.txt.
  - With grep -v

```
lubuntu@lubuntu:~/Documents$ grep -v $fruits food.txt
ham,cheesse,coca cola,jam,croissant,bread
lubuntu@lubuntu:~/Documents$
```

- Write a grep command to find all files in the directory~/projects/and all subdirectories containing a code consisting of three capital letters and six digits (e.g. ABC123456) and display ll corresponding lines.

```
lubuntu@lubuntu:/ $ cd etc/
lubuntu@lubuntu:/etc$ sudo grep -r CBD674803
lubuntu@lubuntu:/etc$
```

- Write a command which prints the names and details of all files in the current directory which were last modified before 7 to 14 days.

```
lubuntu@lubuntu:/etc$ find . -mtime -7 | find . -mtime -14
./
./pwd.lock
./NetworkManager/system-connections
./NetworkManager/system-connections/Erasmrooms.nmconnection
./apt
./apt/apt.conf.d
./apt/apt.conf.d/50unattended-upgrades
./apt/apt.conf.d/00aptitude
./apt/sources.list
./casper.conf
find: './cups/ssl': Permission denied
./cups
find: './cups/ssl': Permission denied
./cups/subscriptions.conf
./default/locale
./fstab
./group
```

- Write a one-line for loop which prints integer numbers between 1 and 20. Use the sequence generating expression or seq command.

```
lubuntu@lubuntu: ~
File Actions Edit View Help
lubuntu@lubuntu: ~
GNU nano 3.2 loop.sh
#!/bin/bash
for i in $(seq 20)
do
    printf "%i "
done
```

This is the result.

```
lubuntu@lubuntu: ~
File Actions Edit View Help
lubuntu@lubuntu: ~
lubuntu@lubuntu:~$ nano loop.sh
lubuntu@lubuntu:~$ bash loop.sh
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 lubuntu@lubuntu:~$
lubuntu@lubuntu:~$
```

## Assignment 3

I download both files .txt in that directory:

```

lubuntu@lubuntu: ~/Documents
File Actions Edit View Help
lubuntu@lubuntu: ~/Documents
lubuntu@lubuntu:~$
lubuntu@lubuntu:~$
lubuntu@lubuntu:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos loop.sh nobleNote
lubuntu@lubuntu:~$ cd Documents/
lubuntu@lubuntu:~/Documents$ ls
assignmentLab1-MiguelArara.odt food.txt localhost_access_log.2008-02-24.txt localhost_access_log.2008-02-25.txt
lubuntu@lubuntu:~/Documents$

```

- Write a bash script that will analyze all log files created in February (according to the filename) and print the statistics of the requested methods. For each file the script should print the date and then for each action (the requested method and web page) print the number of occurrences in that file. The output list should be sorted according to the number of action occurrences, in descending order.

```

lubuntu@lubuntu: ~/Documents
File Actions Edit View Help
lubuntu@lubuntu: ~/Documents
GNU nano 3.2 assignment3.sh Modified
#!/bin/bash
# Program: assignment3.sh
# Author: Miguel Arara
# Fecha: 03/29/2020

echo "Date: 25-02-2008"
echo "-----"

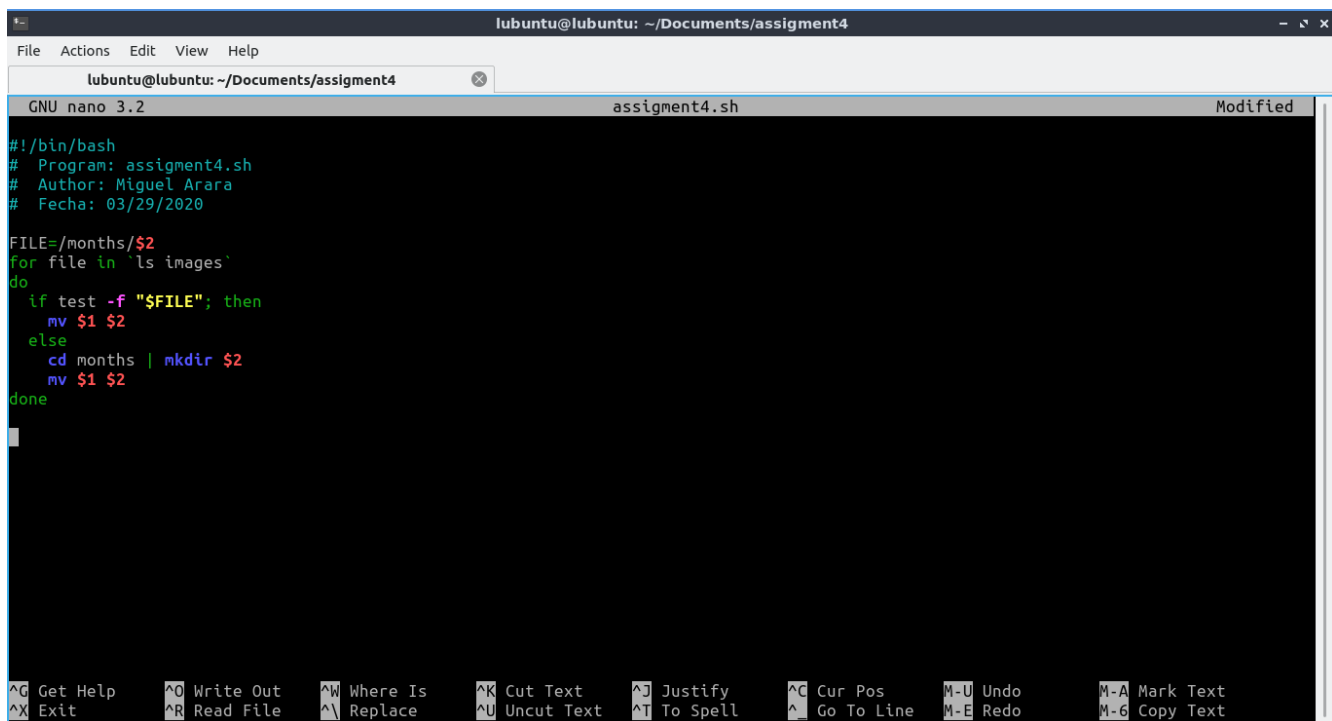
countGET=1
countPOST=1
oldIFS=$IFS
IFS=$'\n'
for line in $(cat localhost_access_log.2008-02-25.txt)
do
    if [grep GET localhost_access_log.2008-02-25.txt]
    then
        echo "$countGET: $line"
        ((countGET++))
    else
        if [grep POST localhost_access_log.2008-02-25.txt]
        then
            echo "$countPOST: $line"
            ((countPOST++))
        fi
    fi
done
IFS=$old_IFS

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^G Cur Pos     M-U Undo       M-A Mark Text
^X Exit          ^R Read File    ^\ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line   M-E Redo       M-G Copy Text

```

## Assignment 4

- Suppose that one subdirectory inside the current directory contains photographs from our camera. The time of modification corresponds to the time of taking the photo. Write a script to move the photographs into folders corresponding to the months (the folder name should be in the format yyyy-mm). The script should accept two command line arguments: name of the source directory containing the images, and the name of the destination directory (where month directories are to be created). The script should check if the needed directory exists, create it, and move each image into the corresponding directory.
  - For example \$1=images \$2=2009-03  
    > assignment4.sh images 2009-03



```
#!/bin/bash
# Program: assignment4.sh
# Author: Miguel Arara
# Fecha: 03/29/2020

FILE=/months/$2
for file in `ls images`
do
    if test -f "$FILE"; then
        mv $1 $2
    else
        cd months | mkdir $2
        mv $1 $2
    done
```



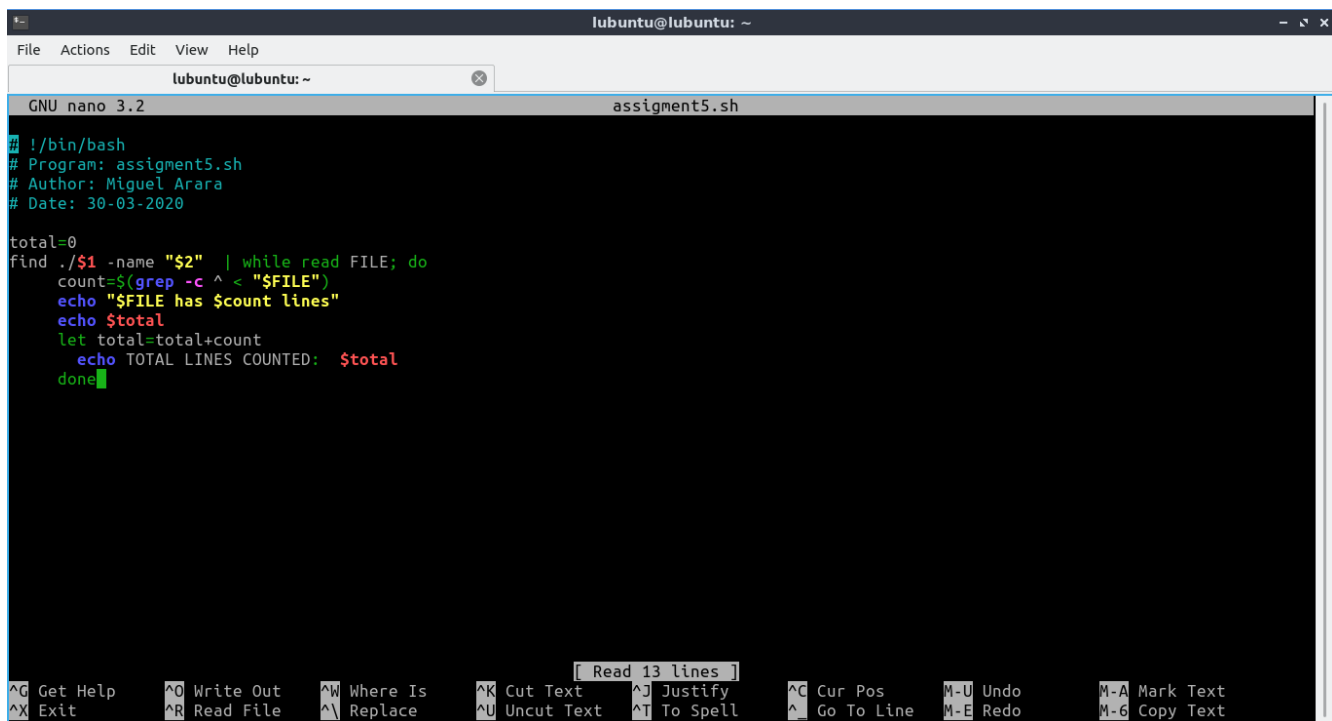
## Assignment 5

- Write a script which will count the lines in all files in the directory given as the first command line argument and its subdirectories. The second command line argument should specify the filename pattern (e.g. '\*.c'). The script should print the given command line arguments and then the total number of lines
  - The script should accept two command line arguments: name of the source directory containing the images, and the name of the destination director

This program counts all lines of all files.

If we want put both line arguments only I have to do a couple of changes:

- Use \$1 and \$2 for use the line arguments
- Simply, I put the parameters:



```
lubuntu@lubuntu: ~  
File Actions Edit View Help  
lubuntu@lubuntu: ~  
GNU nano 3.2 assignment5.sh  
#!/bin/bash  
# Program: assignment5.sh  
# Author: Miguel Arara  
# Date: 30-03-2020  
  
total=0  
find ./ $1 -name "$2" | while read FILE; do  
    count=$(grep -c ^ < "$FILE")  
    echo "$FILE has $count lines"  
    echo $total  
    let total=total+count  
    echo TOTAL LINES COUNTED: $total  
done
```

OUTPUT: (\$1=Documents, \$2=assignment3.sh) → Result: 28 lines

```
TOTAL LINES COUNTED: 73868  
./Documents/food.txt has 3 lines  
73868  
TOTAL LINES COUNTED: 73871  
lubuntu@lubuntu:~$ nano assignment5.sh  
lubuntu@lubuntu:~$ bash assignment5.sh Documents assignment3.sh  
./Documents/assignment3.sh has 28 lines  
0  
TOTAL LINES COUNTED: 28  
lubuntu@lubuntu:~$ nano assignment5.sh  
lubuntu@lubuntu:~$
```

## Assignment 6

Write a script which will make a backup copy of the files given as command line arguments. The last command line argument should specify the destination directory. The script should check if the destination directory exists and create it otherwise. For each filename in the list the script should check the existence and readability. In the case of unsuccessful check, a message should be printed. The script should print a message about the directory creation and report the number of successfully copied files.

Note: In order to access the last command line argument, you can use indirect reference to `$#variable:${!#}`.

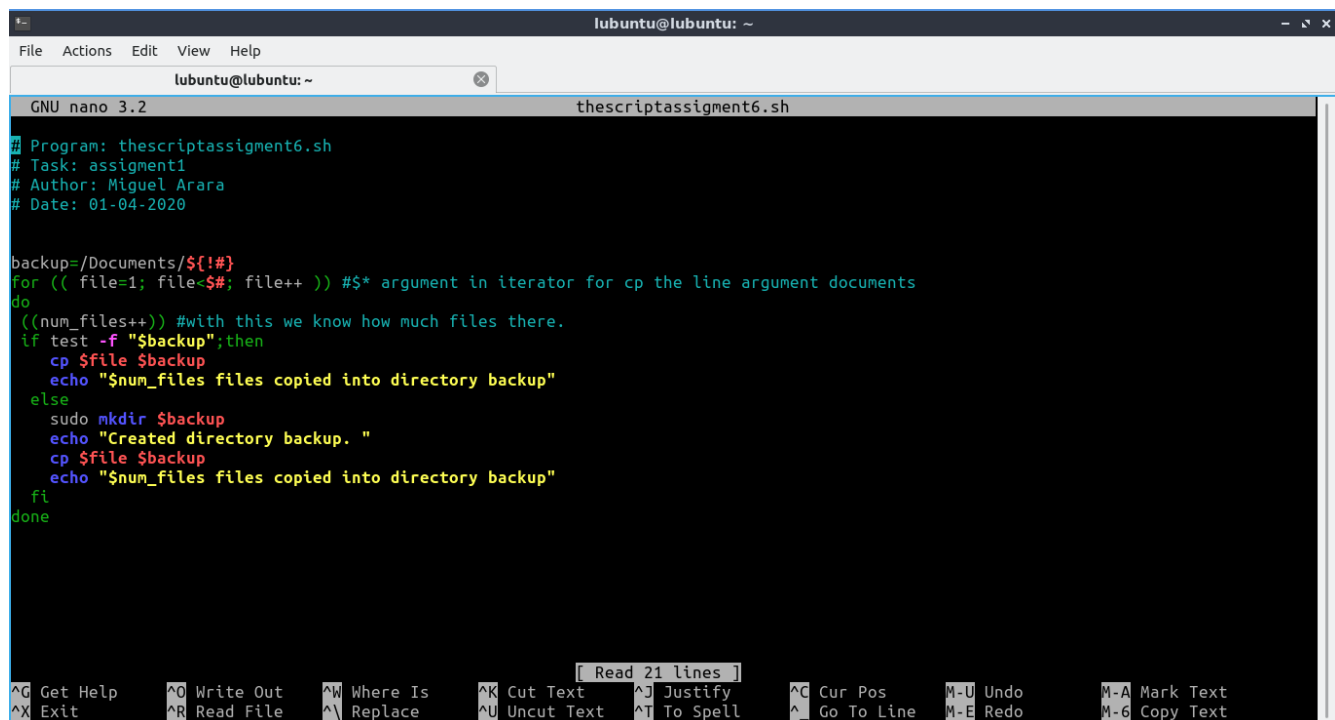
Here is an illustration of the script execution:

```
$ ls assignment6.sh dat1 dat2 dat3 dat4 dat5 dat6$  
./assignment6.sh*backup
```

Created directory backup. 7 files copied into directory backup.

```
$ ls assignment6.sh backup dat1 dat2 dat3 dat4 dat5 dat6  
$ ls backupassignment6.sh dat1 dat2 dat3 dat4 dat5 dat6
```

Upload instruction: Name the script `assignment6.s`



```
lubuntu@lubuntu: ~  
GNU nano 3.2 thescriptassignment6.sh  
# Program: thescriptassignment6.sh  
# Task: assignment1  
# Author: Miguel Arara  
# Date: 01-04-2020  
  
backup=/Documents/${!#}  
for (( file=1; file<${#}; file++ )) # $* argument in iterator for cp the line argument documents  
do  
  ((num_files++)) #with this we know how much files there.  
  if test -f "$backup";then  
    cp $file $backup  
    echo "$num_files files copied into directory backup"  
  else  
    sudo mkdir $backup  
    echo "Created directory backup. "  
    cp $file $backup  
    echo "$num_files files copied into directory backup"  
  fi  
done  
  
[ Read 21 lines ]  
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos   M-U Undo     M-A Mark Text  
^X Exit      ^R Read File  ^_ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line M-E Redo     M-G Copy Text
```

output: (I think that I have a little problem with privileges):

```
lubuntu@lubuntu:~$ nano thescriptassignment6.sh
lubuntu@lubuntu:~$ bash thescriptassignment6.sh assignment1.sh assignment2.sh backup6
mkdir: cannot create directory '/Documents/backup6': No such file or directory
Created directory backup.
cp: cannot stat '1': No such file or directory
1 files copied into directory backup
mkdir: cannot create directory '/Documents/backup6': No such file or directory
Created directory backup.
cp: cannot stat '2': No such file or directory
2 files copied into directory backup
lubuntu@lubuntu:~$
```