IT Workshop I, B.Tech 3rd Semester

भारतीय सूचना प्रौद्योगिकी संस्थान गुवाहाटी

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY GUWAHATI

Assignment 1

Instructions

1. Deadline to submit is 9th August 2023.

Questions

1. Write shell script to perform the following tasks:
   a. Takes a filename as input and checks if the file exists or not. If it exists, display its content; otherwise, create a new file with that name.
   b. Read the file line by line.
   c. Returns the total number of characters, words and line in file.
   d. Display the content of the file in reverse.
   e. Find the frequency of a particular word in a file.
   f. Rename a file.
   g. Takes a C source code file as input, compiles it, and generates an executable with the same name as the source file (without the extension).
   h. Takes a directory name as input and compiles all the C files in that directory and its subdirectories. The resulting executables should be stored in a separate "bin" directory.

Installations

1. **Setting up the Linux environment:**
   We have listed several ways to set up Linux environments on different operating systems.

   a. **Installing linux**
      i. **Setting up a virtual machine (Virtual Box/VM Ware)**
      You can install VMware/VirtualBox and then install a Linux-based OS (e.g. Ubuntu). You can follow the tutorial here.

      ii. **Dual Booting Linux with Windows**
      If your device already has Windows installed, you can dual boot it with a different free and open-source operating system, such as Ubuntu. Here is a tutorial link.

   b. **Windows: Cygwin**
      Cygwin provides a Linux-based terminal to run Linux commands on Windows. To install Cygwin, go to this website and follow each step.

   c. **MAC**
      Since both Linux and Mac are UNIX-based operating systems, you should be able to run most commands without making any modifications.

d. **Online: JSLinux**

It's possible to execute most Linux commands using this excellent online Linux terminal emulator.

2. Linux Commands

a. **Basic Commands and Uses**

Command	Uses	Sample
mkdir	To make a new directory.	`mkdir cs202`
vi / nano	Create/open a file.	`nano test.txt`
cp	To copy the files.	`cp test.txt ./root/cs202`
mv	To move the files	`mv test.txt ./root/cs202`
cd	For changing directory.	`cd cs202`
ls	Show list of contents in a directory.	`ls <directory_name>`
cat	To view the content of a file.	`cat test.txt`
rm	To remove the files.	`rm test.txt`
pwd	To print the current directory.	`pwd`
man	Open manual for a command.	`man <command_name>`

b. **A Use-Case of working using basic commands**

- First, check the current directory using the ***pwd*** command. In this example, we are in the root directory.

```
[root@localhost ~]# pwd
/root
```

- Now we will make a directory with the name “iiitg”.

```
[root@localhost ~]# mkdir iiitg
```

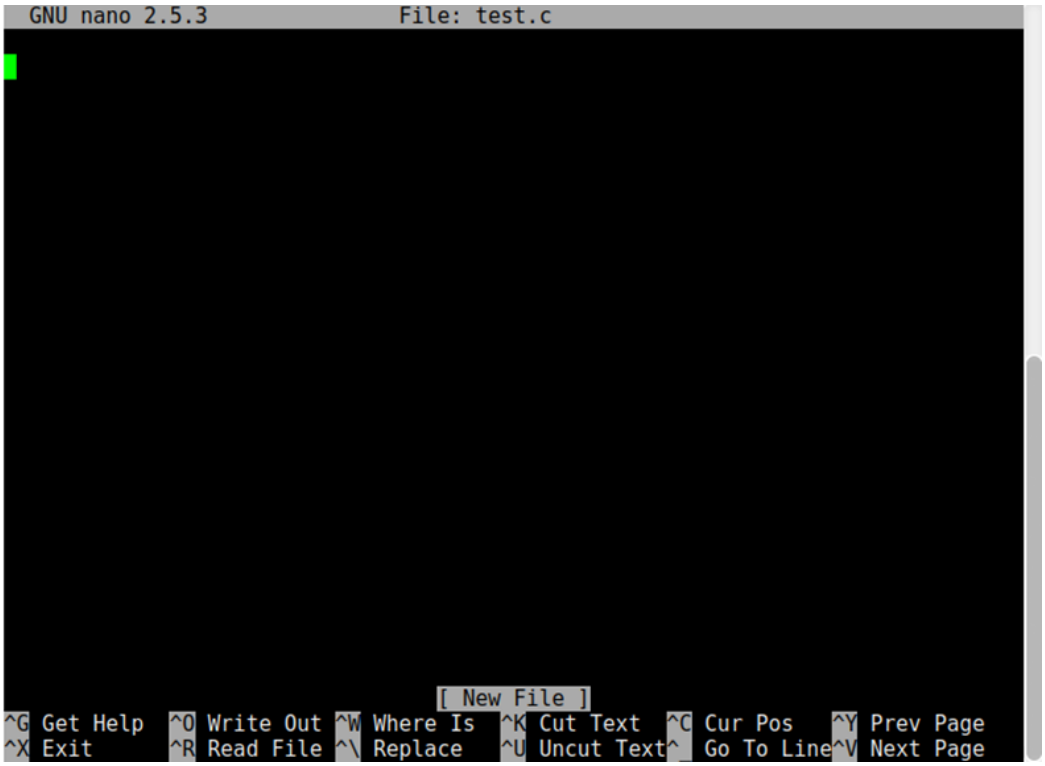
- Visit the “iiitg” directory.

```
[root@localhost ~]# cd iiitg
[root@localhost iiitg]#
```

- Make a **C** program file using the nano editor.

```
[root@localhost iiitg]# nano test.c
```

It will open like the following interface. Here, we write a *main()*, which prints “Hello”.



- To save the file, press ***Ctrl+x*** from the keyboard and then press ***y***.
- To view the file we have created now, use the ***ls*** command.

```
[root@localhost iiitg]# ls
test.c
```

- Compile the c file using the ***gcc*** compiler.

```
[root@localhost iiitg]# gcc test.c -o test
```

- Execute the executable file.

```
[root@localhost iiitg]# ./test
Hello
```

- Moving back to the parent directory.

```
[root@localhost iiitg]# cd ..
```

- Delete the file that we have created.

```
[root@localhost ~]# rm -r iiitg
```

3. Shell Script

a. Some sample shell script

Let's start by creating a simple shell script that prints "Hello, World!" to the console. We'll use Bash, one of the most popular Unix shells.

- Open a text editor and create a new file named *hello.sh*.
- Add the following code to the *hello.sh* file:

```
#!/bin/bash

# This is a comment
echo "Hello, World!"
```

- Save the file.
- Make the script executable using the command *chmod*.

```
[root@localhost ~]# chmod +x hello.sh
```

- Run the shell script.

```
[root@localhost ~]# ./hello.sh
Hello, World!
```

- Shell script example for accepting user input

```
#!/bin/bash

# This is a comment
echo "What's your name?"
read name
echo "Hello, $name!"
```

- Shell script using conditional statements

```
#!/bin/bash

# This is a comment
echo "Enter a number:"
read num

if [ $num -gt 0 ]; then
echo     "The     number     is
positive."
elif [ $num -lt 0 ]; then
echo     "The     number     is
negative."
else
echo "The number is zero."
fi
```

- Shell script using loops

```
#!/bin/bash

# This is a comment
count=5

while [ $count -gt 0 ]; do
echo $count
((count--))
done

echo "Blastoff!"
```