



Karunya INSTITUTE OF TECHNOLOGY AND SCIENCES

(Declared as Deemed to be University under Sec.3 of the UGC Act, 1956)

A CHRISTIAN MINORITY RESIDENTIAL INSTITUTION

AICTE Approved & NAAC Accredited

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LABORATORY RECORD

2020-2021

Subject Code

18CS2064

Subject Name

Open Source Technologies Lab

Register No. URK17CS023

It is hereby certified that this is the bonafide record of work done by
SANDEEP.M during the Even Semester of the academic year **2020-2021** and submitted for the University Practical Examination held on **24/11/2020**.

Staff-in-Charge

Name: SHIBIN D

ID: 1756

LIST OF EXPERIMENTS

Ex. No	DATE	TITLE
1	18.08.2020	ADVANCED LINUX COMMANDS
2	25.08.2020	KERNEL CONFIGURATION AND COMPIRATION
3	29.08.2020	COMPILING FROM THE SOURCE
4	01.09.2020	VIRTUALIZATION
5	08.09.2020	PACKAGE MANAGEMENT SYSTEM
6	22.09.2020	REPOSITORY IN GITHUB
7	06.10.2020	BASIC PYTHON PROGRAMMING
8	03.11.2020	BASIC PERL PROGRAMS
9	10.11.2020	FUNDAMENTAL OPERATIONS USING PERL
10	17.11.2020	KERNEL INSTALLATION

Ex 1

ADVANCED LINUX COMMANDS

Date: 18.08.20

Aim:

To study and implement the Linux commands

Description:

Sl. No.	Command Name	Meaning	options
1.	ls	List files and/or directories.	-a, --all do not ignore entries starting with. -A, --almost-all do not list implied. and. --author with -l, print the author of each file -b, --escape print C-style escapes for nongraphic characters --block-size=SIZE
2.	Who am i	This command reveals the user who is currently logged in.	-a, --all =same as -b -d --login -p -r -t -T -u -b, --boot time of last system boot -d, --dead print dead processes -H, --heading print line of column headings -l, --login print system login processes --lookup attempt to canonicalize hostnames via DNS

18CS2064 - Open Source technologies Lab | URK17CS023

3.	pwd	prints the absolute path to the current working directory	<ul style="list-style-type: none"> -L, --logical use PWD from the environment, even if it contains symlinks -P, --physical avoid all symlinks
4.	cal	Displays the calendar of the current month	<ul style="list-style-type: none"> -1, --one Display single month output. (This is the default.) -3, --three Display prev/current/next month output. -s, --Sunday Display Sunday as the first day of the week. -m, --Monday Display Monday as the first day of the week. -j, --Julian Display Julian dates (days one-based, numbered from January 1). <u>-y, --year</u>
5.	echo	This command will echo whatever you provide it.	<ul style="list-style-type: none"> -n do not output the trailing newline -e enable interpretation of backslash escapes -E disable the interpretation of backslash escapes (default)

6.	date	Displays current time and date.	<p>-d, --date=STRING display time described by STRING, not 'now'</p> <p>-f, --file=DATEFILE like --date once for each line of DATAFILE</p>
7.	tty	Displays the current terminal.	
8.	id	This command prints user and groups (UID and GID) of the current user.	<p>-a ignore, for compatibility with other versions</p> <p>-Z, --context print only the security context of the current user</p> <p>-g, --troup print only the effective group ID</p> <p>-G, --groups print all group IDs</p> <p>-n, --name print a name instead of a number, for -ugG</p>
9.	clear	This command clears the screen.	
10.	man	To show manual page	
11.	cd	Change the current working directory to the directory provided as an argument.	

12.	mkdir	To create a directory, the ‘mkdir’ command is used.	
13.	touch	For creating an empty file, use the touch command.	
14.	cp	Copy files and directories	
15.	mv	Move files or directories. The 'mv' command works like 'cp' command, except that the original file is removed. But, the mv command can be used to rename the files (or directories).	
16.	rmdir	the command removes any empty directories, but cannot delete a directory if a file is present in it.	
17.	file	The file command determines the file type of a given file.	
18.	cat	The 'cat' command is actually a concatenator but can be used to view the contents of a file.	
19.	head	Displays the first few lines of a file. By default, the ‘head’ command displays the first 10 lines of a file.	
20.	tail	the ‘tail’ command shows the last 10 lines by default	<ul style="list-style-type: none"> -c, --bytes=[-]K print the first K bytes of each file -n, --lines=[-]K print the first K lines instead of the first 10 -q, --quiet, --silent never print headers giving file names

21.	wc	This command counts lines, words, and letters of the input given to it.	
22.	grep	The ‘grep’ command searches for a pattern in a file (or standard input).	
23.	vi	Visual editor	
24.	alias	The ‘alias’ is another name for a command.	
25.	history	shows the commands you have entered on your terminal so far.	
26.	passwd	To change your password	
27.	help	With almost every command, ‘--help’ option shows usage summary for that command.	
28.	chmod	The <i>chmod</i> command lets you change access permissions for a file.	
29.	stat	To check the status of a file. This provides more detailed information about a file than ‘ls -l’ output.	<ul style="list-style-type: none"> -L, --dereference follow links -f, --file-system display file system status instead of file status -c --format=FORMAT use the specified FORMAT instead of the default; output a newline after each use of FORMAT --printf=FORMAT
30.	ln	The ln command is used in Linux to create links.	

Exercise

1. List the contents of user's home directory including the hidden files

```
[urk17cs023@code ~]$ ls -a
.
..
7.1.cpp          ex10.3.cpp          .num.cpp.swp
7.1.cppvi        ex103.cpp          occ.cpp
10.cpp           ex11.1.cpp          occu.cpp
1.2.c            ex11.2.cpp          op.c
1a.c             .ex11.2.cpp.swp    operations.c
1b.c             ex31.cpp          operations.c.save
1c.c             7.6.c              pal.c
1d.c             7.7.c              pc.c
1e.c             8.1.cpp          perfect.c
1f.c             8.cpp              pf1.c
2a.c             8ex.cpp          pf.c
2b.c             9.1.cpp          pointer.cpp
2c.c             9.c              .pointer.cpp.swp
2darray.c        9.cpp          porc.
2d.c             a1              porc.c
2d.c.save        a2              power.cpp
2dd.c            a3              qe.c
2dm.c            adfact.c        relopr.c
2f.c             a.out            rev.c
2g.c             arfun.c        sam.c
2h.c             arm.c           sample1.cpp
2i.c             array.c         sample.a
```

2. List the content of /var directory?

```
[urk17cs023@code ~]$ cd /var
[urk17cs023@code var]$ ls
adm      centrifyda  db      gopher   local   mail   preserve  tmp
cache   centrifydc  empty   kerberos lock   nis    run     yp
centrify crash      games   lib      log    opt    spool
[urk17cs023@code var]$ █
```

3. Create two directories named dir1 & dir2

```
[urk17cs023@code ~]$ mkdir dir11 diir22
[urk17cs023@code ~]$ lm
-bash: lm: command not found
[urk17cs023@code ~]$ ls
10.cpp      3a.c      6.1.cpp    82.c      ascii.c      dir1
1.2.c       3b.c      6.2.cpp    83a.c     bitwise.c    dir11
1a.c        3c.c      6.cpp      83.c      bmi.c       dir2
1b.c        3d.c      6e.cpp    8a.c      books.txt   dir22
1c.c        3e.c      7.1.c     8.cpp     borrowed.txt dir3
1d.c        3f.c      71.c      8ex.cpp   b.out      dir4
1e.c        3g.c      7.1.cpp   91a.c     classname   dirr1
1f.c        3h.c      7.1.cppvi 9.1.cpp   co.c       dirr2
2a.c        3i.c      7.2.c     9a1.c    conoperator.c display.txt
2b.c        40.cpp    72.c      9a2.c    coperator.c  distance.c
2c.c        4.10.c    7.3.c     9b.c     coperator.c.save el.c
2darray.c  42.1.cpp  7.4.c     9.c      cricket.cpp enum.c
2d.c         4.2.c     7.5.c     9.cpp    currency.cpp even.c
2d.c.save   42.cpp    7.6.c     9yacc.y cust.c      ex
2dd.c       4.2.cy    7.7.c     a1       data.c      ex10.1.cpp
2dm.c       4.7.c     7a.c      a2       demo2.sh   ex101.cpp
2f.c        5.1.cpp    7b.c      a3       demo3.sh   ex10.2.cpp
2g.c        51.sh     81a.c     adfact.c  demo.sh    ex10.3.cpp
2h.c        5.2.2.cpp 81.c      a.out    diaplay.txt ex103.cpp
2i.c        5.2.cpp    81c.c     arfun.c  diir1     ex11.1.cpp
2l.c        5.3.3.cpp 8.1.cpp   arm.c    diir2     ex11.2.cpp
3.4.cpp    5.3.cpp    82a.c     array.c  diir22   ex31.cpp
```

4. Create a hidden directory with your name?

```
[urk17cs023@code ~]$ mkdir sandy.sandy
[urk17cs023@code ~]$ ls -a
.            3f.c      7.1.cppvi 9.cpp      .cricket.cpp.swp  ex10.3.cpp  .ex9.1.cpp.swp  expt55.sh  memory1.c  q1.sh
..           3g.c      7.2.c     9yacc.y currency.cpp    ex103.cpp   ex9.2.cpp    f1.txt    memory.c  q2.sh
10.cpp      3h.c      72.c     a1          cust.c      ex11.1.cpp  ex9.3.cpp    f2.txt    merge.c   q3.sh
1.2.c       3i.c      7.3.c     a2          data.c      ex11.2.cpp  ex9.3.cpp.swp f3.txt    money.cpp q4.sh
1a.c        40.cpp    7.4.c     a3          demo2.sh   .ex11.2.cpp.swp exam1.c    f4.txt    mozilla  q51.sh
1b.c        4.10.c    7.5.c     adfact.c  demo3.sh   exam1.sh    fev.c    mul.c    q5.sh
1c.c        42.1.cpp  7.6.c     a.out      demo.sh    .ex31.cpp.swp exam2.sh    fib.c    ndo.c    ge.c
1d.c        42.1.cpp.swp 7.7.c     arfun.c  diaplay.txt ex31.1.sh   .exam3.sh.swp  fib.c.save newdir  relopr.c
1e.c        4.2.c     7a.c      arm.c    dir1      ex3.2.sh   exp11.2.cpp  filename1.c newfile.txt rev.c
1f.c        42.cpp    7b.c      array.c  dir2      ex3.3.sh   exp3a.sh    filename.c  num.cpp   sam.c
2a.c        4.2.cy    81a.c     ascii.c  dir22     ex3.4.sh   exp3b.sh    filename.c.save .num.cpp.swp sample1.cpp
2b.c        4.7.c     81.c      bash_history dir1      ex3.5.sh   exp3c.sh    filenew2  occ.cpp   sample23.txt
2c.c        5.1.cpp    81c.c     bash_logout dir11     ex4.1.sh   exp3d.sh    for.c    occu.cpp sample.a
2darray.c  5.1.cpp.swp 8.1.cpp   bash_profile dir2      ex4.2.sh   exp3e.sh    friend1.cpp op.c    sample.c
2d.c        5.1.cpp.swp 82a.c     bashrc   dir22     ex4.3.sh   exp4.sh     friend2.cpp operations.c sample.cc
2d.c.save   51.sh     82.c      bitwise.c dir3      ex4.4.sh   exp51.sh    friend8.cpp operations.c.sample save sample.cpp
2d.c.save  51.sh     82.c      bitwise.c dir3      ex4.4.sh   exp51.txt   .hidden  pal.c    sampleee.c
2d.c.save  51.sh     82.c      bitwise.c dir3      ex4.5.sh   exp52.sh    .hidden1 pc.c    san.cpp
2d.c.save  51.sh     82.c      bitwise.c dir3      ex4.6.sh   exp53.sh    k5login  perfect.c sand1.txt
2g.c        5.2.cpp    8.cpp      b.out    diaplay.txt ex4.6.sh   exp53.sh    largest.c pf1.c    .sanddep
2h.c        5.3.3.cpp 8ex.cpp   .c.cpp.swp  distance.c ex6.1.cpp   exp54.sh    lex1    pf.c    sand.txt
2i.c        5.3.cpp    91a.c     classname el.c     ex6.2.cpp   exp55.sh    lex.l    pip5    sandy
2l.c        6.1.cpp    .91a.c.swp co.c     .emacs    ex7.10.cpp  exp5.sh     lex.yy.c pip6    .sandy
3.4.cpp    6.2.cpp    .91a.c.swp config   enum.c    ex7.12.cpp  exp6.2.sh  lib1.cpp  pointer.cpp sandy.sandy
```

5. Display the content of a hidden directory.

```
[urk17cs023@code ~]$ mkdir .hidden11
[urk17cs023@code ~]$ ls -a
. 3f.c 7.1.cppvi 9.cpp  .cricket.cpp.swp ex10.3.cpp  .ex9.1.cpp.swp expt55.sh
.. 3g.c 7.2.c 9yacc.y  currency.cpp ex103.cpp  ex9.2.cpp f1.txt
10.cpp 3h.c 72.c a1  cust.c ex11.1.cpp  ex9.3.cpp f2.txt
1.2.c 3i.c 7.3.c a2  data.c ex11.2.cpp  .ex9.3.cpp.swp f3.txt
la.c 40.cpp 7.4.c a3  demo2.sh  .ex11.2.cpp.swp exam1.c f4.txt
1b.c 4.10.c 7.5.c adfact.c demo3.sh  ex31.cpp  exam1.sh fev.c
1c.c 42.1.cpp 7.6.c a.out  demo.sh  .ex31.cpp.swp exam2.sh fib.c
1d.c 42.1.cpp.swp 7.7.c arfun.c diaplay.txt  ex31.sh  .exam3.sh.swp fib.c.save
1e.c 4.2.c 7a.c arm.c diir1  ex3.2.sh  exp11.2.cpp filename1.c
1f.c 42.cpp 7b.c array.c diir2  ex3.3.sh  exp3a.sh filename.c
2a.c 4.2.cy 81a.c ascii.c diir22  ex3.4.sh  exp3b.sh filename.c.s
2b.c 4.7.c 81.c .bash_history  diir1  ex3.5.sh  exp3c.sh filenew2
2c.c 5.1.cpp 81c.c .bash_logout  diir11  ex4.1.sh  exp3d.sh for.c
2darray.c 5.1.cpp.swp 8.1.cpp .bash_profile  dir2  ex4.2.sh  exp3e.sh friend1.cpp
2d.c 5.1.cpp.swp 82a.c .bashrc  dir22  .ex4.2.sh.swp  .exp3e.sh.swp friend2.cpp
2d.c.save 51.sh 82.c bitwise.c dir3  ex4.3.sh  exp4.sh friend8.cpp
2dd.c 5.2.2.cpp 83a.c bmi.c dir4  ex4.4.sh  exp51.sh .hidden
2dm.c 5.2.2.cpp.swp 83.c books.txt  dirr1  .ex4.4.sh.swp  exp51.txt .hidden11
2f.c 5.2.cpp 8a.c borrowed.txt  dirr2  ex4.5.sh  exp52.sh .hidden11
2f.c 5.2.2 9  borrowed  dirr2  ex4.6.sh  exp52.sh .hidden11
2f.c 5.2.2 9  borrowed  dirr2  ex4.6.sh  exp52.sh .hidden11
2f.c 5.2.2 9  borrowed  dirr2  ex4.6.sh  exp52.sh .hidden11
```

6. Display the calendar of 2020.

```
[urk17cs023@code ~]$ cal 2020
          2020
      January   February   March
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
  1 2 3 4       1 2 3 4 5 6 7 1 2 3 4 5 6 7
  5 6 7 8 9 10 11 2 3 4 5 6 7 8 8 9 10 11 12 13 14
12 13 14 15 16 17 18 9 10 11 12 13 14 15 15 16 17 18 19 20 21
19 20 21 22 23 24 25 16 17 18 19 20 21 22 22 23 24 25 26 27 28
26 27 28 29 30 31 23 24 25 26 27 28 29 29 30 31

      April   May   June
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
  1 2 3 4       1 2 1 2 3 4 5 6
  5 6 7 8 9 10 11 3 4 5 6 7 8 9 7 8 9 10 11 12 13
12 13 14 15 16 17 18 10 11 12 13 14 15 16 14 15 16 17 18 19 20
19 20 21 22 23 24 25 17 18 19 20 21 22 23 21 22 23 24 25 26 27
26 27 28 29 30 24 25 26 27 28 29 30 28 29 30
31

      July   August   September
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
  1 2 3 4       1 1 2 3 4 5
  5 6 7 8 9 10 11 2 3 4 5 6 7 8 6 7 8 9 10 11 12
12 13 14 15 16 17 18 9 10 11 12 13 14 15 13 14 15 16 17 18 19
19 20 21 22 23 24 25 16 17 18 19 20 21 22 20 21 22 23 24 25 26
26 27 28 29 30 31 23 24 25 26 27 28 29 27 28 29 30
31

      October   November   December
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
  1 2 3       1 2 3 4 5 6 7 1 2 3 4 5
  4 5 6 7 8 9 10 8 9 10 11 12 13 14 6 7 8 9 10 11 12
11 12 13 14 15 16 17 15 16 17 18 19 20 21 13 14 15 16 17 18 19
18 19 20 21 22 23 24 22 23 24 25 26 27 28 20 21 22 23 24 25 26
25 26 27 28 29 30 31 29 30 27 28 29 30 31
```

7. Copy the file /etc/passwd file to current directory with sample.txt as the filename

```
[urk17cs023@code ~]$ cat /etc/passwd > sample.txt
[urk17cs023@code ~]$ cat /etc/passwd > sample1.txt
[urk17cs023@code ~]$ cat sample.txt
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
chrony:x:998:996:/var/lib/chrony:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
nginx:x:997:995:Nginx web server:/var/lib/nginx:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
[urk17cs023@code ~]$ █
```

8. Create a file test1.txt using Vim editor with the following contents to it

Name	RegNo	ResearchInterest
Melvin	07af501	GridComputing
Mithin	07af502	ClusterComputing
James	07af503	ImageProcessing
Jane	07af504	Networking
Caroline	07af505	ClusterComputing
Binu	07af506	GridComputing
Aaron	07af507	ImageProcessing
Selvin	07af508	Networking

Jerwin	07af509	WirelessNetworks
Arun	07af510	GridComputing

Answer the following questions

- a) Display the student names who are having Research Interest as GridComputing

```
[urk17cs023@code ~]$ grep Grid text.txt | cut -f 1  
Melvin  
Binu  
Arun
```

- b) List all the student names & RegNo in the class

```
[urk17cs023@code ~]$ cut -f 1,2 text.txt  
Name      RegNo  
Melvin    07af501  
Mithin    07af502  
James     07af503  
Jane      07af504  
Caroline   07af505  
Binu      07af506  
Aaron     07af507  
Selvin    07af508  
Jerwin    07af509  
Arun      07af510  
[urk17cs023@code ~]$ █
```

- c) List the count of students who have an interest as ImageProcessing and store the result in another file.

```
[urk17cs023@code ~]$ grep ImageProcessing text.txt|wc -l > text  
[urk17cs023@code ~]$ cat text  
2
```

d) Display the first two rows and last two and store them into another file.

```
[urk17cs023@code ~]$ head -3 text.txt >> display.txt | tail -2 text.txt >> display.txt  
[urk17cs023@code ~]$ cat display.txt  
2  
[urk17cs023@code ~]$ vi display.txt
```

```
Jerwin 07af509 WirelessNetworks  
Arun 07af510 GridComputing  
Name RegNo ResearchInterest  
Melvin 07af501 GridComputing  
Mithin 07af502 ClusterComputing
```

9. Display the contents of the file test1.txt without any blank lines

```
[urk17cs023@code ~]$ grep -v '^$' text.txt  
Name RegNo ResearchIntrest  
Melvin 07af501 GridComputing  
Mithin 07af502 ClusterComputing  
James 07af503 ImageProcessing  
Jane 07af504 Networking  
Caroline 07af505 ClusterComputing  
Binu 07af506 GridComputing  
Aaron 07af507 ImageProcessing  
Selvin 07af508 Networking  
Jerwin 07af509 WirelessNetworks  
Arun 07af510 GridComputing
```

10. Move the file sample.txt from dir1 directory to dir2 directory

```
[urk17cs023@code ~]$ mv sample.txt dirr2  
[urk17cs023@code ~]$ cd dirr2  
[urk17cs023@code dirr2]$ ls  
sample.txt  
[urk17cs023@code dirr2]$
```

11. Change directory into dir2 directory

```
[urk17cs023@code dirr2]$ cd ..
[urk17cs023@code ~]$ cd dirr2
```

12. Check whether the file sample.txt is present their

```
(urk17cs023@code ~)$ cd dirr2
(urk17cs023@code dirr2]$ cat sample.txt
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin.sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
chrony:x:998:996::/var/lib/chrony:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev
/null:/sbin/nologin
nginx:x:997:995:Nginx web server:/var/lib/nginx:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
```

13. Rename the file sample.txt to new.txt and check whether sample.txt is there or not?

```
[urk17cs023@code dirr2]$ touch new.txt
[urk17cs023@code dirr2]$ mv sample.txt new.txt
[urk17cs023@code dirr2]$ cat sample.txt
cat: sample.txt: No such file or directory
```

14. Remove the directory dir1

```
[urk17cs023@code ~]$ rm -r dir1  
[urk17cs023@code ~]$ cd dir1
```

15. Display last 3 lines of the file test1.txt

```
[urk17cs023@code ~]$ tail -3 text.txt  
Selvin 07af508 Networking  
Jerwin 07af509 WirelessNetworks  
Arun    07af510 GridComputing
```

16. Display all the commands you have executed so far and save the list into a file named todayhistory.txt

```
1151 cd ..  
1152 cd dirr2  
1153 cat sample.txt  
1154 touch new.txt  
1155 mv sample.txt new.txt  
1156 cat sample.txt  
1157 cd ..  
1158 cal 12 2020  
1159 rm -r dir1  
1160 cd dir1  
1161 tail -3 text.txt  
1162 history  
[urk17cs023@code ~]$ history > todayhistory.txt
```

17. How many files are present under your home directory?

```
[urk17cs023@code ~]$ ls -a | wc -l  
226
```

18. Perform the sorting of three files and store the sorted file in the fourth file.

```
[urk17cs023@code ~]$ cat > f1.txt
First[urk17cs023@code ~]$ cat > f2.txt
Second[urk17cs023@code ~]$
[urk17cs023@code ~]$ cat > f3.txt
Third[urk17cs023@code ~]$
[urk17cs023@code ~]$ sort f1.txt f2.txt f3.txt > f4.txt
[urk17cs023@code ~]$ cat f4.txt
First
Second
Third
```

19. Change the permission of your newly created file such that the group users and others don't access any type of access.

```
[urk17cs023@code ~]$ chmod 700 f4.txt
[urk17cs023@code ~]$ ls -l f4.txt
-rwx----- 1 urk17cs023 urk17cs023 19 Dec 15 23:59 f4.txt
[urk17cs023@code ~]$ █
```

20. Display the network status on the shell.

```
[urk17cs023@code ~]$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp     0      0 code.karunya.edu:58340   code.karunya.edu:hbc1  TIME_WAIT
tcp     0      0 code.karunya.edu:58262   code.karunya.edu:hbc1  TIME_WAIT
tcp     0      0 code.karunya.edu:https   162.158.167.74:16612 ESTABLISHED
tcp     0      0 code.karunya.edu:nfs    192.168.0.32:ftps-data ESTABLISHED
tcp     0      0 code.karunya.edu:58390   code.karunya.edu:hbc1  TIME_WAIT
tcp     0      0 code.karunya.edu:https   162.158.198.185:38434 ESTABLISHED
tcp     0      0 code.karunya.edu:57732   code.karunya.edu:hbc1  ESTABLISHED
tcp     0      0 code.karunya.edu:https   162.158.198.181:27398 ESTABLISHED
tcp     0      0 code.karunya.edu:https   162.158.166.55:57302 ESTABLISHED
tcp     0      0 code.karunya.edu:hbc1   code.karunya.edu:57796  ESTABLISHED
tcp     0      0 code.karunya.edu:https   162.158.198.159:29536 ESTABLISHED
tcp     0      0 code.karunya.edu:https   162.158.50.213:37802 ESTABLISHED
tcp     0      0 code.karunya.edu:58394   code.karunya.edu:hbc1  ESTABLISHED
tcp     0      0 code.karunya.edu:https   162.158.198.159:29860 ESTABLISHED
tcp     0      0 code.karunya.edu:https   162.158.50.215:34996 ESTABLISHED
tcp     0      0 code.karunya.edu:nfs    192.168.0.34:790    ESTABLISHED
tcp     0      0 code.karunya.edu:58120   code.karunya.edu:hbc1  ESTABLISHED
tcp     0      0 code.karunya.edu:58290   code.karunya.edu:hbc1  TIME_WAIT
tcp     0      0 code.karunya.edu:https   172.69.134.127:43822 ESTABLISHED
tcp     0      0 code.karunya.edu:58280   code.karunya.edu:hbc1  TIME_WAIT
tcp     0      0 code.karunya.edu:58366   code.karunya.edu:hbc1  TIME_WAIT
tcp     0      0 code.karunya.edu:58324   code.karunya.edu:hbc1  TIME_WAIT
tcp     0      0 code.karunya.edu:58278   code.karunya.edu:hbc1  TIME_WAIT
tcp     0      0 code.karunya.edu:https   162.158.50.249:30884 ESTABLISHED
tcp     0      383 code.karunya.edu:https  162.158.50.228:61028 ESTABLISHED
```

21. Compares any two files and search for both common and exclusive features

```
[urk17cs023@code ~]$ diff sand.txt sand1.txt
1c1
< sand snkqkfwnq fenjkfnfn efkjfkjqf
---
> sand snkqkfwnq fbkjbfkqbffqq
```

22. Display the user ID, process ID, and parent process ID.

```
[urk17cs023@code ~]$ ps -f
UID      PID  PPID  C STIME TTY          TIME CMD
urk17cs+ 8453  8443  0 10:28 pts/4    00:00:00 -bash
urk17cs+ 10451  8453  0 10:43 pts/4    00:00:00 cat
urk17cs+ 10638  8453  0 10:44 pts/4    00:00:00 cat
urk17cs+ 10835  8453  0 10:45 pts/4    00:00:00 ps -f
[urk17cs023@code ~]$
```

23. Report disk usages of the file system.

```
[urk17cs023@code ~]$ du -h
0      ./mozilla/plugins
0      ./mozilla/extensions
0      ./mozilla
8.0K   ./local/share/jupyter/runtime
8.0K   ./local/share/jupyter
8.0K   ./local/share
8.0K   ./local
16K    ./config/neofetch
16K    ./config
0      ./newdir
0      ./sanddeep
4.0K   ./diir1
4.0K   ./diir2
0      ./hidden
0      ./dirr1
4.0K   ./dirr2
0      ./hidden1
0      ./dir1
0      ./dir3
0      ./dir4
4.0K   ./sandy.sandy
1.4M   .
```

24. Display the statistics of all ports connected to a network.

```
[urk17cs023@code ~]$ netstat -l
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp     0      0 0.0.0.0:nfs              0.0.0.0:*               LISTEN
tcp     0      0 0.0.0.0:46053             0.0.0.0:*               LISTEN
tcp     0      0 0.0.0.0:sunrpc            0.0.0.0:*               LISTEN
tcp     0      0 0.0.0.0:http              0.0.0.0:*               LISTEN
tcp     0      0 0.0.0.0:mountd             0.0.0.0:*               LISTEN
tcp     0      0 0.0.0.0:34067             0.0.0.0:*               LISTEN
tcp     0      0 0.0.0.0:ssh               0.0.0.0:*               LISTEN
tcp     0      0 0.0.0.0:hbc1              0.0.0.0:*               LISTEN
tcp     0      0 localhost:smtp             0.0.0.0:*               LISTEN
tcp     0      0 0.0.0.0:https             0.0.0.0:*               LISTEN
tcp6    0      0 [::]:nfs                [::]:*                 LISTEN
tcp6    0      0 [::]:sunrpc              [::]:*                 LISTEN
tcp6    0      0 [::]:http                [::]:*                 LISTEN
tcp6    0      0 [::]:mountd              [::]:*                 LISTEN
tcp6    0      0 [::]:ssh                 [::]:*                 LISTEN
tcp6    0      0 [::]:57431               [::]:*                 LISTEN
tcp6    0      0 localhost:smtp             [::]:*                 LISTEN
tcp6    0      0 [::]:https               [::]:*                 LISTEN
tcp6    0      0 [::]:39228               [::]:*                 LISTEN
udp     0      0 0.0.0.0:39547             0.0.0.0:*               LISTEN
udp     0      0 0.0.0.0:mountd             0.0.0.0:*               LISTEN
udp     0      0 0.0.0.0:sunrpc            0.0.0.0:*               LISTEN
udp     0      0 0.0.0.0:49440             0.0.0.0:*               LISTEN
udp     0      0 localhost:323              0.0.0.0:*               LISTEN
udp     0      0 localhost:863              0.0.0.0:*               LISTEN
udp     0      0 0.0.0.0:xact-backup        0.0.0.0:*               LISTEN
udp     0      0 0.0.0.0:nfs               0.0.0.0:*               LISTEN
tcp6    0      0 [::]:mountd              [::]:*                 LISTEN
```

25. Display the uptime of the system.

```
[urk17cs023@code ~]$ uptime
 10:47:16 up 11 days, 23:07, 140 users,  load average: 0.02, 0.04, 0.05
[urk17cs023@code ~]$
```

26. Julian day.

```
[urk17cs023@code ~]$ date
Mon Aug 24 10:47:48 IST 2020
[urk17cs023@code ~]$ date +%j
237
```

27. IP information.

```
[urk17cs023@code ~]$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens32: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:50:56:93:66:93 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.29/24 brd 192.168.0.255 scope global noprefixroute ens32
        valid_lft forever preferred_lft forever
    inet6 fe80::1450:18ba:187f:1f02/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

28. Display only the free space in the system.

```
[urk17cs023@code ~]$ df -k
Filesystem           1K-blocks   Used Available Use% Mounted on
devtmpfs                  3992636      0  3992636   0% /dev
tmpfs                     4004520      0  4004520   0% /dev/shm
tmpfs                     4004520  427272  3577248  11% /run
tmpfs                     4004520      0  4004520   0% /sys/fs/cgroup
/dev/mapper/centos_kitscode-root  68066844 3014196  65052648   5% /
/dev/sda1                   1942528  334256  1608272  18% /boot
/dev/mapper/centos_kitscode-home 24404336  32992  24371344   1% /home
/dev/mapper/centos_kitscode-data 97609148 26097684  71511464  27% /data
/dev/mapper/centos_kitscode-var 10004480 9313116   691364  94% /var
tmpfs                      800908      0   800908   0% /run/user/1010875228
tmpfs                      800908      0   800908   0% /run/user/1010875226
tmpfs                      800908      0   800908   0% /run/user/1010883088
tmpfs                      800908      0   800908   0% /run/user/1010875242
tmpfs                      800908      0   800908   0% /run/user/1010875247
tmpfs                      800908      0   800908   0% /run/user/1010875234
tmpfs                      800908      0   800908   0% /run/user/1010883450
```

29. Display the configuration information of your network.

```
[urk17cs023@code ~]$ netstat -nr
Kernel IP routing table
Destination     Gateway         Genmask        Flags  MSS Window irtt Iface
0.0.0.0         192.168.0.254  0.0.0.0        UG            0 0          0 ens32
192.168.0.0     0.0.0.0        255.255.255.0  U             0 0          0 ens32
```

Results:

The Linux commands are studied and executed.

Video: <https://drive.google.com/file/d/17ckiETOsfDmXvv8hJpwI3BMSWkcgj7n/view>

Ex 2 KERNEL CONFIGURATION, COMPILEDATION AND INSTALLATION

Date: 25.08.20

Aim:

To study and implement the kernel configuration, compilation and installation.

Description:

The Linux Kernel :

The main purpose of a computer is to run a predefined sequence of instructions, known as a program. A program under execution is often referred to as a process.

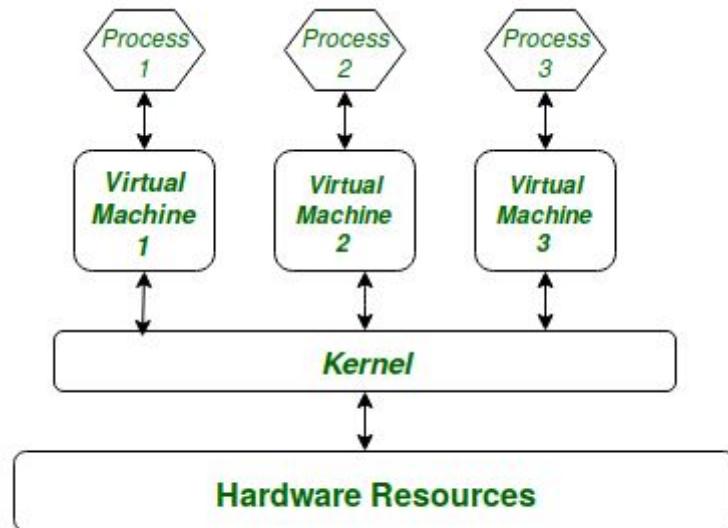
Now, most special purpose computers are meant to run a single process, but in a sophisticated system such a general purpose computer, are intended to run many processes simultaneously. Any kind of process requires hardware resources such as Memory, Processor time, Storage space, etc.

In a General Purpose Computer running many processes simultaneously, we need a middle layer to manage the distribution of the hardware resources of the computer efficiently and fairly among all the various processes running on the computer. This middle layer is referred to as the kernel. Basically the kernel virtualizes the common hardware resources of the computer to provide each process with its own virtual resources. This makes the process seem as it is the sole process running on the machine. The kernel is also responsible for preventing and mitigating conflicts between different processes.

The Core Subsystems of the Linux Kernel are as follows:

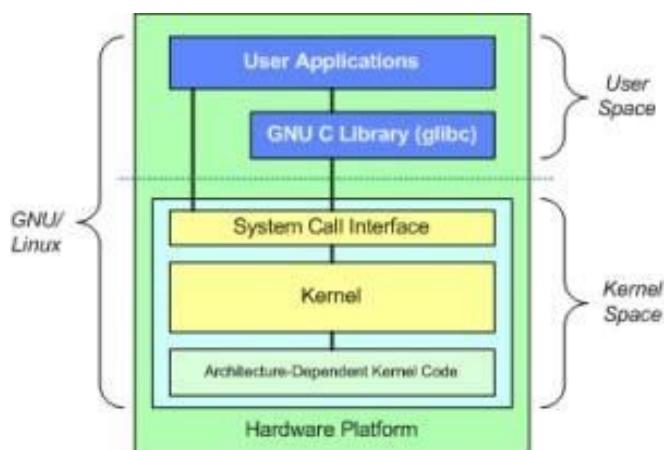
1. The Process Scheduler
2. The Memory Management Unit (MMU)
3. The Virtual File System (VFS)
4. The Networking Unit
5. Inter-Process Communication Unit

This schematically represented below:



Architecture of system kernel :

We can think of Linux Kernel architecture to be divided into two levels – User Space and Kernel Space.



At the top is the user space. Below the user space is the kernel space. Here, the Linux kernel exists.

User Space:

This is where the user applications are executed. There is also the GNU C Library (glibc). This provides the system call interface that connects to the kernel and provides the mechanism to transition between the user-space application and the kernel.

Kernel Space:

Here, the Linux Kernel exists which can be further divided into three levels. At the top is the system call interface, which implements the basic functions such as read and write. Below the system call interface is the kernel code, which can be more accurately defined as the architecture-independent kernel code. This code is common to all of the processor architectures supported by Linux. Below this is the architecture-dependent code, which forms what is more commonly called a BSP (Board Support Package). This code serves as the processor and platform-specific code for the given architecture.

Sl. N. o.	Command Name	Meaning	Description
1.	rpm -qa kernel-devel	It displays the version of the kernel.	Kernel-devel - This package provides kernel headers and makes files sufficient to build modules against the kernel package.
2.	uname -r	uname displays the information about the system.	The command ‘uname‘ displays the information about the system. option : -a It prints all the system information in the following order: Kernel name, network node hostname, kernel release date, kernel version, machine hardware name, hardware platform, operating system

			<p>-s It prints the kernel name.</p> <p>-n It prints the hostname of the network node</p> <p>-r It prints the kernel release date</p> <p>-v It prints the version of the current kernel</p>
3.	tar	tar' stands for tape archive, is used to create Archive and extract the Archive files	<p>tar command in Linux is one of the important commands which provides archiving functionality in Linux. We can use Linux tar command to create compressed or uncompressed Archive files and also maintain and modify them.</p> <p>Options:</p> <ul style="list-style-type: none"> -c : Creates Archive -x : Extract the archive -f : creates archive with given filename -t : displays or lists files in archive file -u : archives and adds to an existing archive file -v : Displays Verbose Information -A : Concatenates the archive files -z : zip, tells tar command that create tar file using gzip -j : filter archive tar file using tbzip -W : Verify a archive file -r : update or add file or directory in already existed .tar file
4.	ln	A symbolic link, also known as a symlink or soft link, is a special type of file that points to another file or directory.	<p>There are two types of links in Linux/UNIX systems:</p> <ol style="list-style-type: none"> 1. Hard links 2. Soft links
5.	make	utility for building and maintaining groups of programs.	The purpose of the make utility is to determine automatically which pieces of a large program need to be recompiled, and issue the commands to recompile

			<p>them. you can use make with any programming language whose compiler can be run with a shell command. In fact, make is not limited to programs. You can use it to describe any task where some files must be updated automatically from others whenever the others change.</p> <p>Options :</p> <ul style="list-style-type: none">-b,-m prints online help and exitThese options are ignored for compatibility with other versions of make.-B Unconditionally make all targets-d Print debugging information in addition to normal processing-e Give variables taken from the environment precedence over variables from makefiles-k Continue as much as possible after an error
--	--	--	--

Exercise

Configure and compilation the kernel

Step 1 :

Download the latest kernel source from www.kernel.org or from a repository.

Step 2 :

Check the current kernel version and name of the kernel.

step 3 :

Move the module from downloads to /usr/src and uzip the file.

step 4 :

Make a systemlink to the existing kernel and clean the existing kernel.

step 5 :

Building kernel and its modules.

step 6 :

Check the current kernel version and name of the kernel.

Output :

Kernel Version

```
ubuntu@ubuntu:/proc$ cat version
Linux version 5.4.0-45-generic (buildd@lgw01-amd64-033) (gcc version 9.
3.0 (Ubuntu 9.3.0-10ubuntu2)) #49-Ubuntu SMP Wed Aug 26 13:38:52 UTC 20
20
ubuntu@ubuntu:/proc$ █
```

Kernel Name

```
ubuntu@ubuntu:/proc$ uname -r
5.4.0-45-generic
ubuntu@ubuntu:/proc$ █
```

Moving the kernel to /usr/src

```
ubuntu@ubuntu:/usr/src$ sudo cp /home/ubuntu/Downloads/linux-5.8.7.tar.xz .
ubuntu@ubuntu:/usr/src$ ls
linux-5.8.7  linux-5.8.7.tar.xz  linux-headers-5.4.0-42  linux-headers-5.4.0-42-
-generic  linux-headers-5.4.0-45  linux-headers-5.4.0-45-generic
ubuntu@ubuntu:/usr/src$ █
```

Cleaning the kernel using make

```
ubuntu@ubuntu:/usr/src/linux-5.8.7$ sudo make clean
ubuntu@ubuntu:/usr/src/linux-5.8.7$ █
```

System link to existing kernel

```
ubuntu@ubuntu:~$ sudo ln -s /usr/src/linux-5.8.7 /usr/src/linux-headers-5.4.0-42-gener
ic/
ubuntu@ubuntu:~$ █
```

Making target files

```
ubuntu@ubuntu:/usr/src/linux-5.8.7$ sudo make menuconfig
LEX      scripts/kconfig/lexer.lex.c
YACC     scripts/kconfig/parser.tab.[ch]
HOSTCC   scripts/kconfig/lexer.lex.o
HOSTCC   scripts/kconfig/parser.tab.o
HOSTCC   scripts/kconfig/preprocess.o
HOSTCC   scripts/kconfig/symbol.o
HOSTCC   scripts/kconfig/util.o
HOSTLD   scripts/kconfig/mconf
scripts/kconfig/mconf Kconfig
#
# using defaults found in /boot/config-5.4.0-45-generic
#
/boot/config-5.4.0-45-generic:3815:warning: symbol value 'm' invalid for ISDN_CAPI
/boot/config-5.4.0-45-generic:8245:warning: symbol value 'm' invalid for ASHMEM
/boot/config-5.4.0-45-generic:9205:warning: symbol value 'm' invalid for ANDROID_BIND
ER_IPC
/boot/config-5.4.0-45-generic:9206:warning: symbol value 'm' invalid for ANDROID_BIND
ERFS
/boot/config-5.4.0-45-generic:9274:warning: symbol value 'm' invalid for INTERCONNECT

*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.
```

New kernel proof

```
ubuntu@ubuntu:/usr/src/linux-5.8.7$ sudo make
HOSTCC  scripts/kconfig/conf.o
HOSTLD  scripts/kconfig/conf
scripts/kconfig/conf --syncconfig Kconfig
SYSTBL  arch/x86/include/generated/asm/syscalls_32.h
SYSHDR  arch/x86/include/generated/asm/unistd_32_ia32.h
SYSHDR  arch/x86/include/generated/asm/unistd_64_x32.h
SYSTBL  arch/x86/include/generated/asm/syscalls_64.h
HYPERCALLS arch/x86/include/generated/asm/xen-hypercalls.h
SYSHDR  arch/x86/include/generated/uapi/asm/unistd_32.h
SYSHDR  arch/x86/include/generated/uapi/asm/unistd_64.h
SYSHDR  arch/x86/include/generated/uapi/asm/unistd_x32.h
HOSTCC  arch/x86/tools/relocs_32.o
HOSTCC  arch/x86/tools/relocs_64.o
HOSTCC  arch/x86/tools/relocs_common.o
HOSTLD  arch/x86/tools/relocs
HOSTCC  scripts/selinux/genheaders/genheaders
HOSTCC  scripts/selinux/mdp/mdp
HOSTCC  scripts/bin2c
HOSTCC  scripts/kallsyms
HOSTCC  scripts/recordmcount
HOSTCC  scripts/sorttable
HOSTCC  scripts asn1_compiler
HOSTCC  scripts/sign-file
```

Results:

The study and implement the kernel configuration, compilation and installation is studied and executed.

Video:

[https://drive.google.com/file/d/1n4owALSZJ_y9BsFhd7kRZhNEiCEFbLg/](https://drive.google.com/file/d/1n4owALSZJ_y9BsFhd7kRZhNEiCEFbLg/view)
view

Ex 3

COMPILING FROM THE SOURCE

Date: 29.08.20

Aim:

To study and implement the compiling from the source.

Description:

tar Command :

The Linux ‘tar’ stands for tape archive, is used to create Archive and extract the Archive files. tar command in Linux is one of the important commands which provides archiving functionality in Linux. We can use Linux tar command to create compressed or uncompressed Archive files and also maintain and modify them.

Syntax :

`tar [options] [archive-file] [file or directory to be archived]`

Options:

-c : Creates Archive

-x : Extract the archive

-f : creates archive with given filename

-t : displays or lists files in archive file

-u : archives and adds to an existing archive file

-v : Displays Verbose Information

-A : Concatenates the archive files

-z : zip, tells tar command that create tar file using gzip

-j : filter archive tar file using tbzip

-W : Verify a archive file

-r : update or add file or directory in already existed .tar file

zip Command :

ZIP is a compression and file packaging utility for Unix. Each file is stored in a single .zip { .zip-filename} file with the extension .zip.

- zip is used to compress the files to reduce file size and also used as file package utility. zip is available in many operating systems like unix, linux, windows etc.
- If you have a limited bandwidth between two servers and want to transfer the files faster, then zip the files and transfer.
- The zip program puts one or more compressed files into a single zip archive, along with information about the files (name, path, date, time of last modification, protection, and check information to verify file integrity). An entire directory structure can be packed into a zip archive with a single command.
- Compression ratios of 2:1 to 3:1 are common for text files. zip has one compression method (deflation) and can also store files without compression. zip automatically chooses the better of the two for each file to be compressed. The program is useful for packaging a set of files for distribution; for archiving files; and for saving disk space by temporarily compressing unused files or directories.

Syntax :

`zip [options] zipfile files_list`

Syntax for Creating a zip file:

`$zip myfile.zip filename.txt`

Options :

-d : Removes the file from the zip archive

-u : Updates the file in the zip archive

-m : Deletes the original files after zipping

-r : To zip a directory recursively

-x : Exclude the files in creating the zip

-v : Verbose mode or print diagnostic version info

gzip Command :

gzip command compresses files. Each single file is compressed into a single file. The compressed file consists of a GNU zip header and deflated data.

If given a file as an argument, gzip compresses the file, adds a “.gz” suffix, and deletes the original file. With no arguments, gzip compresses the standard input and writes the compressed file to standard output.

Difference between Gzip and zip command in Unix and when to use which command

- ZIP and GZIP are two very popular methods of compressing files, in order to save space, or to reduce the amount of time needed to transmit the files across the network, or internet.
- In general, GZIP is much better compared to ZIP, in terms of compression, especially when compressing a huge number of files.
- The common practice with GZIP, is to archive all the files into a single tarball before compression. In ZIP files, the individual files are compressed and then added to the archive.
- When you want to pull a single file from a ZIP, it is simply extracted, then decompressed. With GZIP, the whole file needs to be decompressed before you can extract the file you want from the archive.
- When pulling a 1MB file from a 10GB archive, it is quite clear that it would take a lot longer in GZIP, than in ZIP.
- GZIP’s disadvantage in how it operates, is also responsible for GZIP’s advantage. Since the compression algorithm in GZIP compresses one large file instead of multiple smaller ones, it can take advantage of the redundancy in the files to reduce the file size even further.
- If you archive and compress 10 identical files with ZIP and GZIP, the ZIP file would be over 10 times bigger than the resulting GZIP file.

Syntax :

gzip [Options] [filenames]

Options:

-f : Sometimes a file cannot be compressed

-k : By default when you compress a file using the “gzip” command you end up with a new file with the extension

-L : This option displays the gzip license

-r : This option can compress every file in a folder and its subfolders

-[1-9] : It allows to change the compression level

-v : his option displays the name and percentage reduction for each file compressed or decompressed

-d : This option allows you to decompress a file using the “gzip” command.

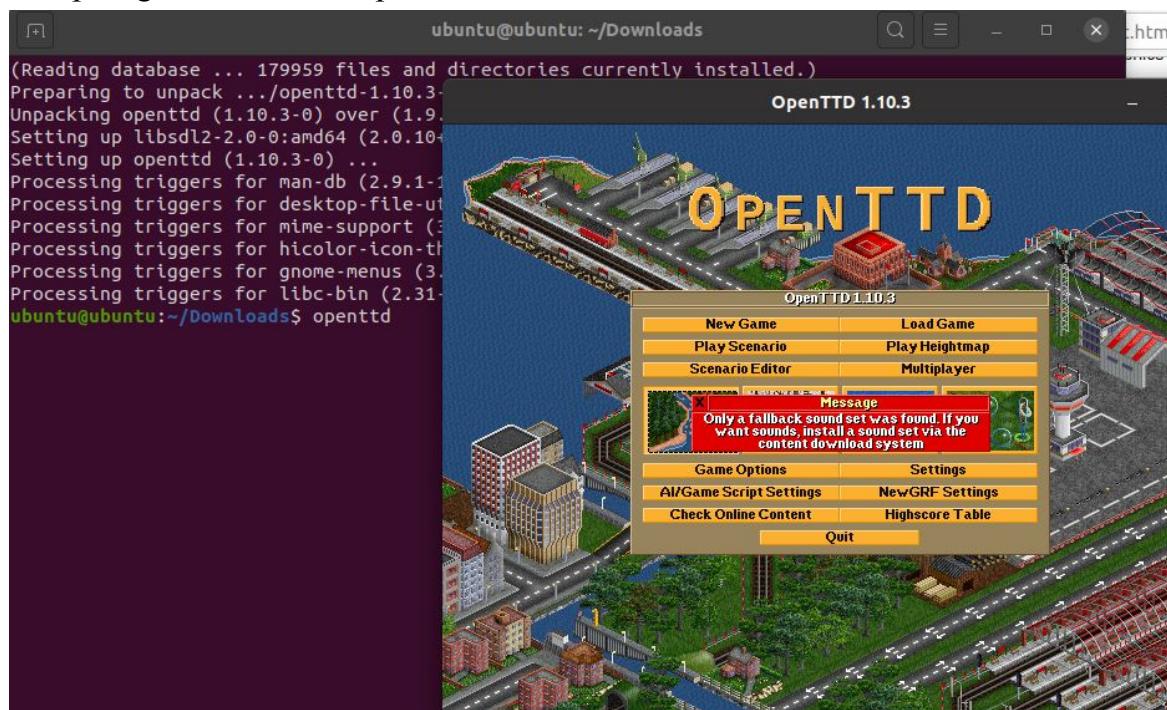
Exercise

1. Compile the source from openttd package

Installing openttd using dep(Debian) package

```
ubuntu@ubuntu:~/Downloads$ sudo apt install ./openttd-1.10.3-linux-ubuntu-focal-amd64.deb
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'openttd' instead of './openttd-1.10.3-linux-ubuntu-focal-amd64.deb'
The following packages were automatically installed and are no longer required:
  libsdl1.2debian libxdg-basedir1 openttd-openmsx
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libsdl2-2.0-0
Suggested packages:
  openttd-opensfx freepats
The following packages will be REMOVED:
  openttd-data
The following NEW packages will be installed:
  libsdl2-2.0-0
The following packages will be upgraded:
  openttd
1 upgraded, 1 newly installed, 1 to remove and 129 not upgraded.
Need to get 407 kB/5,266 kB of archives.
After this operation, 2,471 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 /home/ubuntu/Downloads/openttd-1.10.3-linux-ubuntu-focal-amd64.deb openttd amd64 1.10.3-0 [4,859 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 libsdl2-2.0-0 amd64 2.0.10+dfsg1-3 [407 kB]
Fetched 407 kB in 3s (152 kB/s)
Selecting previously unselected package libsdl2-2.0-0:amd64.
(Reading database ... 180064 files and directories currently installed.)
Preparing to unpack .../libsdl2-2.0-0_2.0.10+dfsg1-3_amd64.deb ...
Unpacking libsdl2-2.0-0:amd64 (2.0.10+dfsg1-3) ...
dpkg: openttd-data: dependency problems, but removing anyway as you requested:
  openttd depends on openttd-data (= 1.9.3-1build2).
```

Compiling the source of openttd



2. Compile the source from JFrog package

Installing conan c++ package using dep(Debian) package

```
ubuntu@ubuntu:~/Downloads$ sudo apt install ./conan-ubuntu-64_1_29_0.deb
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'conan' instead of './conan-ubuntu-64_1_29_0.deb'
The following packages were automatically installed and are no longer required:
  libsdl1.2debian libxdg-basedir1 openttd-openmsx
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev fakeroot g++ g++-9
  gcc gcc-9 libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5
  libatomic1 libbinutils libc-dev-bin libc6-dev libcrypt-dev libctf-nobfd0 libctf0 libfakeroot
  libgcc-9-dev libitm1 liblsan0 libquadmath0 libstdc++-9-dev libtsan0 libubsan1 linux-libc-dev
  make manpages-dev
Suggested packages:
  binutils-doc debian-keyring g++-multilib g++-9-multilib gcc-9-doc gcc-multilib autoconf
  automake libtool flex bison gcc-doc gcc-9-multilib gcc-9-locales glibc-doc libstdc++-9-doc
  make-doc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential conan dpkg-dev fakeroot g++
  g++-9 gcc gcc-9 libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl
  libasan5 libatomic1 libbinutils libc-dev-bin libc6-dev libcrypt-dev libctf-nobfd0 libctf0
  libfakeroot libgcc-9-dev libitm1 liblsan0 libquadmath0 libstdc++-9-dev libtsan0 libubsan1
  linux-libc-dev make manpages-dev
0 upgraded, 33 newly installed, 0 to remove and 129 not upgraded.
Need to get 31.4 MB/47.4 MB of archives.
After this operation, 166 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

creating a sample package

```
ubuntu@ubuntu:~/Desktop/open source$ conan new Ex3/0.1
[File saved: conanfile.py]
ubuntu@ubuntu:~/Desktop/open source$ mkdir src
ubuntu@ubuntu:~/Desktop/open source$ cd src
ubuntu@ubuntu:~/Desktop/open source/src$ touch MyLib.h
ubuntu@ubuntu:~/Desktop/open source/src$ touch MyLib.cpp
ubuntu@ubuntu:~/Desktop/open source/src$ ls
MyLib.cpp  MyLib.h
ubuntu@ubuntu:~/Desktop/open source/src$ gedit MyLib.h
ubuntu@ubuntu:~/Desktop/open source/src$ gedit MyLib.cpp
ubuntu@ubuntu:~/Desktop/open source/src$ gedit MyLib.h
ubuntu@ubuntu:~/Desktop/open source/src$ touch CMakeLists.txt
ubuntu@ubuntu:~/Desktop/open source/src$ gedit CMakeLists.txt
ubuntu@ubuntu:~/Desktop/open source/src$ mkdir build
ubuntu@ubuntu:~/Desktop/open source/src$ cd build
ubuntu@ubuntu:~/Desktop/open source/src$ gedit main.cpp
```

Making up the package

```
ubuntu@ubuntu:~/Desktop/open source/src$ cmake .. /src/
-- The C compiler identification is GNU 9.3.0
-- The CXX compiler identification is GNU 9.3.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Configuring done
-- Generating done
-- Build files have been written to: /home/ubuntu/Desktop/open source/src
```

Build package output

```
ubuntu@ubuntu:~/Desktop/open source/src$ conan create . myself/MyLib
Exporting package recipe
MyLib/0.1@myself/MyLib: A new conanfile.py version was exported
MyLib/0.1@myself/MyLib: Folder: /home/ubuntu/.conan/data/MyLib/0.1/myself/MyLib/export
MyLib/0.1@myself/MyLib: Package recipe modified in export, forcing source folder removal
MyLib/0.1@myself/MyLib: Use the --keep-source, -k option to skip it
MyLib/0.1@myself/MyLib: Exported revision: e8aa540f83f3827d3af317087a2ac665
Configuration:
[settings]
arch=x86_64
arch_build=x86_64
build_type=Release
compiler=gcc
compiler.libcxx=libstdc++
compiler.version=9
os=linux
os_build=linux
[options]
[build_requires]
[env]

MyLib/0.1@myself/MyLib: Forced build from source
Installing package: MyLib/0.1@myself/MyLib
Requirements
    MyLib/0.1@myself/MyLib from local cache - Cache
Packages
    MyLib/0.1@myself/MyLib:2a30b7d6ea2202e5393ebda51c8729dd2162b9f8 - Build
```

3. Compile the source from gcc

Checking the version of gcc

```
ubuntu@ubuntu:~$ gcc --version
gcc (Ubuntu 9.3.0-10ubuntu2) 9.3.0
Copyright (C) 2019 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

Program :

```
#include <stdio.h>
int main() {
    printf("Welcome to Open Source Lab \n");
    return 0;
}
```

Output :

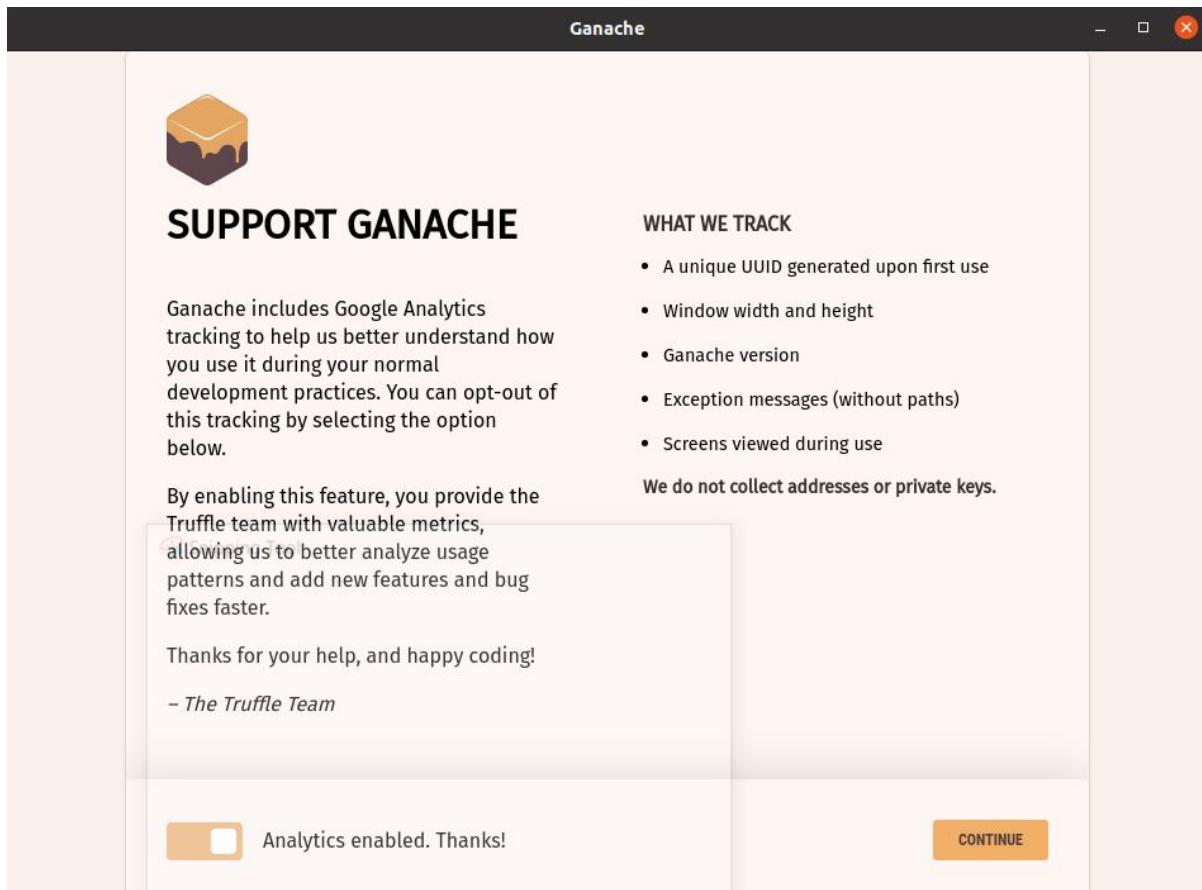
```
ubuntu@ubuntu:~$ gedit hello.c
ubuntu@ubuntu:~$ gcc -o hello hello.c
ubuntu@ubuntu:~$ ./hello
Welcome to Open Source Lab
ubuntu@ubuntu:~$
```

4. Compile the source from any open source package

Installing ganache(open source package) using AppImage

```
ubuntu@ubuntu:~/Downloads$ sudo chmod a+x ganache-2.4.0-linux-x86_64.AppImage
[sudo] password for ubuntu:
ubuntu@ubuntu:~/Downloads$ sudo cp ganache-2.4.0-linux-x86_64.AppImage /usr/bin
ubuntu@ubuntu:~/Downloads$ ./ganache-2.4.0-linux-x86_64.AppImage
12:41:13.025 > Checking for update
12:41:13.378 > Generated new staging user ID: f8615ccc-cfcf-5663-93b5-6f96070461c5
12:41:16.956 > Update for version 2.4.0 is not available (latest version: 2.4.0, downgrade is disallowed).
```

Compiling an open source package



Results:

The compiling from the source is studied and executed.

Video:

https://drive.google.com/file/d/1uxj3u9YuxIMRk1qjmn_RVu5gWcYY66I7/view

Ex 4

VIRTUALIZATION

Date: 01.09.2020

Aim:

To study and implement the installation of FreeDOS and overlay the emulator.

1. Install QEMU emulator and run FreeDOS in it.
2. Install any open-source virtualization application and run FreeDOS in it.

Description:

FreeDOS:

DOS is a "disk operating system" created when personal computers ran from floppy disks. Even when computers supported hard drives, it was common in the 1980s and 1990s to switch frequently between the different drives. For example, you might make a backup copy of your most important files to a floppy disk.

FreeDOS is an old operating system, but it is new to many people. In 1994, several developers and I came together to create FreeDOS—a complete, free, DOS-compatible operating system you can use to play classic DOS games, run legacy business software, or develop embedded systems. Any program that works on MS-DOS should also run on FreeDOS.

QEMU:

QEMU is a generic and open source machine & userspace emulator and virtualizer. In computing, an emulator is a hardware or software that enables one computer system (called the host) to behave like another computer system (called the guest).

In 1963, when microcode was first used to speed up this simulation process, IBM engineers coined the term “emulator” to describe the concept. In the 2000s, it has become common to use the word “emulate” in the context of software. However, before 1980, “emulation” referred only to emulation with a hardware or microcode assist, while “simulation” referred to pure software emulation.

Purists continue to insist on this distinction, but currently, the term “emulation” often means the complete imitation of a machine executing binary code while “simulation” often refers to computer simulation, where a computer program is used to simulate an abstract model.

VirtualBox:

VirtualBox is a powerful x86 and AMD64/Intel64 virtualization product for enterprise as well as home use. Not only is VirtualBox an extremely feature-rich, high-performance product for enterprise customers, it is also the only professional solution that is freely available as Open Source Software under the terms of the GNU General Public License (GPL) version 2.

Presently, VirtualBox runs on Windows, Linux, Macintosh, and Solaris hosts and supports a large number of guest operating systems including but not limited to Windows (NT 4.0, 2000, XP, Server 2003, Vista, Windows 7, Windows 8, Windows 10), DOS/Windows 3.x, Linux (2.4, 2.6, 3.x and 4.x), Solaris and OpenSolaris, OS/2, and OpenBSD.

VirtualBox is being actively developed with frequent releases and has an ever-growing list of features, supported guest operating systems, and platforms it runs on. VirtualBox is a community effort backed by a dedicated company: everyone is encouraged to contribute while Oracle ensures the product always meets professional quality criteria.

Exercise:

1. Install QEMU emulator and run FreeDOS in it.

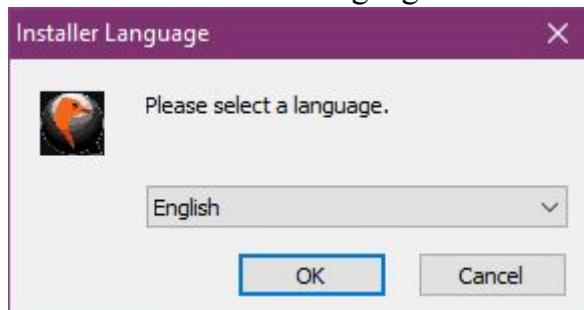
Download QEMU file from its official site

QEMU Binaries for Windows (64 bit)

Here you get QEMU related binaries for 64 bit versions of Microsoft Windows.

Name	Last modified	Size	Description
 Parent Directory		-	
 2011/	2016-04-15 10:48	-	experimental QEMU for Windows
 2012/	2016-04-15 10:48	-	experimental QEMU for Windows
 2013/	2016-04-15 10:48	-	experimental QEMU for Windows
 2014/	2016-04-15 10:48	-	experimental QEMU for Windows
 2015/	2016-04-15 10:47	-	experimental QEMU for Windows
 2016/	2017-02-19 08:41	-	experimental QEMU for Windows
 2017/	2018-03-21 21:24	-	experimental QEMU for Windows
 2018/	2018-11-28 15:44	-	experimental QEMU for Windows
 2019/	2020-02-01 15:42	-	experimental QEMU for Windows
 2020/	2020-11-19 20:12	-	experimental QEMU for Windows
 old/	2018-11-17 20:09	-	
 qemu-w64-setup-20200814.exe	2020-08-14 11:52	140M	QEMU Installer for Windows (64 bit)
 qemu-w64-setup-20200814.sha512	2020-08-14 11:53	158	SHA-512 for installer
 qemu-w64-setup-20201119.exe	2020-11-19 23:44	185M	QEMU Installer for Windows (64 bit)
 qemu-w64-setup-20201119.sha512	2020-11-19 23:44	158	SHA-512 for installer

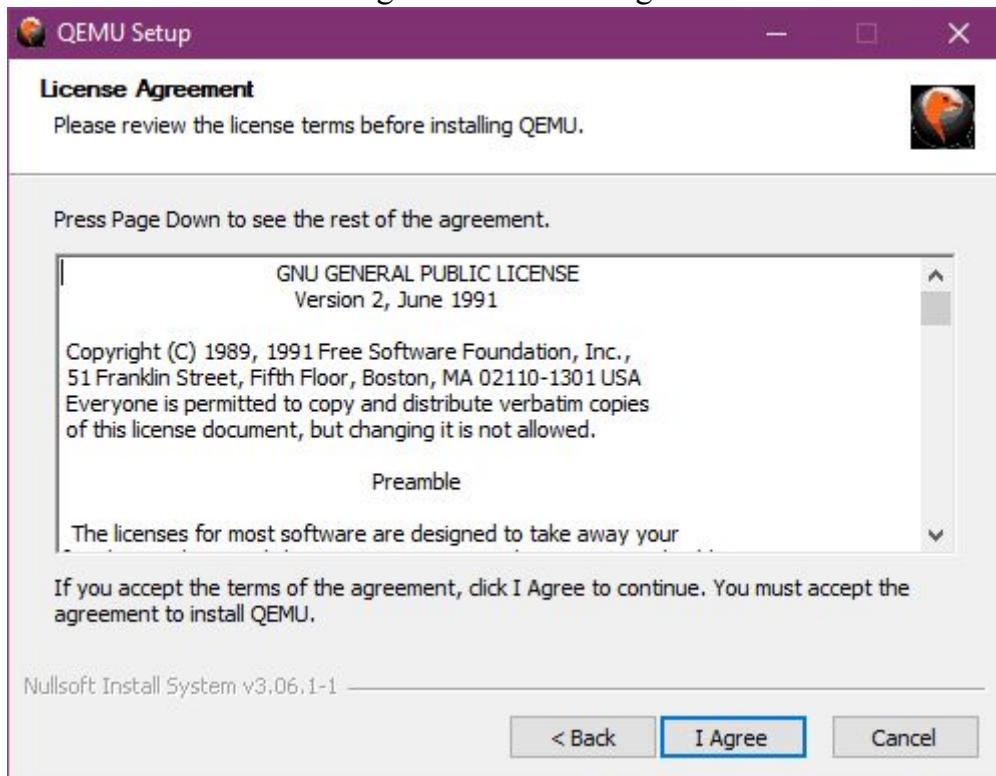
Select the language



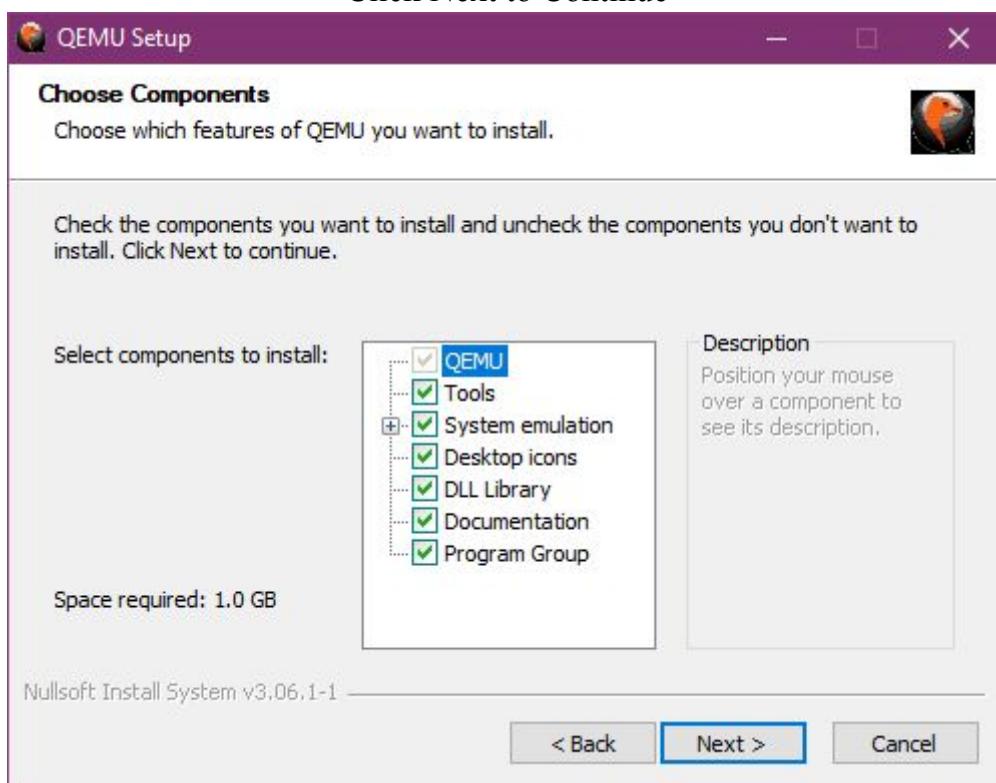
Click Next to Continue

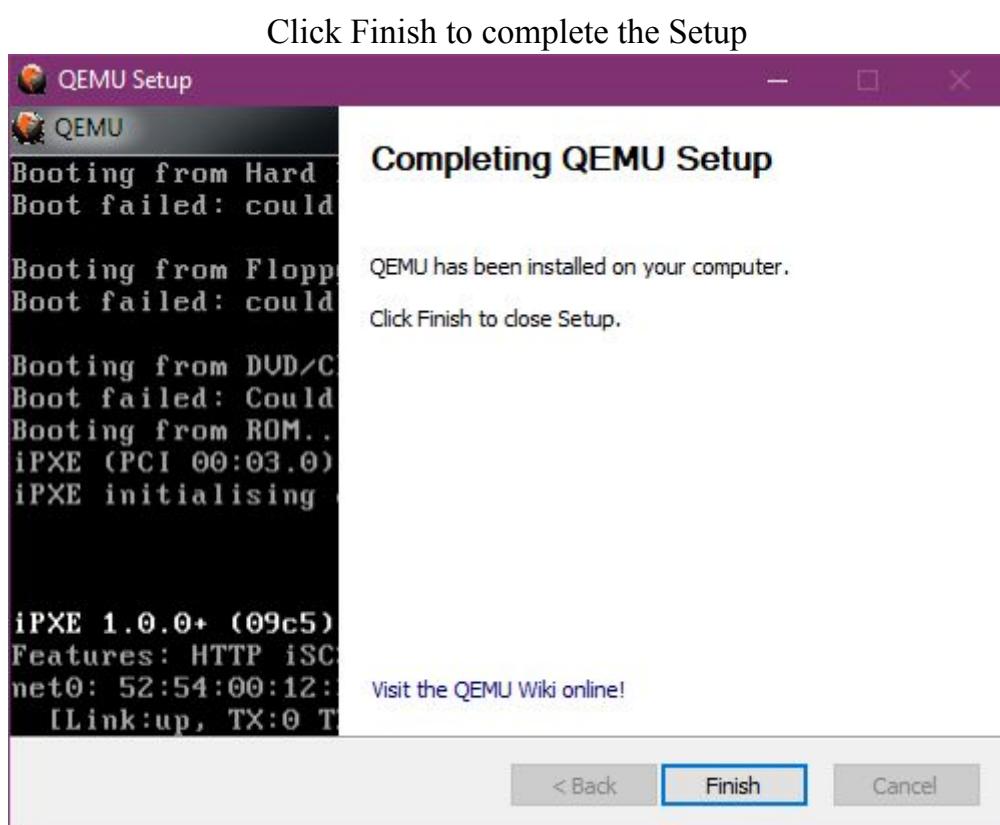
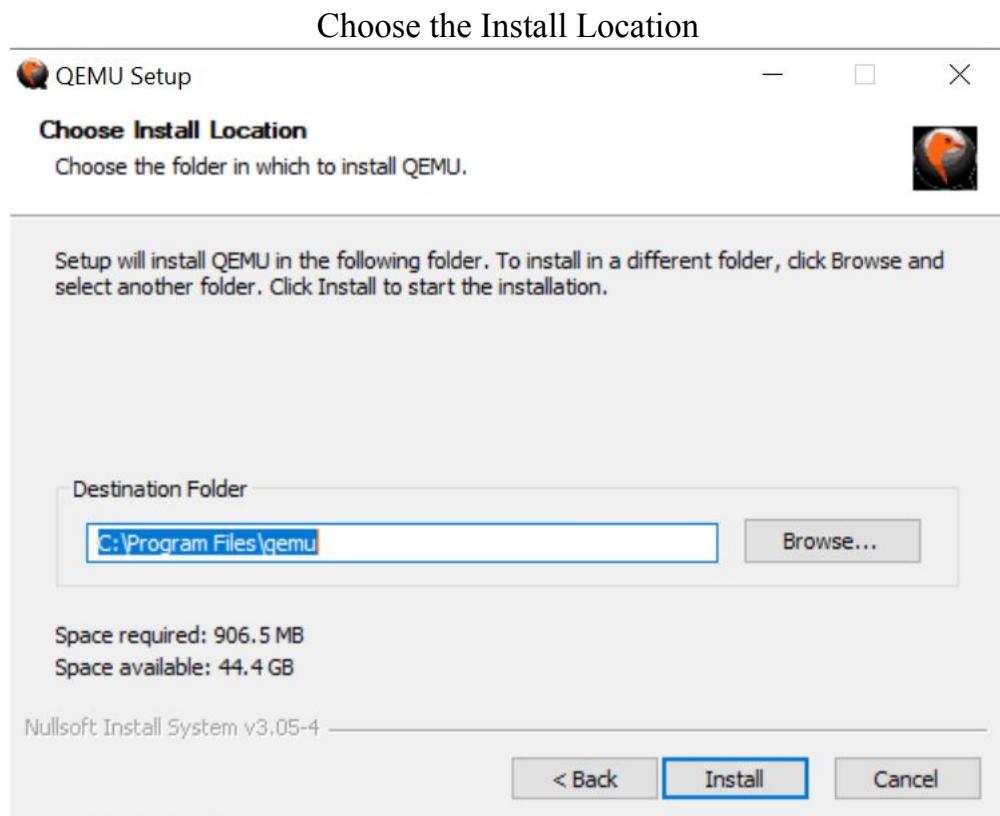


Click I Agree for License Agreement



Click Next to Continue





Open the PowerShell and Locate to the ISO Folder

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Gloria> cd C:/
PS C:\> dir

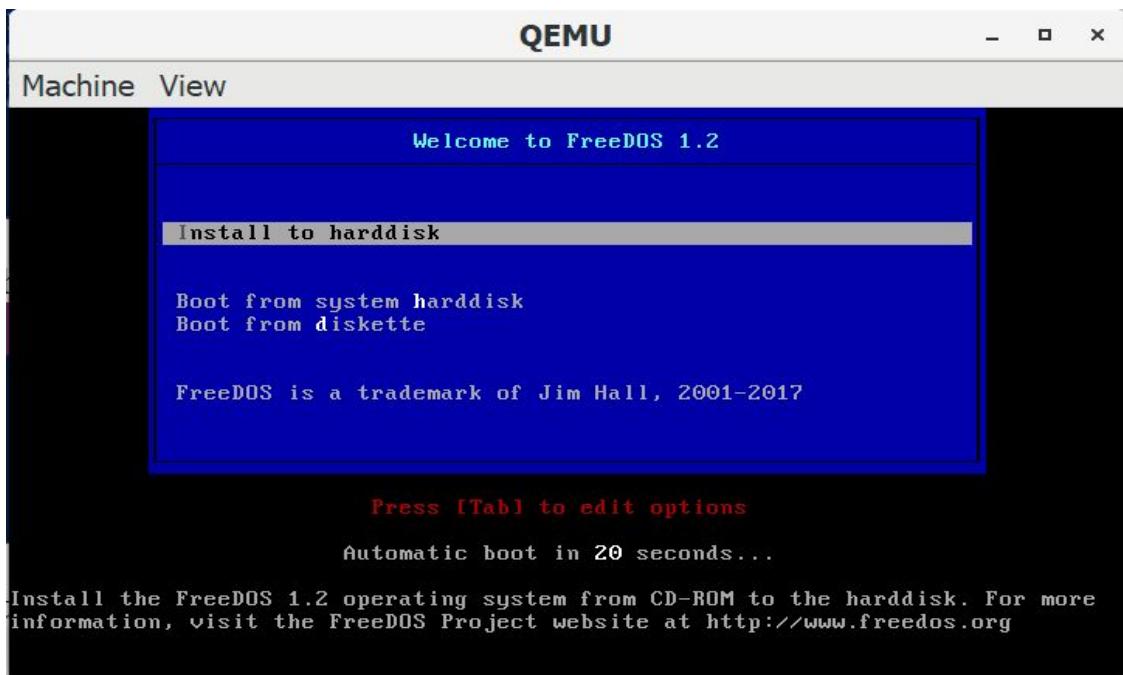
    Directory: C:\

Mode                LastWriteTime         Length Name
----                -----         -----
d----        12-02-2020      15:20             ESD
d----        12-09-2018      12:18             Intel
d----        07-12-2019      14:44             PerfLogs
d-r----       21-11-2020      12:29             Program Files
d-r----       05-11-2020      11:38             Program Files (x86)
d----        06-06-2020      22:27             SWSetup
d-r----       04-11-2020      22:20             Users
d----        18-11-2020      02:56             Windows
d----        03-03-2019      17:48             xampp
-a----       21-11-2020     438777856 FD12CD.iso
```

Type the Command To Run the FreeDOS

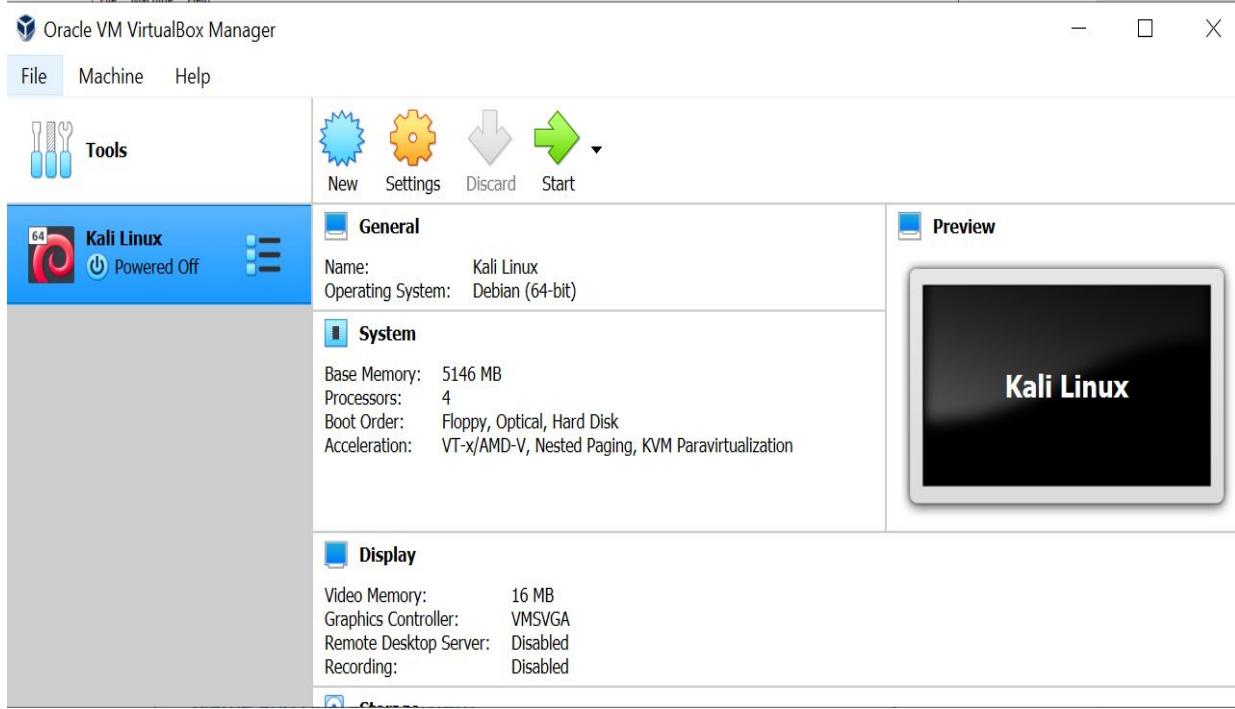
```
Select Windows PowerShell
PS C:\> qemu-system-x86_64.exe -boot d -cdrom FD12CD.iso -m 512
```

Finish the Installation Process of FreeDOS



2. Install any open-source virtualization application and run FreeDOS in it.

Download VirtualBox and run it And Click New



Give Name as FreeDOS and Version as DOS

← Create Virtual Machine

Name and operating system

Please choose a descriptive name and destination folder for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

Name:

Machine Folder:

Type: 

Version:

Select the Memory Size

?

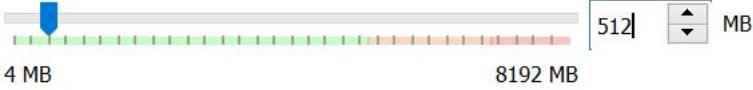
X

← Create Virtual Machine

Memory size

Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **32 MB**.



A horizontal slider with a blue arrow pointing right, indicating the current value of 512 MB. Below the slider, the range is labeled from 4 MB to 8192 MB. An input field shows the value 512 MB with up and down arrows for adjustment.

Next Cancel

Select the Hard Disk

?

X

← Create Virtual Machine

Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select one from the list or from another location using the folder icon.

If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

The recommended size of the hard disk is **500.00 MB**.

- Do not add a virtual hard disk
- Create a virtual hard disk now
- Use an existing virtual hard disk file

Kali Linux.vdi (Normal, 102.91 GB)

Create Cancel

Select the Hard Disk file type

? X

← Create Virtual Hard Disk

Hard disk file type

Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

- VDI (VirtualBox Disk Image)
- VHD (Virtual Hard Disk)
- VMDK (Virtual Machine Disk)

Expert Mode Next Cancel

Select the language

? X

← Create Virtual Hard Disk

Storage on physical hard disk

Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

- Dynamically allocated
- Fixed size

Next Cancel

Select the File Location and Size

← Create Virtual Hard Disk

File location and size

Please type the name of the new virtual hard disk file into the box below or click on the folder icon to select a different folder to create the file in.

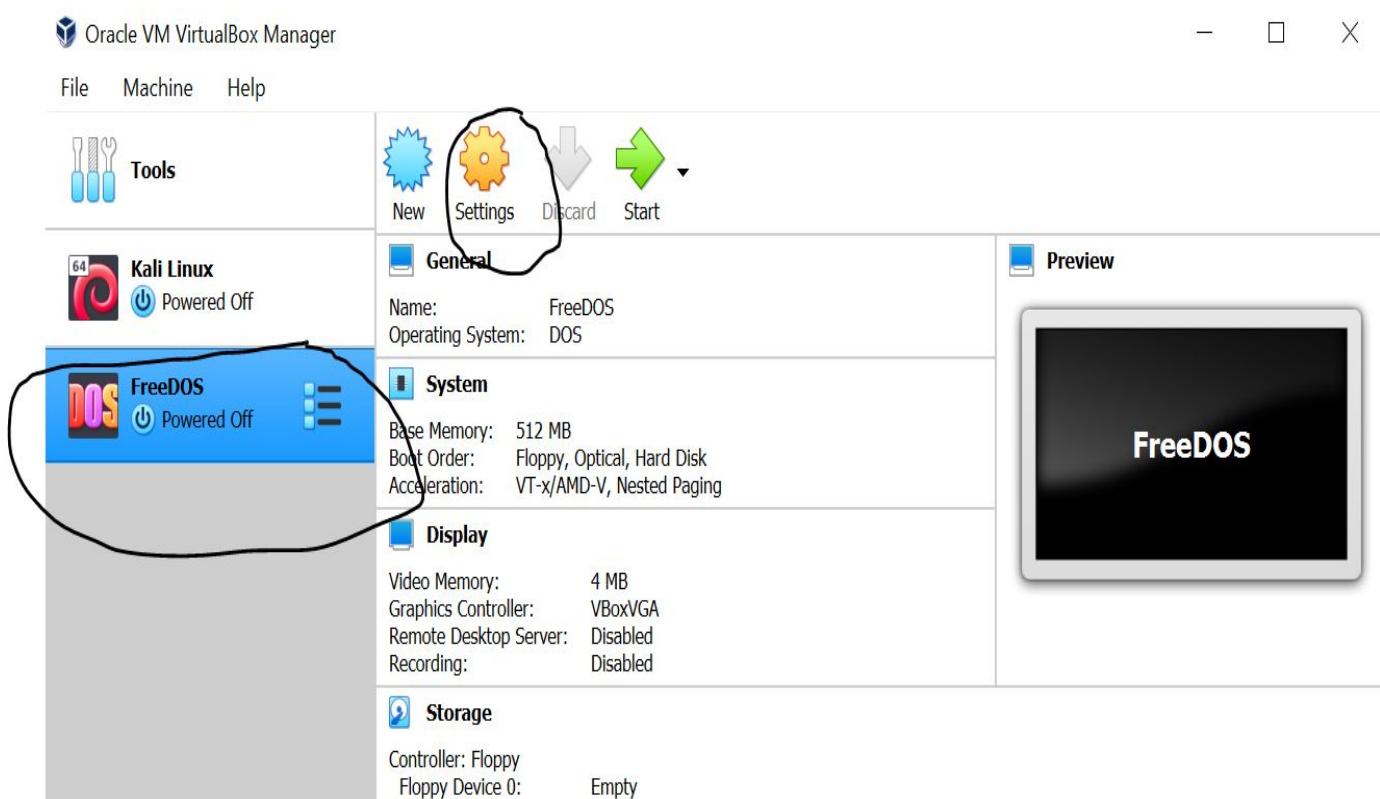
C:\Users\Sandy\VirtualBox VMs\FreeDOS\FreeDOS.vdi 

Select the size of the virtual hard disk in megabytes. This size is the limit on the amount of file data that a virtual machine will be able to store on the hard disk.

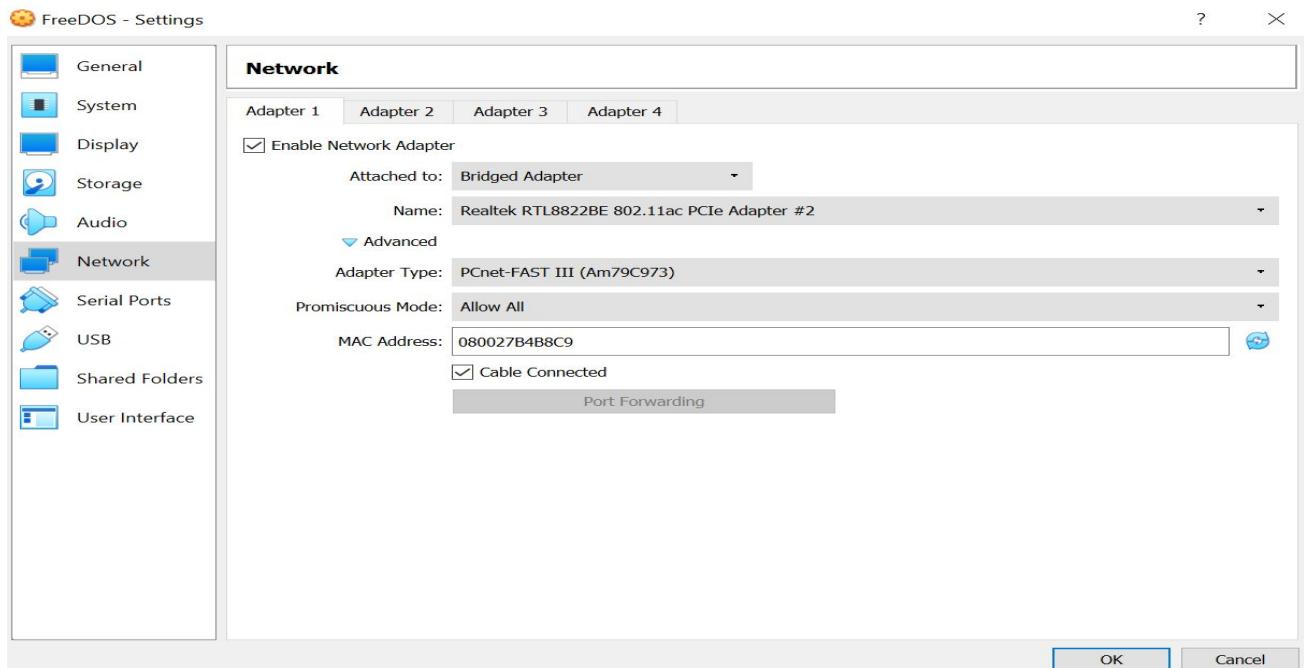
 2000.00 MB
4.00 MB 2.00 TB

Create Cancel

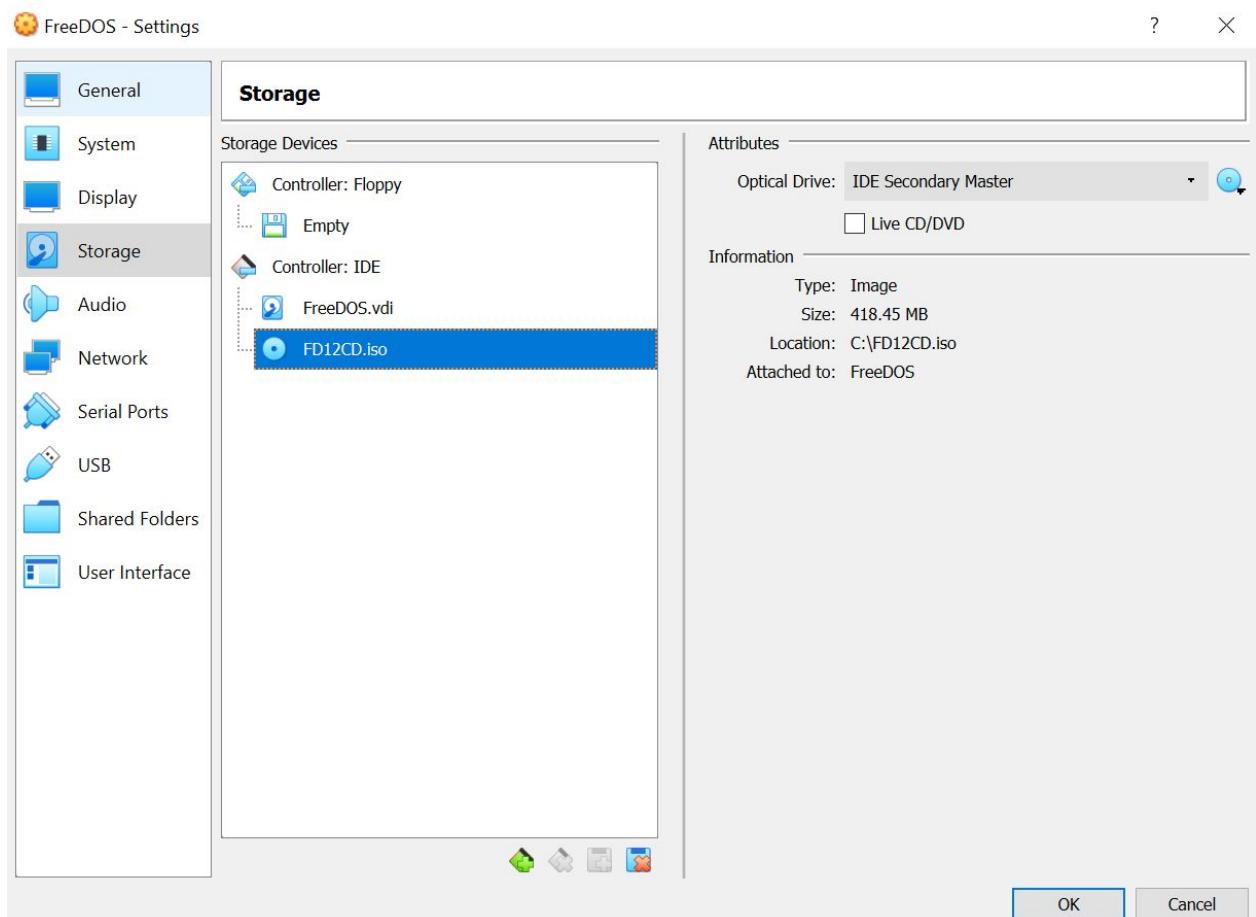
Select the Settings to Configure



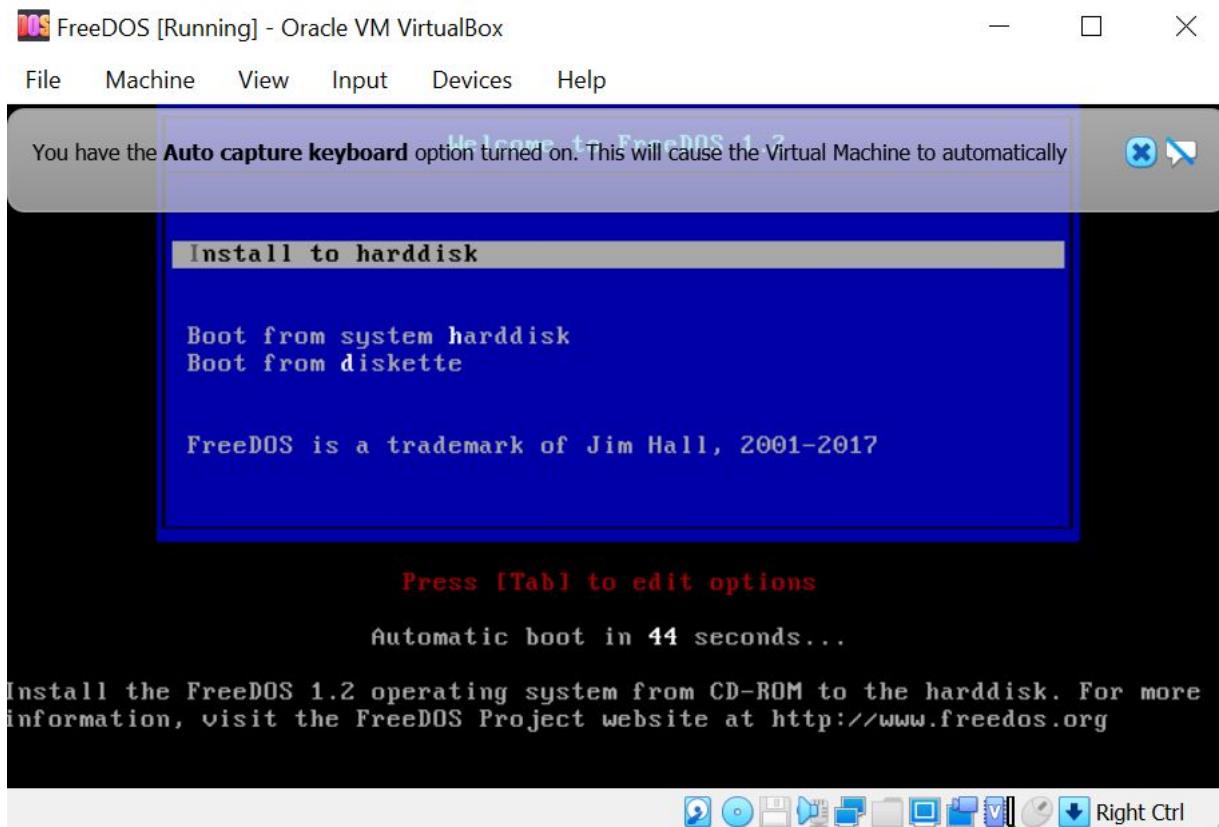
Go to Network and Change to Bridged Adapter and give Allow all



Go to Storage and Select the FreeDOS ISO file



Finish the Installation Process of FreeDOS



Results:

The package management system is studied and executed.

Video Link: <https://youtu.be/OBzW-jOVoBc>

Ex 5 Date: 29.08.20

Package Management System

Aim:

To study and implement the package management system

Description:

Samba :

Samba is a free and open-source re-implementation of the SMB/CIFS network file sharing protocol that allows end-users to access files, printers, and other shared resources.

This tutorial explains how to install Samba on Ubuntu 18.04 and configure it as a standalone server to provide file sharing across different operating systems over a network.

We'll create the following Samba shares and users.

Users:

- **sadmin** - An administrative user with read and write access to all shares.
- **josh** - A regular user with its own private file share

Shares:

- **users** - This share will be accessible with read/write permissions by all users.
- **josh** - This share will be accessible with read/write permissions only by users josh and sadmin.

The file shares will be accessible from all devices on your network. Later in this tutorial, we will also provide detailed instructions on how to connect to the Samba server from Linux, Windows and macOS clients.

Exercise

1. Install (Samba) free software re-implementation of the SMB/CIFS networking protocol.

Installing samba using terminal

```
ubuntu@ubuntu:~$ sudo apt install samba
[sudo] password for ubuntu:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libsdl1.2debian libxdg-basedir1 openttd-openmsx
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  attr ibverbs-providers libcephfs2 libibverbs1 librados2 librdmacm1
  libsmclient libwbclient0 python3-crypto python3-dnspython python3-gpg
  python3-ldb python3-markdown python3-packaging python3-pgments
  python3-pyparsing python3-samba python3-tdb samba-common samba-common-bin
  samba-dsdb-modules samba-libs samba-vfs-modules tdb-tools
Suggested packages:
  python-markdown-doc python-pgments-doc ttf-bitstream-vera
  python-pyparsing-doc bind9 bind9utils ctdb ldb-tools ntp | chrony
  smbldap-tools winbind heimdal-clients
The following NEW packages will be installed:
  attr ibverbs-providers libcephfs2 libibverbs1 librados2 librdmacm1
  python3-crypto python3-dnspython python3-gpg python3-ldb python3-markdown
  python3-packaging python3-pgments python3-pyparsing python3-samba
  python3-tdb samba samba-common samba-common-bin samba-dsdb-modules
```

Checking the status of Samba service

```
ubuntu@ubuntu:~$ sudo systemctl status smbd
● smbd.service - Samba SMB Daemon
   Loaded: loaded (/lib/systemd/system/smbd.service; enabled; vendor >
   Active: active (running) since Tue 2020-09-15 10:48:53 PDT; 54s ago
     Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
   Main PID: 3332 (smbd)
      Status: "smbd: ready to serve connections..."
        Tasks: 4 (limit: 2285)
       Memory: 10.0M
      CGroup: /system.slice/smbd.service
              ├─3332 /usr/sbin/smbd --foreground --no-process-group
              ├─3334 /usr/sbin/smbd --foreground --no-process-group
              ├─3335 /usr/sbin/smbd --foreground --no-process-group
              └─3337 /usr/sbin/smbd --foreground --no-process-group

Sep 15 10:48:53 ubuntu systemd[1]: Starting Samba SMB Daemon...
Sep 15 10:48:53 ubuntu update-apparmor-samba-profile[3326]: grep: /etc/>
Sep 15 10:48:53 ubuntu update-apparmor-samba-profile[3329]: diff: /etc/>
Sep 15 10:48:53 ubuntu systemd[1]: Started Samba SMB Daemon.
```

Starting Samba using start smbd

```
ubuntu@ubuntu:~$ sudo systemctl start smbd
ubuntu@ubuntu:~$ sudo systemctl status smbd
● smbd.service - Samba SMB Daemon
   Loaded: loaded (/lib/systemd/system/smbd.service; enabled; vendor preset: ▶
   Active: active (running) since Tue 2020-09-15 10:48:53 PDT; 3min 31s ago
     Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
   Main PID: 3332 (smbd)
      Status: "smbd: ready to serve connections..."
        Tasks: 4 (limit: 2285)
       Memory: 10.0M
      CGroup: /system.slice/smbd.service
              └─3332 /usr/sbin/smbd --foreground --no-process-group
                  ├─3334 /usr/sbin/smbd --foreground --no-process-group
                  ├─3335 /usr/sbin/smbd --foreground --no-process-group
                  ├─3337 /usr/sbin/smbd --foreground --no-process-group

Sep 15 10:48:53 ubuntu systemd[1]: Starting Samba SMB Daemon...
Sep 15 10:48:53 ubuntu update-apparmor-samba-profile[3326]: grep: /etc/apparmor>
Sep 15 10:48:53 ubuntu update-apparmor-samba-profile[3329]: diff: /etc/apparmor>
Sep 15 10:48:53 ubuntu systemd[1]: Started Samba SMB Daemon.
lines 1-20/20 (END)
```

Restart Samba using restart smbd

```
ubuntu@ubuntu:~$ sudo systemctl restart smbd
ubuntu@ubuntu:~$ sudo systemctl restart nmbd
```

Allow Samba service from firewall

```
ubuntu@ubuntu:~$ sudo ufw allow 'Samba'
Rules updated
Rules updated (v6)
```

Enforce Permissive State

```
ubuntu@ubuntu:~$ getenforce
Disabled
```

```
ubuntu@ubuntu:~$ setenforce
usage: setenforce [ Enforcing | Permissive | 1 | 0 ]
ubuntu@ubuntu:~$ setenforce 0
setenforce: SELinux is disabled
ubuntu@ubuntu:~$
```

Creating a normal user named Josh and setting up the Password

```
ubuntu@ubuntu:~$ sudo systemctl restart smbd
ubuntu@ubuntu:~$ sudo systemctl restart nmbd
ubuntu@ubuntu:~$ sudo mkdir /samba
ubuntu@ubuntu:~$ sudo chgrp sambashare /samba
ubuntu@ubuntu:~$ sudo useradd -M -d /samba/josh -s /usr/sbin/nologin -G sambashare josh
ubuntu@ubuntu:~$ sudo mkdir /samba/josh
ubuntu@ubuntu:~$ sudo chown josh:sambashare /samba/josh
ubuntu@ubuntu:~$ sudo chmod 2770 /samba/josh
ubuntu@ubuntu:~$ sudo smbpasswd -a josh
New SMB password:
Retype new SMB password:
Added user josh.
ubuntu@ubuntu:~$ sudo smbpasswd -e josh
Enabled user josh.
ubuntu@ubuntu:~$
```

Installing smbclient from terminal to access localhost

```
ubuntu@ubuntu:~$ sudo apt install smbclient
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libssl1.2debian libxdg-basedir1 openttd-openmsx
Use 'sudo apt autoremove' to remove them.
Suggested packages:
  cifs-utils heimdal-clients
The following NEW packages will be installed:
  smbclient
0 upgraded, 1 newly installed, 0 to remove and 133 not upgraded.
Need to get 365 kB of archives.
After this operation, 1,857 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 smbclient amd64 2:4.11.6+dfsg-0ubuntu1.4 [365 kB]
Fetched 365 kB in 7s (51.3 kB/s)
Selecting previously unselected package smbclient.
```

Login into smbclient and listing number of documents

```
akpkvk@akpkvk-VirtualBox:~$ smbclient //localhost/anil -U anil
Enter WORKGROUP\anil's password:
```

Results:

The package management system is studied and executed.

Video : <https://youtu.be/tvozIc2KECA>

Ex 6

REPOSITORY IN GITHUB

Date: 25.08.20

Aim:

To study and implement the git and git repository.

Description:

The Linux Kernel :

GitHub is a Git repository hosting service, but it adds many of its own features. While Git is a command line tool, **GitHub** provides a Web-based graphical interface. It also provides access control and several collaboration features, such as a wikis and basic task management tools for every project.

GitHub is a website for developers and programmers to collaboratively work on code. The primary benefit of **GitHub** is its version control system, which allows for seamless collaboration without compromising the integrity of the original project.

COMMANDS:

Sl. No.	COMMAND	DESCRIPTION
1.	git init	Initialize git repository
2.	git add filename	To add individual files
3.	git add *	To add all files
4.	git status	To see the added files status
5.	git commit -m "text"	Added text note with added file
6.	git remote add origin path	To link the files to the github repository
7.	git push -u origin master	To push files into the repository

Exercise

Create repository and uploading files in github

Step 1 :

Installing git

```
kali㉿kali:~$ sudo apt-get install git
[sudo] password for kali:
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version (1:2.27.0-1).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
kali㉿kali:~$
```

Step 2 :

Check version

```
kali㉿kali:~$ git --version
git version 2.27.0
kali㉿kali:~$
```

Step 3 :

Initializing git

```
kali㉿kali:~/Desktop/git$ git init
Reinitialized existing Git repository in /home/kali/Desktop/git/.git/
kali㉿kali:~/Desktop/git$
```

Step 4: checking Repository

The screenshot shows a GitHub repository page for the user 'sandy970' named 'opensourcelab'. The repository has 1 branch ('main') and 0 tags. A recent commit by 'sandy970' titled 'Add files via upload' was made 6 minutes ago. The commit details show four files added: 'README.md', 'URK17CS023_EXP2 (1).pdf', 'URK17CS023_LAB 1.pdf', and 'URK17CS023_LABEXP3 (1).pdf'. Below the commit history, there is a preview of the 'README.md' file which contains the text 'opensourcelab'.

Github link :

<https://github.com/sandy970/opensourcelab>

Results:

The study and implementation of the git and git repository is studied and executed.

Video link : <https://youtu.be/Fc9eMqzVzzs>

Ex 7

Basic Python Programming

Date: 06.10.2020

Aim:

To study and implement the basic python programming.

Exercise

1. Design and implement a basic calculator.

Source Code :

```
a = int(input("enter the first number :"))
b = int(input("enter the second number :"))
c = int(input("enter the operation you want :\n 1.Add \n 2.Sub \n 3.Mul \n 4.Div \n"))
if c==1:
    print(a+b)
elif c==2:
    print(a-b)
elif c==3:
    print(a*b)
elif c==4:
    print(a/b)
else:
    print("choose the correct operation _ ")
```

Output :

```
enter the first number :12
enter the second number :34
enter the operation you want :
 1.Add
 2.Sub
 3.Mul
 4.Div
1
46
```

```
enter the first number :42
enter the second number :22
enter the operation you want :
1.Add
2.Sub
3.Mul
4.Div
2
20
```

```
enter the first number :45
enter the second number :15
enter the operation you want :
1.Add
2.Sub
3.Mul
4.Div
4
3.0
```

2. Find whether the given number is perfect or not.

Source Code :

```
Number = int(input(" Please Enter any Number: "))
Sum = 0
for i in range(1, Number):
    if(Number % i == 0):
        Sum = Sum + i
if (Sum == Number):
    print("%d is a Perfect Number" %Number)
else:
    print("%d is not a Perfect Number" %Number)
```

Output :

```
Please Enter any Number: 6
6 is a Perfect Number
> |
```

3. Find whether the given number is Adam's number or not.

Source Code :

```
def isAdam(num):
    n = num
    rev = 0
    while n != 0:
        rev = rev * 10 + n % 10
        n = n // 10
    sn = num ** 2
    sr = rev ** 2
    n = sr
    rev = 0
    while n != 0:
        rev = rev * 10 + n % 10
        n = n // 10
    return sn == rev

num = int(input("Enter the number: "))
if isAdam(num):
    print(str(num) + " is an Adam Number.")
else:
    print(str(num) + " is NOT an Adam Number.")
```

Output :

```
Enter the number: 234
234 is NOT an Adam Number.
```

```
Enter the number: 13
13 is an Adam Number.
```

4. Write a program to check whether the given number is Armstrong or not.**Source Code :**

```
num = int(input("Enter a number: "))
sum = 0
temp = num

while temp > 0:
    digit = temp % 10
```

```
sum += digit ** 3
temp //= 10

if num == sum:
    print(num,"is an Armstrong number")
else:
    print(num,"is not an Armstrong number")
```

Output :

```
Enter a number: 125
125 is not an Armstrong number
```

```
Enter a number: 153
153 is an Armstrong number
```

Results:

The study and implementation of the basic python programming are studied and executed.

Video : https://youtu.be/_8995DFtF-c

Ex 8

Basic Perl Programming

Date: 03.11.2020

Aim:

To study and implement the basic Perl programming.

Description

Perl is a programming language developed by Larry Wall, specially designed for text processing. It stands for Practical Extraction and Report Language. It runs on a variety of platforms, such as Windows, Mac OS, and the various versions of UNIX. This tutorial provides a complete understanding of Perl.

Perl Features

- Perl takes the best features from other languages, such as C, awk, sed, sh, and BASIC, among others.
- Perl's database integration interface DBI supports third-party databases including Oracle, Sybase, Postgres, MySQL, and others.
- Perl works with HTML, XML, and other mark-up languages.
- Perl supports Unicode.
- Perl is Y2K compliant.
- Perl supports both procedural and object-oriented programming.
- Perl interfaces with external C/C++ libraries through XS or SWIG.
- Perl is extensible. There are over 20,000 third-party modules available from the Comprehensive Perl Archive Network.
- The Perl interpreter can be embedded into other systems.

Exercise

1. Perform arithmetic operations using Perl

Source Code :

```
print "enter 2 numbers\n";
$a = <STDIN>;
$b = <STDIN>;
$c = $a + $b;
print 'Value of $a + $b = ' . $c . "\n";

$c = $a - $b;
print 'Value of $a - $b = ' . $c . "\n";

$c = $a * $b;
print 'Value of $a * $b = ' . $c . "\n";

$c = $a / $b;
print 'Value of $a / $b = ' . $c . "\n";

$c = $a % $b;
print 'Value of $a % $b = ' . $c. "\n";
```

Output :

```
$perl main.pl
enter 2 numbers
Value of $a + $b = 30
Value of $a - $b = 10
Value of $a * $b = 200
Value of $a / $b = 2
Value of $a % $b = 0
```

2. Demonstrate all the escape sequences using print and say statements.

Source Code :

```
print("Displaying text \n in new line by using \\n. \\n");
print("Displaying text \t in new tab by using \\t. \\n");
print(\"Printing in double quotes.\"\\n");
print('\\'Printing in single quotes.'\\n");
```

```
print("\L printing text in lower case.\n");
print("\U PRINTING TEXT IN UPPER CASE.\n");
```

Output :

```
$perl main.pl
Displaying text
in new line by using \n.
Displaying text      in new tab by using \t.
"Printing in double quotes."
'Printing in single quotes.'
printing text in lower case.
PRINTING TEXT IN UPPER CASE.
```

3. Online shopping application.

Source Code :

```
print("Enter the item:\n 1.Air conditioner(40k) \n 2.television(30k) \n
3.Refrigerator(20k) \n 4.Air cooler(15k) \n");
$c = <STDIN>;
print("Enter quantity\n");
$a = <STDIN>;
if ($c ==1)
{
    $cost = $a * 40000;
    print("cost of Air conditioner: Rs $cost\n");
}
elsif ($c ==2)
{
    $cost = $a * 30000;
    print("cost of television: Rs $cost\n");
}
elsif($c ==3)
{
    $cost = $a * 20000;
    print("cost of Refrigerator: Rs $cost\n");
}
elsif($c ==4)
{
    $cost = $a * 15000;
    print("cost of Air cooler: Rs $cost\n");
}
```

```
else
{
    print("Invalid option");
}
```

Output :

```
$perl main.pl
Enter the item:
1.Air conditioner(40k)
2.telivision(30k)
3.Refrigerator(20k)
4.Air cooler(15k)
Enter quantity
cost of Refrigerator: Rs 20000
```

```
$perl main.pl
Enter the item:
1.Air conditioner(40k)
2.telivision(30k)
3.Refrigerator(20k)
4.Air cooler(15k)
Enter quantity
cost of telivision: Rs 90000
```

4.Demonstrate arithmetic assignment operators in Perl.

Source Code :

```
print("Enter a value\n");
$a = <STDIN>;
print("Enter Increment value\n");
$b = <STDIN>;
$b += $a;
```

```
print("Using increment assignment operator $b\n");
print("Enter decrement value\n");
$b = <STDIN>;
$b -= $a;
print("Using decrement assignment operator $b\n");
print("Enter Multiply value\n");
$b = <STDIN>;
$b *= $a;
print("Using multiplication assignment operator $b\n");
print("Enter division value\n");
$b = <STDIN>;
$b /= $a;
print("Using division assignment operator $b\n");
print("Enter modulus value\n");
$b = <STDIN>;
$b %= $a;
print("Using modulus assignment operator $b\n");
```

Output :

```
$perl main.pl
Enter a value
Enter Increment value
Using increment assignment operator 13
Enter decrement value
Using decrement assignment operator -6
Enter Multiply value
Using multiplication assignment operator 60
Enter division value
Using division assignment operator 1
Enter modulus value
Using modulus assignment operator 8
```

Results:

The study and implementation of basic Perl programming are studied and executed.

Video : <https://youtu.be/kq5J5Vj8zHs>

Ex 9

Fundamental operations of Perl

Date: 10.11.2020

Aim:

To study and implement the fundamental operations of Perl programming.

Description

Perl is a programming language developed by Larry Wall, specially designed for text processing. It stands for Practical Extraction and Report Language. It runs on a variety of platforms, such as Windows, Mac OS, and the various versions of UNIX. This tutorial provides a complete understanding of Perl.

Perl Arrays

An array is a variable that stores an ordered list of scalar values. Array variables are preceded by an "at" (@) sign. To refer to a single element of an array, you will use the dollar sign (\$) with the variable name followed by the index of the element in square brackets.

Here is a simple example of using the array variables :

```
#!/usr/bin/perl  
  
@ages = (25, 30, 40);  
  
@names = ("John Paul", "Lisa", "Kumar");  
  
print "\$ages[0] = $ages[0]\n";  
print "\$ages[1] = $ages[1]\n";  
print "\$ages[2] = $ages[2]\n";  
  
print "\$names[0] = $names[0]\n";  
print "\$names[1] = $names[1]\n";  
print "\$names[2] = $names[2]\n";
```

Arithmetic Operators of Perl

Sl.No	Operator & Description
1	+ (Addition) Example – \$a + \$b will give 30
2	- (Subtraction) Example – \$a - \$b will give -10
3	* (Multiplication) Example – \$a * \$b will give 200
4	/ (Division) Example – \$b / \$a will give 2
5	% (Modulus) Example – \$b % \$a will give 0
6	** (Exponent) Example – \$a**\$b will give 10 to the power 20

Exercise

1. Demonstrate various operators in Perl.

Source Code :

```
print("choose an option: \n 1.ArithmetiC operation \n\n 2. comparison operation \n 3.string operation \n 4.Bitwise operation\n");

$c = <STDIN>;\n\nif ($c == 1)\n{\n    print("ARITHMETIC OPERATIONS\n");\n\n    print("Enter two values\n");\n}
```

```
$a = <STDIN>;  
  
$b = <STDIN>;  
  
print("choose an option: \n 1.Addition \n 2.Subtraction \n 3.Multiplication \n  
4.Division \n 5.Modulus \n");  
  
$op = <STDIN>;  
  
if($OP == 1)  
{  
    $tot = $a+$b;  
  
    print("sum of numbers: $tot\n");  
  
}  
  
elsif ($op == 2)  
{  
    $tot = $a-$b;  
  
    print("Difference beteween both values: $tot\n");  
  
}  
  
elsif($op == 3)  
{  
    $tot = $a*$b;  
  
    print("product of both the numbers: $tot\n")  
  
}  
  
elsif ($op == 4)  
{  
    $tot = $a/$b;  
  
    print("Quotinent of both the numbers: $tot\n");
```

```
    }

    elseif ($op == 5)

    {

        $tot = $a % $b;

        print("Modulus of both the numbers: $tot\n");

    }

    else

    {

        print ("Invalid Option\n");

    }

}

elseif ($c == 2)

{

    print("COMPARISION OPERATOR\n");

    print("Enter two values\n");

    $a = <STDIN>;

    $b = <STDIN>;

    if($a == $b)

    {

        print("The values are equal\n");

    }

    elseif($a > $b)

    {

        print ("Greater value: $a\n");

    }

}
```

```
print ("Lower values: $b\n");
}

else

{

print("Lower value: $a\n");
print("Greater value: $b\n");
}

if($a != $b)

{

print("Both are not equal\n");

}

}

elseif ($c == 3)

{

print("STRING OPERATION\n");

print("Enter two strings\n");

$x = <STDIN>;

$y = <STDIN>;

$z = $x.$y;

print("Concatenated String: $z");

}

elseif ($c == 4)

{

print("BITWISE OPERATOR\n");
```

```
print("Enter two values\n");

$a = <STDIN>;
$b = <STDIN>;
$and = $a & $b;
print("Performing bitwise AND: $and\n");
$or = $a | $b;
print("Printing bitwise OR: $or");
$left = $a<<2;
print("Performing left shift: $left");
$right = $a>>2;
print("Performing right shift: $right");
}

else
{
    print("Invalid Operation");
}
```

Output :

```
choose an option:  
1.Arithmetic operation  
  
2. comparison operation  
3.string operation  
4.Bitwise operation  
1  
ARITHMETIC OPERATIONS  
Enter two values  
10  
20  
choose an option:  
1.Addition  
2.Subtraction  
3.Multiplication  
4.Division  
5.Modulus  
2  
Difference beteween both values: -10
```

```
choose an option:  
1.Arithmetic operation  
  
2. comparison operation  
3.string operation  
4.Bitwise operation  
4  
BITWISE OPERATOR  
Enter two values  
10  
70  
Performing bitwise AND: 10  
  
Printing bitwise OR: 70  
Performing left shift: 40Performing right shift: 2
```

2.Print 10 inputs received from the user.

Source Code :

```
print("Enter 10 values\n");
for($i=0;$i<10;$i++)
{
    $arr[$i]=<STDIN>;
}
print("The Values\n");
for($i=0;$i<10;$i++)
{
    print("$arr[$i]");
}
```

Output :

```
Enter 10 values
hello
there
how
are
you
whats
your
name
whats up

The Values
hello
there
how
are
you
whats
your
name
whats up
```

Results:

The study and implementation of fundamental operations of Perl programming are studied and executed.

Video : <https://youtu.be/BS5iyoA-Aqc>

Ex 10

KERNEL INSTALLATION

Date: 17.11.20

Aim:

To study and implement the kernel installation .

Description:

Kernel Space:

Here, the Linux Kernel exists which can be further divided into three levels. At the top is the system call interface, which implements the basic functions such as read and write. Below the system call interface is the kernel code, which can be more accurately defined as the architecture-independent kernel code. This code is common to all of the processor architectures supported by Linux. Below this is the architecture-dependent code, which forms what is more commonly called a BSP (Board Support Package). This code serves as the processor and platform-specific code for the given architecture.

Sl. N o.	Command Name	Meaning	Description
1	rpm -qa kernel-devel	It displays the version of the kernel.	Kernel-devel - This package provides kernel headers and makes files sufficient to build modules against the kernel package.
2	uname –r	uname displays the information about the system.	The command ‘uname‘ displays the information about the system. option : -a It prints all the system information in the following order: Kernel name, network node hostname, kernel release date, kernel version, machine hardware name, hardware platform, operating system -s It prints the kernel name. -n It prints the hostname of the network node -r It prints the kernel release date

			-v It prints the version of the current kernel
3	tar	tar stands for tape archive, is used to create Archive and extract the Archive files	<p>tar command in Linux is one of the important commands which provides archiving functionality in Linux. We can use Linux tar command to create compressed or uncompressed Archive files and also maintain and modify them.</p> <p>Options:</p> <ul style="list-style-type: none"> -c : Creates Archive -x : Extract the archive -f : creates archive with given filename -t : displays or lists files in archive file -u : archives and adds to an existing archive file -v : Displays Verbose Information -A : Concatenates the archive files -z : zip, tells tar command that create tar file using gzip -j : filter archive tar file using tbzip -W : Verify a archive file -r : update or add file or directory in already existed .tar file
4	ln	A symbolic link, also known as a symlink or soft link, is a special type of file that points to another file or directory.	There are two types of links in Linux/UNIX systems: 1. Hard links 2. Soft links

Exercise

Step 1 :

Building kernel and its modules.

Step 2 :

Check the current kernel version and name of the kernel.

Output:

Moving the kernel to /usr/src

```
ubuntu@ubuntu:/usr/src$ sudo cp /home/ubuntu/Downloads/linux-5.8.7.tar.xz .
ubuntu@ubuntu:/usr/src$ ls
linux-5.8.7  linux-5.8.7.tar.xz  linux-headers-5.4.0-42  linux-headers-5.4.0-42
-generic  linux-headers-5.4.0-45  linux-headers-5.4.0-45-generic
ubuntu@ubuntu:/usr/src$
```

Cleaning the kernel using make

```
ubuntu@ubuntu:/usr/src/linux-5.8.7$ sudo make clean
ubuntu@ubuntu:/usr/src/linux-5.8.7$
```

System link to existing kernel

```
ubuntu@ubuntu:~$ sudo ln -s /usr/src/linux-5.8.7 /usr/src/linux-headers-5.4.0-42-generic/
ubuntu@ubuntu:~$
```

Making target files

```
ubuntu@ubuntu:/usr/src/linux-5.8.7$ sudo make menuconfig
LEX      scripts/kconfig/lexer.lex.c
YACC     scripts/kconfig/parser.tab.[ch]
HOSTCC   scripts/kconfig/lexer.lex.o
HOSTCC   scripts/kconfig/parser.tab.o
HOSTCC   scripts/kconfig/preprocess.o
HOSTCC   scripts/kconfig/symbol.o
HOSTCC   scripts/kconfig/util.o
HOSTLD   scripts/kconfig/mconf
scripts/kconfig/mconf  Kconfig
#
# using defaults found in /boot/config-5.4.0-45-generic
#
/boot/config-5.4.0-45-generic:3815:warning: symbol value 'm' invalid for ISDN_CAPI
/boot/config-5.4.0-45-generic:8245:warning: symbol value 'm' invalid for ASHMEM
/boot/config-5.4.0-45-generic:9205:warning: symbol value 'm' invalid for ANDROID_BINDER_IPC
/boot/config-5.4.0-45-generic:9206:warning: symbol value 'm' invalid for ANDROID_BINDERFS
/boot/config-5.4.0-45-generic:9274:warning: symbol value 'm' invalid for INTERCONNECT

*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.
```

New kernel proof

```
ubuntu@ubuntu:/usr/src/linux-5.8.7$ sudo make
HOSTCC scripts/kconfig/conf.o
HOSTLD scripts/kconfig/conf
scripts/kconfig/conf --syncconfig Kconfig
SYSTBL arch/x86/include/generated/asm/syscalls_32.h
SYSHDR arch/x86/include/generated/asm/unistd_32_ia32.h
SYSHDR arch/x86/include/generated/asm/unistd_64_x32.h
SYSTBL arch/x86/include/generated/asm/syscalls_64.h
HYPERCALLS arch/x86/include/generated/asm/xen-hypercalls.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_32.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_64.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_x32.h
HOSTCC arch/x86/tools/relocs_32.o
HOSTCC arch/x86/tools/relocs_64.o
HOSTCC arch/x86/tools/relocs_common.o
HOSTLD arch/x86/tools/relocs
HOSTCC scripts/selinux/genheaders/genheaders
HOSTCC scripts/selinux/mdp/mdp
HOSTCC scripts/bin2c
HOSTCC scripts/kallsyms
HOSTCC scripts/recordmcount
HOSTCC scripts/sorttable
HOSTCC scripts/asn1_compiler
HOSTCC scripts/sign-file
```

Results:

The kernel installation is studied and executed.

Video : <https://youtu.be/qTaEOzyoNs4>