

Lecture-1

Computer Networks Laboratory

Goddu Vishal

17114034(CS-1)

Introduction

This is the report containing the snapshots of my terminal which demonstrates that my code for each of the problem statements is executing.

Problems

1)

Write a C program in the UNIX system that creates two children and four grandchildren (two for each child). The program should then print the process-IDs of the two children, the four grandchildren and the parent in this order.

In this problem all I have used is a pointer to each of the processes and a fork() function to validate the process and add child and grandchild processes.

```
vishal@Bob:~/Desktop/1$ ./fir-out
[son] pid 4058 from [parent] pid 4057
[son] pid 4059 from [parent] pid 4057
[grandson] pid 4060 from [son] pid 4058
[grandson] pid 4062 from [son] pid 4059
[grandson] pid 4061 from [son] pid 4058
[grandson] pid 4063 from [son] pid 4059
```

2)

Write a C++ program to print the MAC address of your computer.

This code uses the structure ifreq that is part of the library if.h

```
vishal@Bob:~/Desktop/1$ ./mac-out
AC:E2:D3:3A:94:6C
```

3)

Write a C program to find the host name from IP address

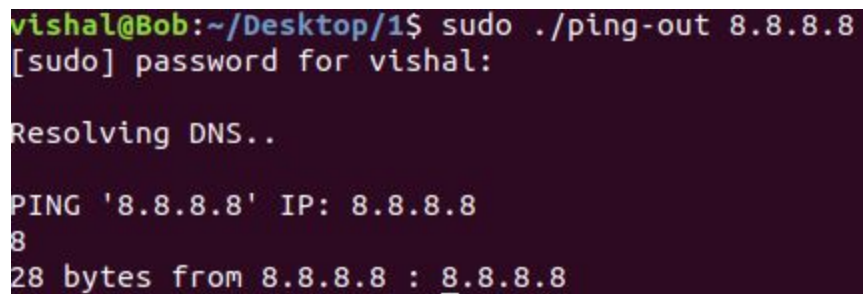
This code uses the structure hostent.

```
vishal@Bob:~/Desktop/1$ ./ip-out
Hostname: Bob
Host IP: 127.0.1.1vishal@Bob:~/Desktop/1$
```

4)

Write your own version of ping program in C language.

This code contains the icmp structure whose description is in the code, socket structure is used from the library.

A terminal window with a dark purple background. The prompt is 'vishal@Bob:~/Desktop/1\$'. The command 'sudo ./ping-out 8.8.8.8' is entered. The output shows '[sudo] password for vishal:', 'Resolving DNS..', 'PING '8.8.8.8' IP: 8.8.8.8', '8', and '28 bytes from 8.8.8.8 : 8.8.8.8'.

```
vishal@Bob:~/Desktop/1$ sudo ./ping-out 8.8.8.8
[sudo] password for vishal:
Resolving DNS..
PING '8.8.8.8' IP: 8.8.8.8
8
28 bytes from 8.8.8.8 : 8.8.8.8
```