LINUX PROGRAMMING

WEEK 8

Write a C program that illustrates how an orphan is created.

```
#include <stdio.h>
main()
{
int pid ;
printf("I'am the original process with PID %d and PPID
%d. \n",getpid(),getppid());
pid=fork();
if(pid!=0 )
{
printf("I'am the parent with PID %d and PPID
%d. \n",getpid(),getppid());
printf("My child's PID is %d \n",pid) ;
}
else
{
sleep(4);
printf("I'm the child with PID %d and PPID %d. \n",getpid(),
getppid()) ;
}
printf ("PID %d terminates. \n", getpid()) ;
}
```

Output:

guest-glcbls@ubuntu:~$gcc –o prg18.out prg18.c

guest-glcbls@ubuntu:~$./prg18.out

I am the original process with PID2242 and PPID1677.

I am the parent with PID2242 and PPID1677

My child’s PID is 2243

PID2243 terminates.

$ I am the child with PID2243 and PPID1.

PID2243 termanates.

Write a program that illustrates how to execute two commands concurrently with a command pipe.

```
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <stdlib.h>
int main()
{
int pfds[2];
char buf[30];
if(pipe(pfds)==-1)
{
perror("pipe failed");
exit(1);
}
if(!fork())
{
close(1);
dup(pfds[1];
system (“ls –l”);
}
else
{
printf("parent reading from pipe \n");
while(read(pfds[0],buf,80))
printf("%s \n" ,buf);
}
}
```

```
[STUDENT@GCET ~]$ VI PIPES2.C
[STUDENT@GCET ~]$ CC PIPES2.C
[STUDENT@GCET ~]$ ./A.OUT
PARENT READING FROM PIPE
TOTAL 24
-RWXRWXR-X L STUDENT STUDENT 5563AUG 3 10:39 A.OUT
-RW-RW-R—L
STUDENT STUDENT 340 JUL 27 10:45 PIPE2.C
-RW-RW-R—L STUDENT STUDENT
PIPES2.C
-RW-RW-R—L STUDENT STUDENT 401 34127 10:27 PIPE2.C
STUDENT
```

Write a C programs that illustrate communication between two unrelated processes using named pipe.

```
#include<stdio.h>
#include<stdlib.h>
#include<errno.h>
#include<unistd.h>
int main()
{
int pfds[2];
char buf[30];
if(pipe(pfds)==-1)
{
perror("pipe");
exit(1);
}
printf("writing to file descriptor #%d\n", pfds[1]);
write(pfds[1],"test",5);
printf("reading from file descriptor #%d\n ", pfds[0]);
read(pfds[0],buf,5);
printf("read\"%s\"\n" ,buf);
}
```

```
[STUDENT@GCET ~]$ VI PIPES1.C
[STUDENT@GCET ~]$ CC PIPES1.C
[STUDENT@GCET ~]$ ./A.OUT
WRITING TO FILE DESCRIPTOR #4
READING FROM FILE DESCRIPTOR #3
READ"TEST"
```

THANK YOU