**LAB REPORT NO 6**



Submitted by:  **Muhammad Ali**

Registration No: - **19PWCSE1801**

Class Section: A

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Submitted to:

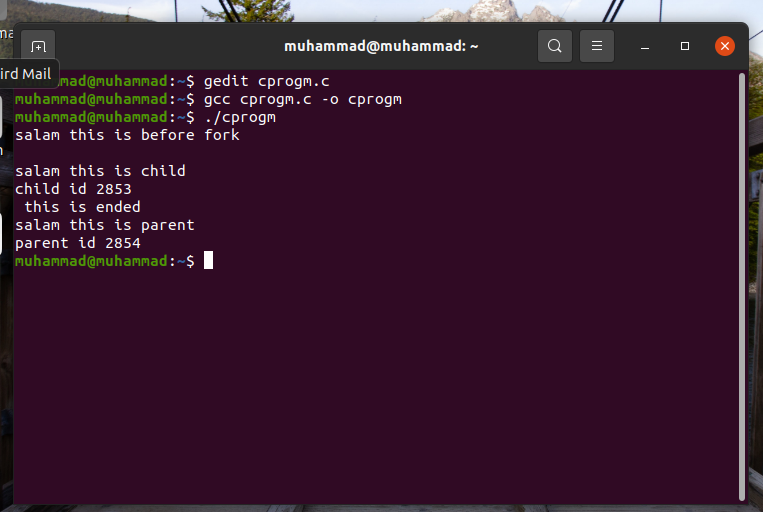
**Engr.Mian Ibad Ali shah**

Data:(15,07,2021)

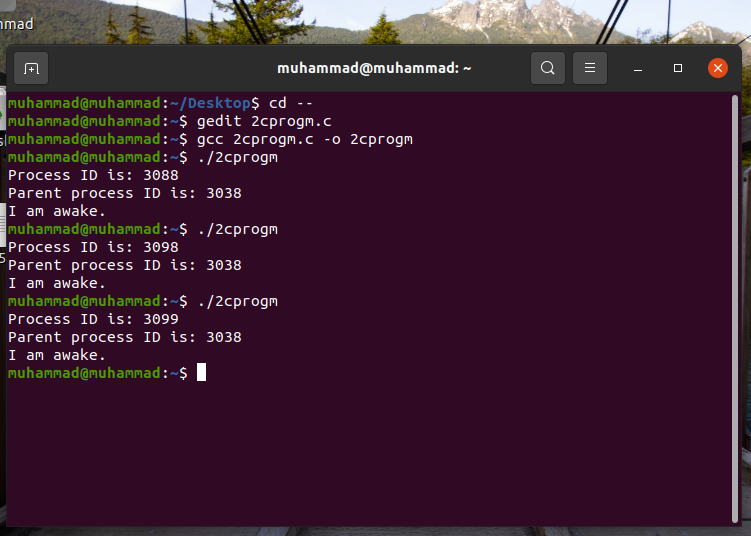
Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

**Creating simple child: -**



**Task no 1: -**



**Observation: -**

I run the program three times with printing process and parent id’s, in which process id of the child are different but parent id are same within three times execution. This is because make a child in same main() function so id of main() function is same for each time execution.

**Code: -**

#include<stdio.h>

#include<unistd.h>

main ( ) {

    printf ("Process ID is: %d\n", getpid( ) ) ;

    printf ("Parent process ID is: %d\n", getppid( ) ) ;

    sleep (60);

    printf ("I am awake. \n");

}

**Task no 2: -**

**Observation: -**

Two fork() will create three child process, in the below program two child created while executed the same print function by parent.

**Code: -**

#include<stdio.h>

#include<unistd.h>

main () {

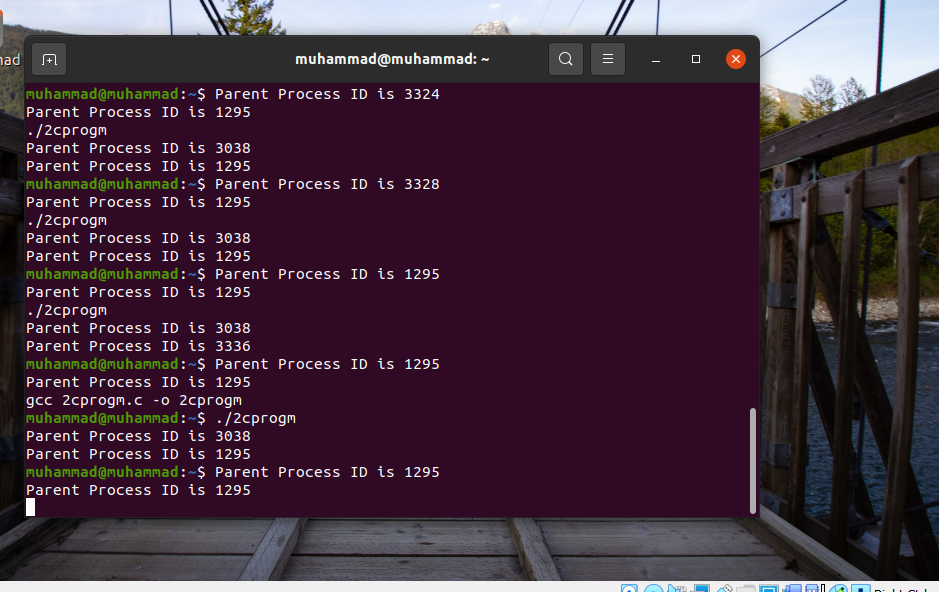
    fork ();

    fork ();

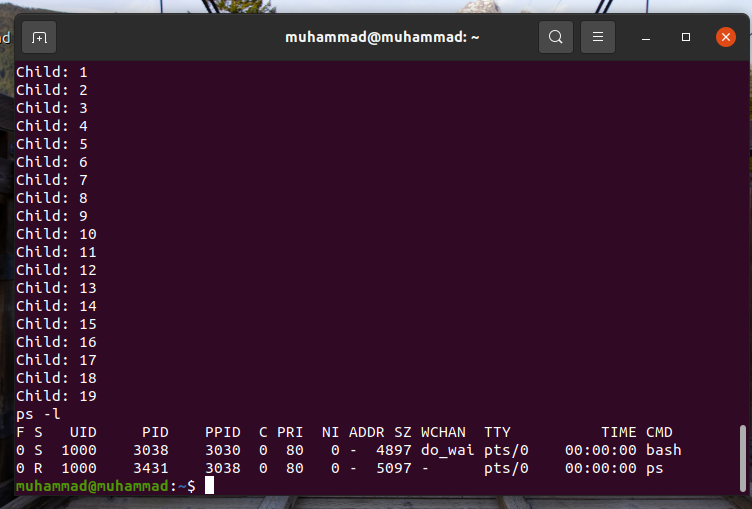
    printf ("Parent Process ID is %d\n", getppid ());

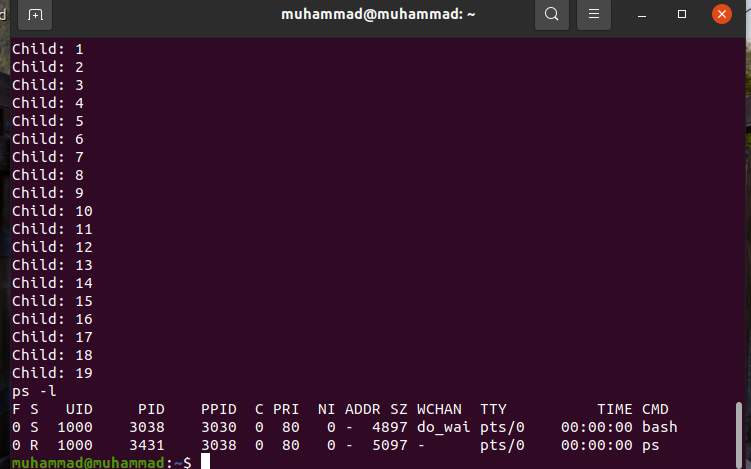
}

**Tree process: -**



**Task no 3: -**





**Task no 4: -**

**Observation: -**

In this program I created 8 child by calling three fork() functions from one parent.

**Code: -**

#include<stdio.h>

#include<unistd.h>

int main(void)

{

fork();

fork();

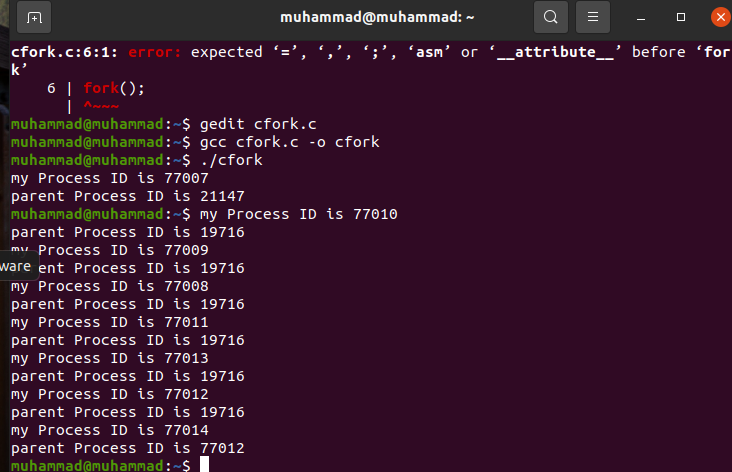
fork();

printf ("my Process ID is %d\n", getpid());

printf ("parent Process ID is %d\n", getppid());

return 0;

}

****

**Task no 5: -**

**Observation: -**

In this program I create chain tree in which child is created by previous one as code below, 6 child is created one after another.

**Code: -**

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

#include<sys/wait.h>

int main()

{

int pid=fork();

for(int i=1;i<7;i++)

{

if(pid==0)

{

printf ("this parent id %d and my child id is %d \n",getppid(),getpid());

pid=fork();

}

}

for (int i=1; i<7; i++)

{

wait(NULL);

}

exit(0);

}

****

**Task no 6: -**

**Code: -**

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

#include<sys/wait.h>

int main()

{

fork();

fork();

fork();

fork();

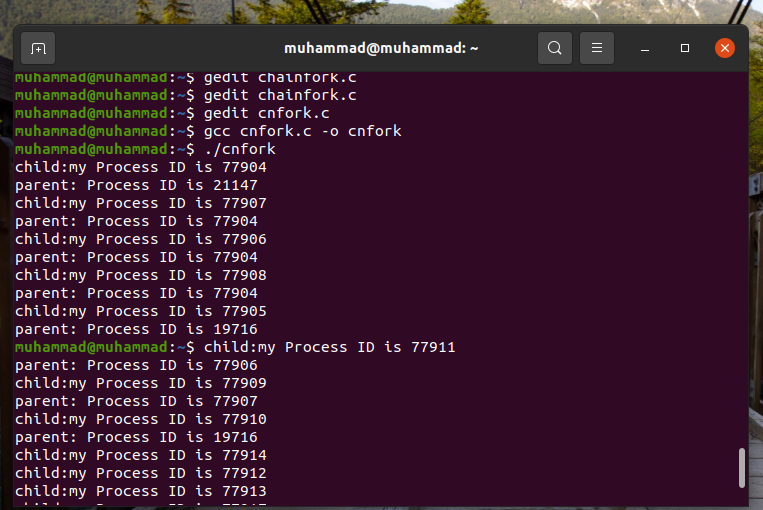
printf ("child:my Process ID is %d\n", getpid());

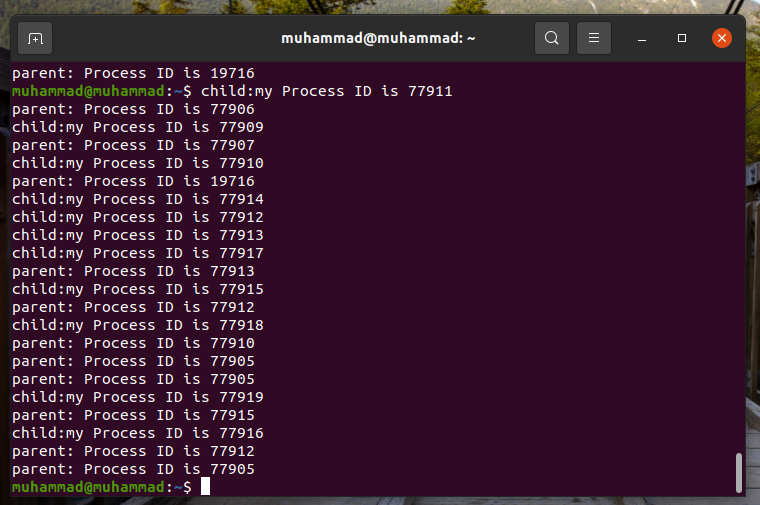
printf ("parent: Process ID is %d\n", getppid());

wait(NULL);

}

**Tree diagram: -**

****

****