

Name - Suresh Dub(2021A1R030)

Experiment : 10

1.) Write a program that creates a child process, wait for the termination of child and list its PID.

- C Program

```
suresh@suresh-QEMU-Virtual-Machine: ~  
#include<unistd.h>  
#include<sys/types.h>  
#include<stdio.h>  
#include<sys/wait.h>  
int main()  
{  
    pid_t q;  
    q= fork();  
    if (q==0) // child nodes  
    {  
        printf("I am child having id %d\n", getpid());  
        printf("My parent id %d\n", getppid());  
    }  
    else  
    {  
        printf("My child id is %d \n", q);  
        wait (NULL); // Basically we're using wait : to wait parent process id to print children process id first.  
        printf("I am parent having id %d\n", getpid());  
    }  
    printf("Common\n");  
    return 0;  
}  
~  
~  
~  
~
```

- Run the following C Program

```
suresh@suresh-QEMU-Virtual-Machine: ~  
suresh@suresh-QEMU-Virtual-Machine:~$ gcc pro.c  
suresh@suresh-QEMU-Virtual-Machine:~$ ./a.out  
My child id is 4186  
I am child having id 4186  
My parent id 4185  
Common  
I am parent having id 4185  
Common  
suresh@suresh-QEMU-Virtual-Machine:~$ |
```