

## **Experiment9:**

**Write a program in C that creates a child process, waits for the termination of the child and lists its PID.**

**Command:** nano filename.c

**Program:** #include<unistd.h>

```
#include<sys/types.h>
```

```
#include<stdio.h>
```

```
#include<sys/wait.h>
```

```
int main()
```

```
{
```

```
pid_t p;
```

```
printf("before fork\n");
```

```
p=fork();
```

```
if(p==0)
```

```
{
```

```
printf("I am child having id %d\n",getpid());
```

```
printf("My parent's id is %d\n",getppid());
```

```
}
```

```
else
```

```
{
```

```
wait(NULL);
```

```
printf("My child's id is %d\n",p);
```

```
printf("I am parent having id %d\n",getpid());
```

```
}  
  
printf("Common\n");  
  
}
```

**Terminal:** gcc filename.c

./a.out

```
#include<unistd.h>  
#include<sys/types.h>  
#include<stdio.h>  
#include<sys/wait.h>  
int main()  
{  
    pid_t p;  
    printf("before fork\n");  
    p=fork();  
    if(p==0)  
    {  
        printf("I am child having id %d\n",getpid());  
        printf("My parent's id is %d\n",getppid());  
    }  
    else  
    {  
        wait(NULL);  
        printf("My child's id is %d\n",p);  
        printf("I am parent having id %d\n",getpid());  
    }  
    printf("Common\n");  
}
```

```
sahil@sahil-virtualbox:/mnt/d$ nano exp9.c  
sahil@sahil-virtualbox:/mnt/d$ gcc exp9.c  
sahil@sahil-virtualbox:/mnt/d$ ./a.out  
before fork  
I am child having id 162  
My parent's id is 161  
Common  
My child's id is 162  
I am parent having id 161  
Common
```