

# **EXPERIMENT 8**

**NAME: Rahil Sharma**

**PRN: 18070123062**

**BATCH: EA-3**

**SUBJECT: ESRTOS**

**AIM:** Write a C program to print Child and Parent PID and PPID

**THEORY:** fork() is the primary method of process creation on Unix-like operating systems. This function creates a new copy called the child out of the original process, that is called the parent. When the parent process closes or crashes for some reason, it also kills the child processes.

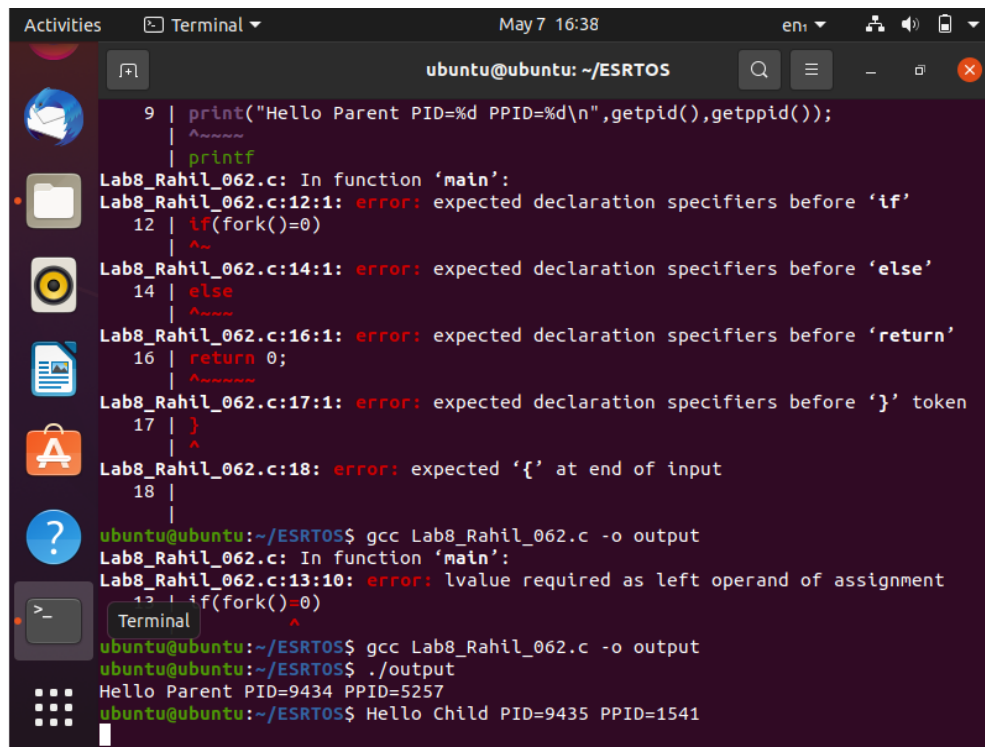
PID stands for Process ID, Which means Identification Number for currently running process in Memory.

PPID stands for Parent Process ID, Which means Parent Process is the responsible for creating the current process(Child Process). Through Parent Process, The child process will be created.

## CODE OF THE PROGRAM:

```
# include <stdio.h>
#include<sys/types.h>
#include<unistd.h>
void child()
{
printf("Hello Child PID=%d PPID=%d\n", getpid() , getppid() );
}void parent()
{
printf("Hello Parent PID=%d PPID=%d\n", getpid() , getppid() );
}
int main()
{
if(fork()==0)
child();
else
parent();
return 0;
}
```

## SCREENSHOTS OF THE OUTPUT OF THE PROGRAM:



The screenshot shows a terminal window titled 'Terminal' with the prompt 'ubuntu@ubuntu: ~/ESRTOS'. The user has entered the following code into the terminal:

```
9 | printf("Hello Parent PID=%d PPID=%d\n",getpid(),getppid());
   | printf
Lab8_Rahil_062.c: In function 'main':
Lab8_Rahil_062.c:12:1: error: expected declaration specifiers before 'if'
12 | if(fork()==0)
   | ^~
Lab8_Rahil_062.c:14:1: error: expected declaration specifiers before 'else'
14 | else
   | ^~~~~
Lab8_Rahil_062.c:16:1: error: expected declaration specifiers before 'return'
16 | return 0;
   | ^~~~~~
Lab8_Rahil_062.c:17:1: error: expected declaration specifiers before '}' token
17 | }
   | ^
Lab8_Rahil_062.c:18: error: expected '{' at end of input
18 |
   |
```

The user then attempts to compile the program with the command:

```
ubuntu@ubuntu:~/ESRTOS$ gcc Lab8_Rahil_062.c -o output
```

This results in another error:

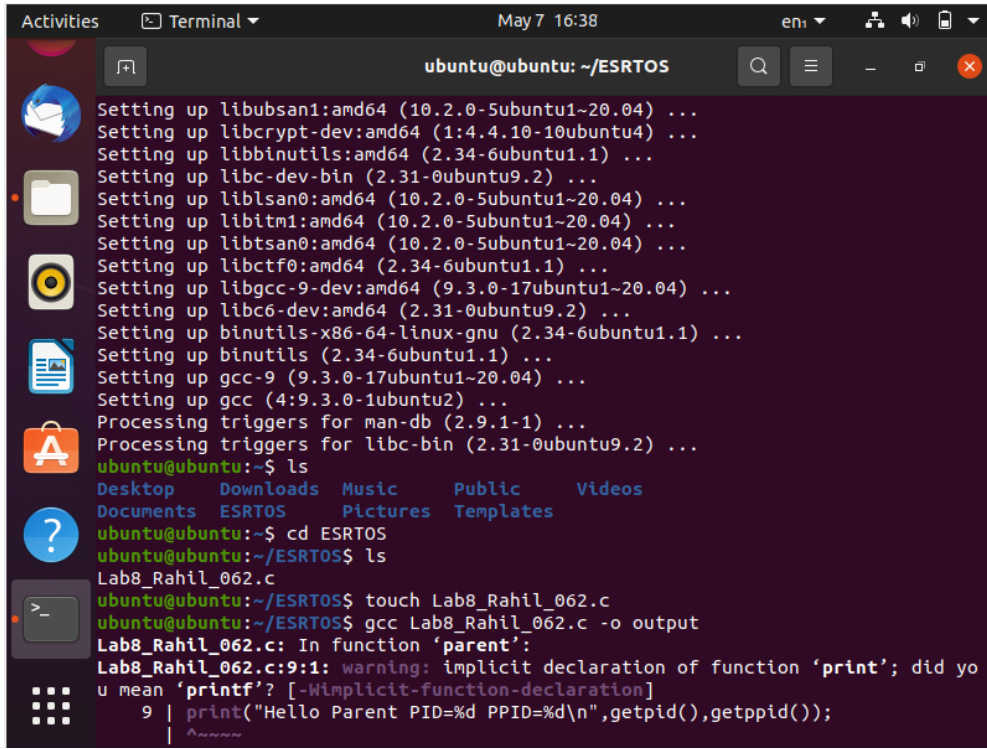
```
Lab8_Rahil_062.c: In function 'main':
Lab8_Rahil_062.c:13:10: error: lvalue required as left operand of assignment
13 | if(fork()==0)
   | ^~
```

The user then runs the command:

```
ubuntu@ubuntu:~/ESRTOS$ gcc Lab8_Rahil_062.c -o output
ubuntu@ubuntu:~/ESRTOS$ ./output
```

The output of the program is displayed:

```
Hello Parent PID=9434 PPID=5257
ubuntu@ubuntu:~/ESRTOS$ Hello Child PID=9435 PPID=1541
```



```
Activities Terminal May 7 16:38 en: ubuntu@ubuntu: ~/ESRTOS

Setting up libubsan1:amd64 (10.2.0-5ubuntu1~20.04) ...
Setting up libcrypt-dev:amd64 (1:4.4.10-10ubuntu4) ...
Setting up libbinutils:amd64 (2.34-6ubuntu1.1) ...
Setting up libc-dev-bin (2.31-0ubuntu9.2) ...
Setting up liblsan0:amd64 (10.2.0-5ubuntu1~20.04) ...
Setting up libitm1:amd64 (10.2.0-5ubuntu1~20.04) ...
Setting up libtsan0:amd64 (10.2.0-5ubuntu1~20.04) ...
Setting up libctf0:amd64 (2.34-6ubuntu1.1) ...
Setting up libgcc-9-dev:amd64 (9.3.0-17ubuntu1~20.04) ...
Setting up libc6-dev:amd64 (2.31-0ubuntu9.2) ...
Setting up binutils-x86-64-linux-gnu (2.34-6ubuntu1.1) ...
Setting up binutils (2.34-6ubuntu1.1) ...
Setting up gcc-9 (9.3.0-17ubuntu1~20.04) ...
Setting up gcc (4:9.3.0-1ubuntu2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
ubuntu@ubuntu:~$ ls
Desktop  Downloads  Music      Public     Videos
Documents ESRTOS     Pictures   Templates
ubuntu@ubuntu:~$ cd ESRTOS
ubuntu@ubuntu:~/ESRTOS$ ls
Lab8_Rahil_062.c
ubuntu@ubuntu:~/ESRTOS$ touch Lab8_Rahil_062.c
ubuntu@ubuntu:~/ESRTOS$ gcc Lab8_Rahil_062.c -o output
Lab8_Rahil_062.c: In function 'parent':
Lab8_Rahil_062.c:9:1: warning: implicit declaration of function 'print'; did you
mean 'printf'? [-Wimplicit-function-declaration]
   9 |   print("Hello Parent PID=%d PPID=%d\n",getpid(),getppid());
     |   ^~~~~
```

**CONCLUSION:** From this program we learnt the fork() statement function in C and about PID and PPID.