

## Experiment – 9

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

```
GNU nano 6.2 exp9.c
#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)//child
    {
        printf("I am child having id : %d\n", getpid());
        printf("My parent's id is : %d\n", getppid());
    }
    else //parent
    {
        wait(NULL);
        printf("My child's id is %d \n", p);
        printf("I am parent having id %d\n", getpid());
    }
    printf("Common\n");
}
```

```
sia@sia-VirtualBox: ~/folder1
Address of malloc in parent = 0x5569a76cb2a0
Address of var in child = 0x7ffc45540538
sia@sia-VirtualBox:~/folder1$ ls
10pt1.c  3pt3.sh  5pt0.sh  a.out  dir.sh  exp7io.c  matrixmult.c  sample-f
ile.txt  test
10pt2.c  4pt1.sh  5pt1.c  a.txt  exp6.c  exp9.c  mergesort2.c  signin_p
age.sh   timer2.sh
3pt1.sh  4pt2.sh  5pt2.c  b.txt  exp7cpu2.c  index_page.sh  mergesort.c  signup_p
age.sh   timer.sh
3pt2.sh  4pt3.sh  5pt2.sh  c.txt  exp7cpu.c  indexpage.sh  nf.txt       start.sh
user_database.csv
sia@sia-VirtualBox:~/folder1$ gcc exp9.c
sia@sia-VirtualBox:~/folder1$ ./a.out
Before fork
I am child having id : 3753
My parent's id is : 3752
Common
My child's id is 3753
I am parent having id 3752
Common
sia@sia-VirtualBox:~/folder1$ nano exp9.c
sia@sia-VirtualBox:~/folder1$
```

