

## **ASSIGNMENT - II**

### **TITLE :**

Implement a program using fork(), exec(), and wait() system calls to create a child process, execute a new program in the child, and synchronize the parent process to wait for the child to complete.

**NAME :** Shinde Shubham Dnyandev,      **DIV :** SY-B,      **ROLL NO. :** 23107121.

### **PROGRAMS :**

#### **1) Fork() & wait() :**

```
#include<stdio.h>
#include<unistd.h>
#include<stdlib.h>
#include<sys/types.h>

int main() {
    pid_t ID;
    ID = fork();
    if (ID<0)
        {
            printf("Error");
        }
    else if(ID==0)
        {
            printf("\nChild Process : %d",getpid());
        }
    else
        {
            wait();
            printf("\nP
```

arent

Process :

```
%d",getpid
```

```
();
```

```
printf("\n")
```

```
;
```

```
}
```

```
return 0;
```

```
}
```

## OUTPUT :

```
● shubham@ShubhsPC:~$ gcc fork.c
fork.c: In function ‘main’:
fork.c:20:9: warning: implicit declaration of function ‘wait’ [-Wimplicit-function-declaration]
  20 |         wait();
     |         ^
● shubham@ShubhsPC:~$ ./a.out
Child Process : 5025
Parent Process : 5024
● shubham@ShubhsPC:~$
```

## 2) execl() & execlp() :

### a) execl() :

```
#include<stdio.h>

#include<unistd.h> int main() {

char*file="/bin/ls";
char*arg1="-r";
char*arg2="-l";
execl(file,file,arg1,arg2,NULL);

return 0;
}
```

## OUTPUT :

```

● shubham@ShubhsPC:~$ gcc execl.c
● shubham@ShubhsPC:~$ ./a.out
total 24
-rw-r--r-- 1 shubham shubham  379 Jan 15 13:30 fork.c
-rw-r--r-- 1 shubham shubham  167 Jan 15 13:48 execl.c
-rwxr-xr-x 1 shubham shubham 15960 Jan 15 13:48 a.out
○ shubham@ShubhsPC:~$ █

```

### b) execlp() :

```

#include<stdio.h>

#include<unistd.h>

int main()

{
    char*file_name="ls";

    char*arg1="-r";    char*arg2="-l";

    execlp(file_name,file_name,arg1,arg2,NULL);

}

```

### OUTPUT :

```

● shubham@ShubhsPC:~$ gcc execlp.c
● shubham@ShubhsPC:~$ ./a.out
total 28
-rw-r--r-- 1 shubham shubham  379 Jan 15 13:30 fork.c
-rw-r--r-- 1 shubham shubham  164 Jan 15 13:52 execlp.c
-rw-r--r-- 1 shubham shubham  167 Jan 15 13:51 execl.c
-rwxr-xr-x 1 shubham shubham 15960 Jan 15 13:52 a.out
○ shubham@ShubhsPC:~$ █

```

### 3) execv() & execvp() :

#### a) execv() :

```

#include<stdio.h>

#include<sys/types.h>

#include<unistd.h>

int main()

{

```

```

printf("We are in Execv %d\n",getpid());

char *file[] = {"./subtract",NULL};

execv(file[0],file);

printf("Not Showing Content");

return 0;

}

```

**OUTPUT :**

```

● shubham@ShubhsPC:~$ gcc execv.c -o execv
● shubham@ShubhsPC:~$ gcc subtract.c -o subtract
● shubham@ShubhsPC:~$ ./execv
We are in Execv 11103
46
24
Substraction is : 22
● we are in Subtraction file 11103shubham@ShubhsPC:~$ []

```

• **Substration File :**

```

#include<stdio.h>

#include<unistd.h>

int main()

{

    int a,b,c;

    scanf("%d %d",&a,&b);

    c=a-b;

    printf("Substraction is : %d",c);

    printf("\nwe are in Subtraction file %d",getpid());

}

```

**OUTPUT :**

```
● shubham@ShubhsPC:~$ gcc subtract.c
● shubham@ShubhsPC:~$ ./a.out
48
32
Subtraction is : 16
● we are in Substraction file 1001shubham@ShubhsPC:~$
```

### b) execvp() :

```
#include<stdio.h>

#include<unistd.h>

#include<sys/types.h> int

main()

{

printf("We are in

Execv%d\n",getpid()); char *file[] =

{"../subtract",NULL};

execvp(file[0],file); return 0;

}
```

### OUTPUT :

```
● shubham@ShubhsPC:~$ gcc execvp.c -o execvp
● shubham@ShubhsPC:~$ gcc subtract.c -o subtract
● shubham@ShubhsPC:~$ ./execvp
We are in Execv12031
65
42
Subtraction is : 23
● we are in Substraction file 12031shubham@ShubhsPC:~$
```

### 4) Include fork(), exec() & wait() :

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

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

int main()
{ pid_t pid;
pid =
fork();

if (pid < 0) {
    perror("Fork failed");
}

else if (pid == 0)
{
printf("Child Process Created\n");
printf("Child Process Id: %d, Parent Process Id: %d\n", getpid(), getppid());
printf("We are in Execv%d\n",getpid());
char *file[] = {"./substract",NULL};
execv(file[0],file);
}

else
{
wait();
printf("We are in Parent Process:");
printf("Process Id: %d\n", getpid());
char *file ="/bin/ls";
char *arg1 = "-a";
char *arg2 = "-s";
execl(file,file,arg1,arg2,NULL);
```

```
}
```

```
return 0;
```

```
}
```

## OUTPUT :

```
● shubham@shubhsPC:~$ gcc combine2.c -o combine2
combine2.c: In function ‘main’:
combine2.c:21:9: warning: implicit declaration of function ‘wait’ [-Wimplicit-function-declaration]
  21 |         wait();
     |         ^~~~
● shubham@shubhsPC:~$ gcc subtract.c -o subtract
● shubham@shubhsPC:~$ ./combine2
Child Process Created
Child Process Id: 2175, Parent Process Id: 2174
We are in Execv2175
46
36
Subtraction is : 10
we are in Subtraction file 2175We are in Parent Process:Process Id: 2174
total 156
 4 .           4 .profile          16 execv
 4 ..          0 .sudo_as_admin_successful  4 execv.c
 4 .bash_history 4 .vscode          16 execvp
 4 .bash_logout   4 .vscode-server    4 execvp.c
 4 .bashrc        16 a.out          4 fork.c
 4 .cache         16 combine2       4 sds.txt
 4 .config        4 combine2.c      16 subtract
 4 .dotnet        4 execl.c        4 subtract.c
 0 .motd_shown    4 execlp.c
○ shubham@shubhsPC:~$
```