Operating Systems

UE20CS254

Name: Naman Choudhary	**SRN: PES2UG20CS209**	**Section: D**

Assignment

Program Number	1
Program Qn	Write a program to create a child process which lists all files in the current directory along with the size

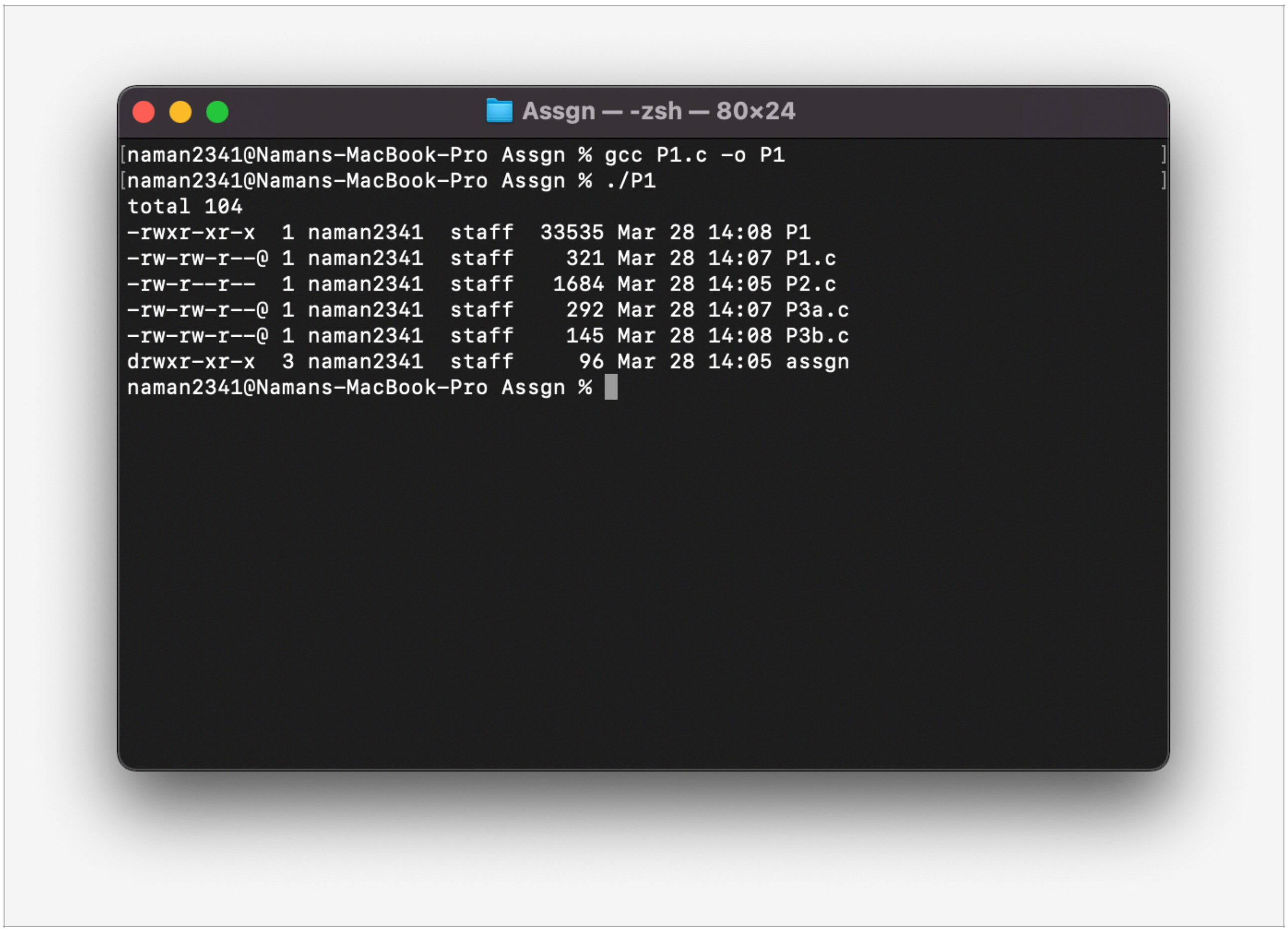
Source Code

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

int main()
{
    pid_t p1;
    p1 = fork();
    if (p1 > 0)
    {
        //parent proc
        wait(NULL); //wait for child proc to exit(0)
    }
    if(p1 == 0)
    {
        execl("/bin/ls", "./", "-l", NULL);
        exit(0);
    }
}

//We are using wait(NULL) that keeps the parent process
from not finishing until the child process return a 0
signal
```

Output Screenshot



```
[naman2341@Namans-MacBook-Pro Assgn % gcc P1.c -o P1                            ]
[naman2341@Namans-MacBook-Pro Assgn % ./P1                                      ]
total 104
-rwxr-xr-x  1 naman2341  staff  33535 Mar 28 14:08 P1
-rw-rw-r--@ 1 naman2341  staff    321 Mar 28 14:07 P1.c
-rw-r--r--  1 naman2341  staff   1684 Mar 28 14:05 P2.c
-rw-rw-r--@ 1 naman2341  staff    292 Mar 28 14:07 P3a.c
-rw-rw-r--@ 1 naman2341  staff    145 Mar 28 14:08 P3b.c
drwxr-xr-x  3 naman2341  staff     96 Mar 28 14:05 assgn
naman2341@Namans-MacBook-Pro Assgn % 
```

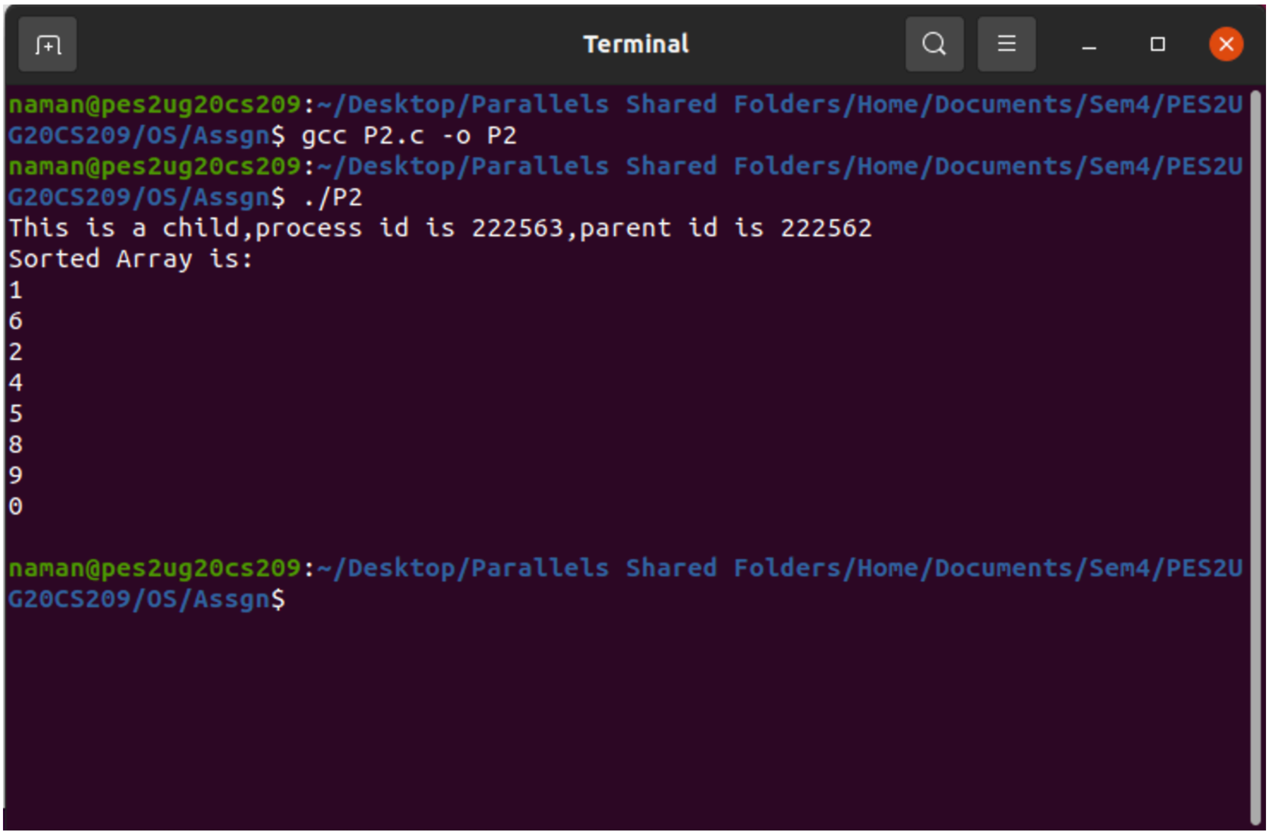
Program Number	2
Program Qn	Create a global array with values 1,6,2,4,5,8,9,0; sort the same with a child process a display the values in the parent process.
Source Code	

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

#define SIZE 25
#define READ 0
#define WRITE 1
int write_msg[SIZE] = {1,6,2,4,5,8,9,0};
int main()
{
    int fd[2];
    pid_t pid;
    pid = fork();
    if(pid ==0)
    {
        close(fd[READ]);
        close(fd[WRITE]);
        wait(NULL);
        printf("This is a child,process id is %d,parent id is %d\n",getpid(),getppid());
        printf("Sorted Array is:\n");
        for(int i=0;i<8;i++)
        {
            printf("%d\n",write_msg[i]);
        }
        printf("\n");
    }
    else if(pid>0)
    {
        close(fd[WRITE]);
        int i;
        for(i=0;i<0;i++)
        {
            for(int j=i+1;j<8;j++)
            {
                if(write_msg[i]>write_msg[j])
                {
                    int temp = write_msg[i];
                    write_msg[i] = write_msg[j];
                    write_msg[j] = temp;
                }
            }
            printf("this is a parent ,process id is %d\n",getpid());
            printf("Sorted Array is:\n");
            for(int k=0;k<8;k++)
            {
                printf("%d\n",write_msg[k]);
            }
            printf("\n");
            close(fd[READ]);
        }
    }
    else
    {
        printf("Fork failed\n");
        return 0;
    }
}

//The reason for it not sorting is that we are sorting it in child but since memoery is not being shared with parent process, the output is not sorrted
```

Output Screenshot



```
naman@pes2ug20cs209:~/Desktop/Parallels Shared Folders/Home/Documents/Sem4/PES2UG20CS209/OS/Assgn$ gcc P2.c -o P2
naman@pes2ug20cs209:~/Desktop/Parallels Shared Folders/Home/Documents/Sem4/PES2UG20CS209/OS/Assgn$ ./P2
This is a child,process id is 222563,parent id is 222562
Sorted Array is:
1
6
2
4
5
8
9
0

naman@pes2ug20cs209:~/Desktop/Parallels Shared Folders/Home/Documents/Sem4/PES2UG20CS209/OS/Assgn$
```

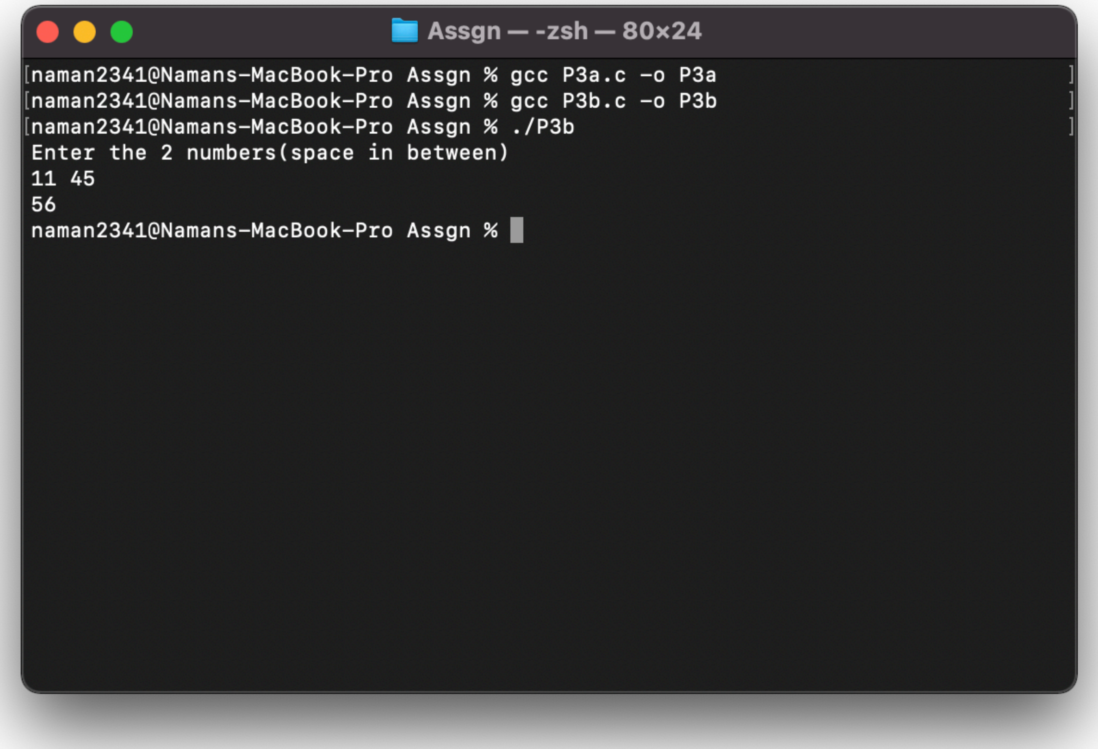
Program Number	3
Program Qn	WAP which accepts 2 integers x and y, use exec() to execute another user defined program that prints the sum of x and y.

Source Code

```c
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
int sum(int x,int y)
{
    return x+y;
}
int main()
{
    int res;
    int x,y;
    printf("Enter the 2 numbers(space in between)\n");
    scanf("%d %d",&x,&y);
    res=sum(x,y);
    printf("%d\n",res);
    return 0;
}
```

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
int sum(int x,int y)
{
    return x+y;
}
int main()
{
    int res;
    int x,y;
    printf("Enter the 2 numbers(space in between)\n");
    scanf("%d %d",&x,&y);
    res=sum(x,y);
    printf("%d\n",res);
    return 0;
}
```

Output Screenshot



```
[naman2341@Namans-MacBook-Pro Assgn % gcc P3a.c -o P3a                          ]
[naman2341@Namans-MacBook-Pro Assgn % gcc P3b.c -o P3b                          ]
[naman2341@Namans-MacBook-Pro Assgn % ./P3b                                     ]
Enter the 2 numbers(space in between)
11 45
56
naman2341@Namans-MacBook-Pro Assgn % 
```