Assignment 2a: Using pipe()

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
main()
{
pid_t pid;
int pfd[2];
int i,j,flg,f1,f2,f3;
static unsigned int ar[25],br[25];
if(pipe(pfd) == -1)
{
printf("Error in
pipe"); exit(-1);
}
pid=fork();
if (pid == 0)
{
printf("Child process generates Fibonacci series\n"
);
f1 = -1;
f2 = 1;
for(i = 0;i < 25; i++)
{
f3 = f1 + f2;
printf("%d\t",f3);
f1 = f2;
f2 = f3;
ar[i] = f3;
}
write(pfd[1],ar,25*sizeof(int));
}
else if (pid > 0)
{
wait(NULL);
read(pfd[0], br, 25*sizeof(int));
printf("\nParent prints Fibonacci that are Prime\n");for(i = 0;i < 25; i++)
{
flg = 0;
if (br[i] <=1)
flg = 1;
for(j=2; j<=br[i]/2; j++)
{
if (br[i]%j == 0)
{
flg=1;
break;
}
}
if (flg == 0)
printf("%d\t", br[i]);
```

```
}
printf("\n");
}
else
{
printf("Process creation
failed"); exit(-1);
}
}
```

Output

```
$ gcc fibprime.c
$ ./a.out
Child process generates Fibonacci series
0
1
1
2
3
21
34
55
89
144
987
1597
2584
4181
6765
5
233
10946
8
377
17711
13
610
28657
46368
Parent prints Fibonacci that are Prime
2
3
5
13
89
233
1597
28657
```