NAME: ASJID TAHIR

ROLL NO: 19P-0085

SECTION: BSE-A

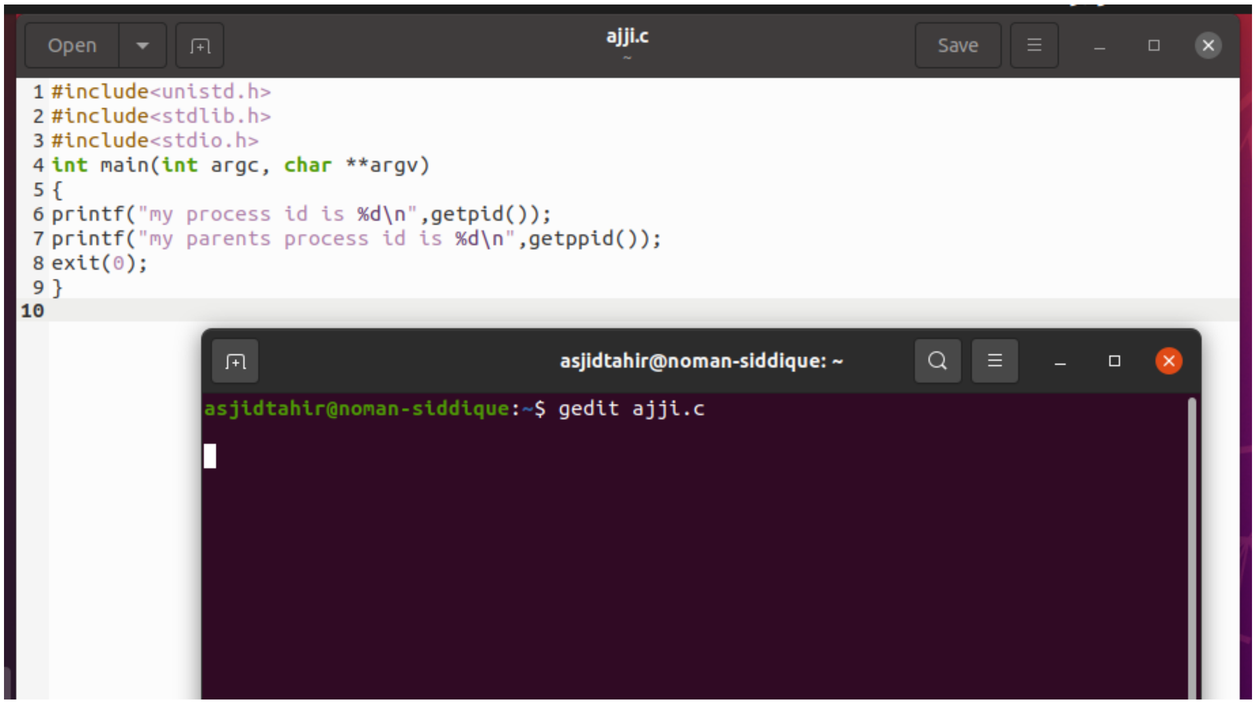
COURSE: OPERATING SYSTEM

INSTRUCTOR: DR.NOUMAN

ASSIGNMENT # 02:

FORK SYSTEM CALL:

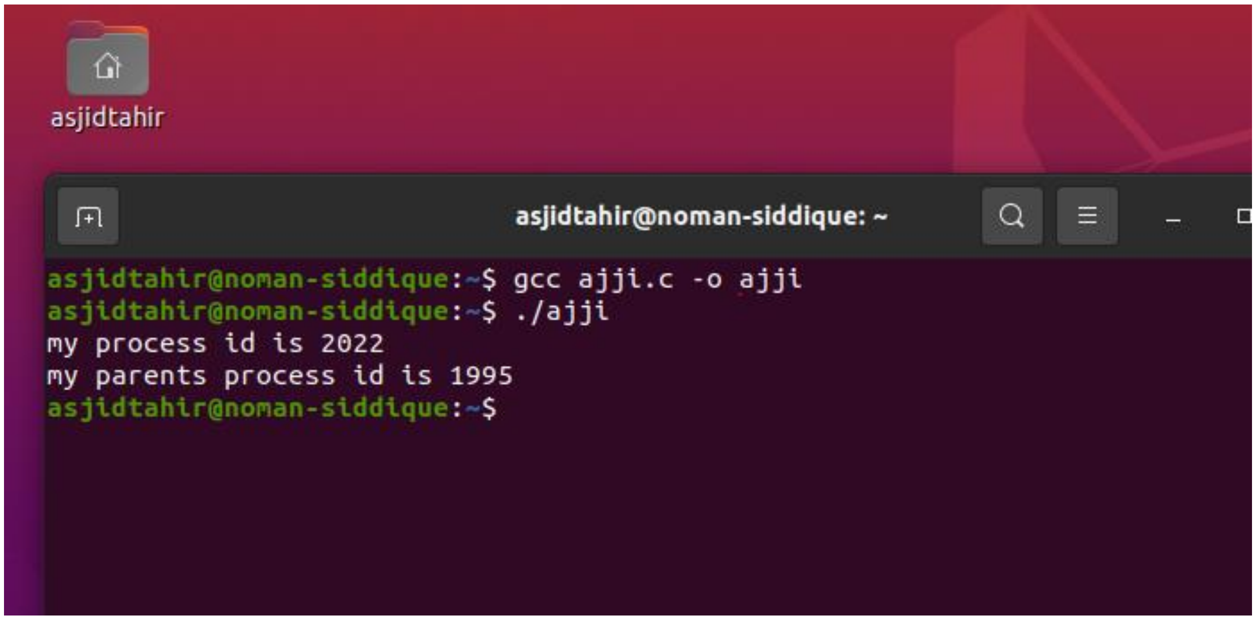
QUESTION NO 01:



```c
#include<unistd.h>
#include<stdlib.h>
#include<stdio.h>
int main(int argc, char **argv)
{
printf("my process id is %d\n",getpid());
printf("my parents process id is %d\n",getppid());
exit(0);
}
```

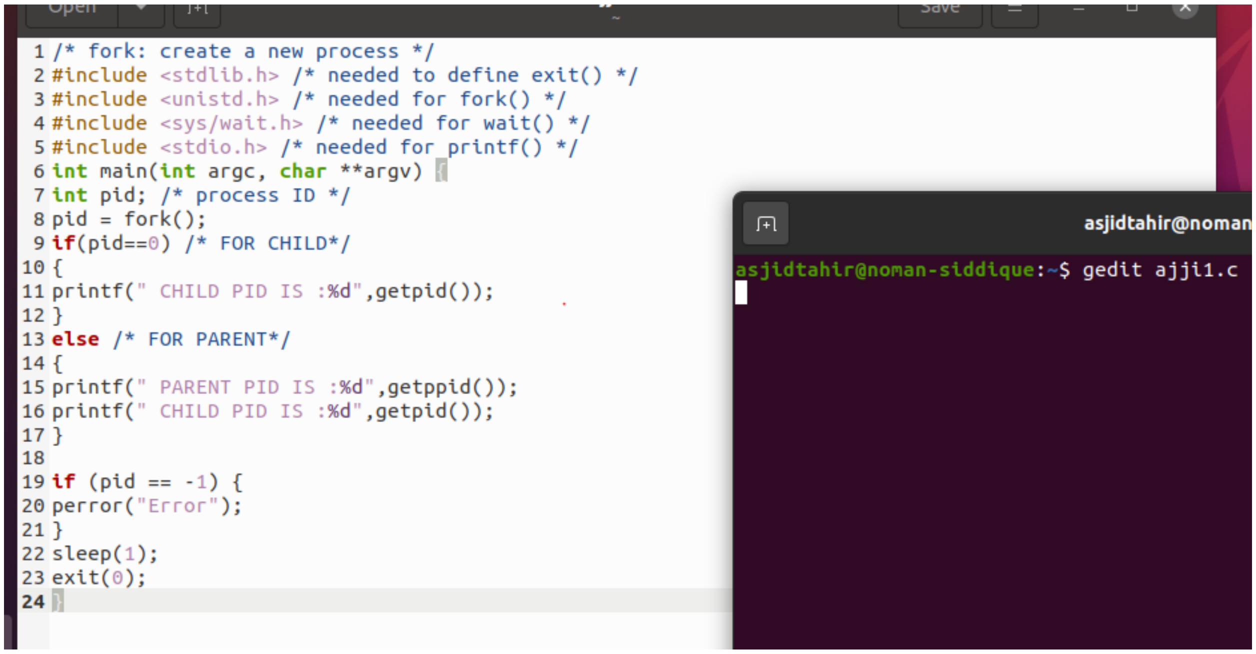
```
asjidtahir@noman-siddique:~$ gedit ajji.c
```

OUTPUT:



```
asjidtahir@noman-siddique:~$ gcc ajji.c -o ajji
asjidtahir@noman-siddique:~$ ./ajji
my process id is 2022
my parents process id is 1995
asjidtahir@noman-siddique:~$
```

QUESTION NO 02:

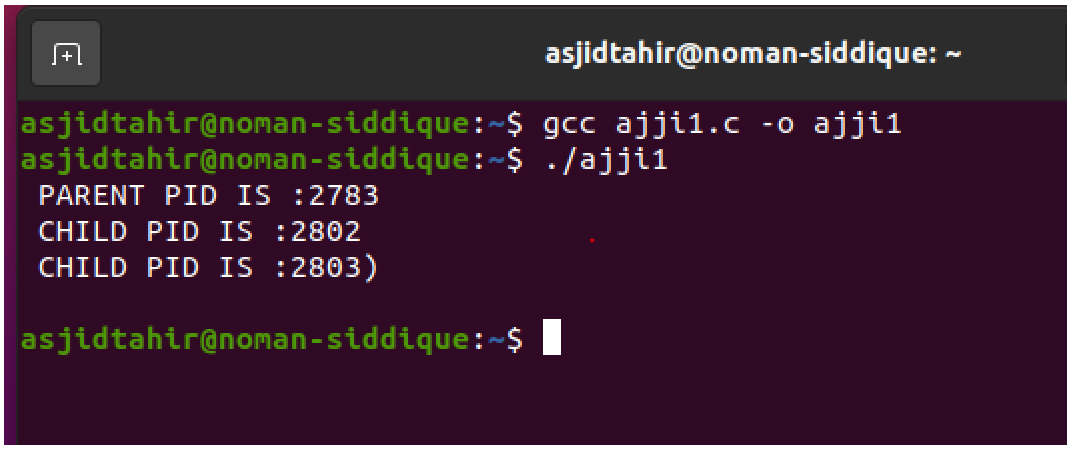


```c
/* fork: create a new process */
#include <stdlib.h> /* needed to define exit() */
#include <unistd.h> /* needed for fork() */
#include <sys/wait.h> /* needed for wait() */
#include <stdio.h> /* needed for printf() */
int main(int argc, char **argv) {
int pid; /* process ID */
pid = fork();
if(pid==0) /* FOR CHILD*/
{
printf(" CHILD PID IS :%d",getpid());
}
else /* FOR PARENT*/
{
printf(" PARENT PID IS :%d",getppid());
printf(" CHILD PID IS :%d",getpid());
}

if (pid == -1) {
perror("Error");
}
sleep(1);
exit(0);
}
```

```
asjidtahir@noman-siddique:~$ gedit ajji1.c
```

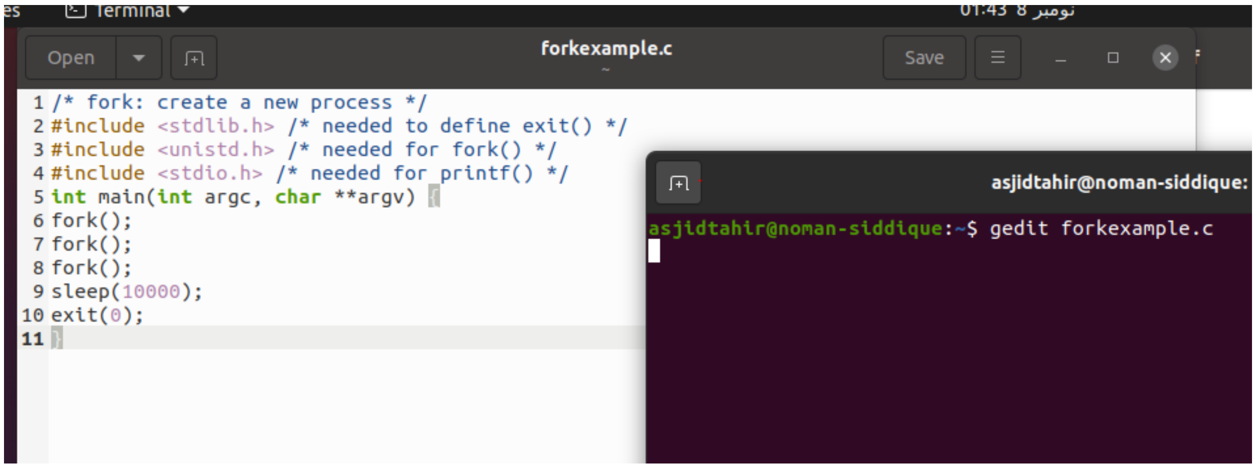
OUTPUT:



```
asjidtahir@noman-siddique:~$ gcc ajji1.c -o ajji1
asjidtahir@noman-siddique:~$ ./ajji1
 PARENT PID IS :2783
 CHILD PID IS :2802
 CHILD PID IS :2803)

asjidtahir@noman-siddique:~$
```

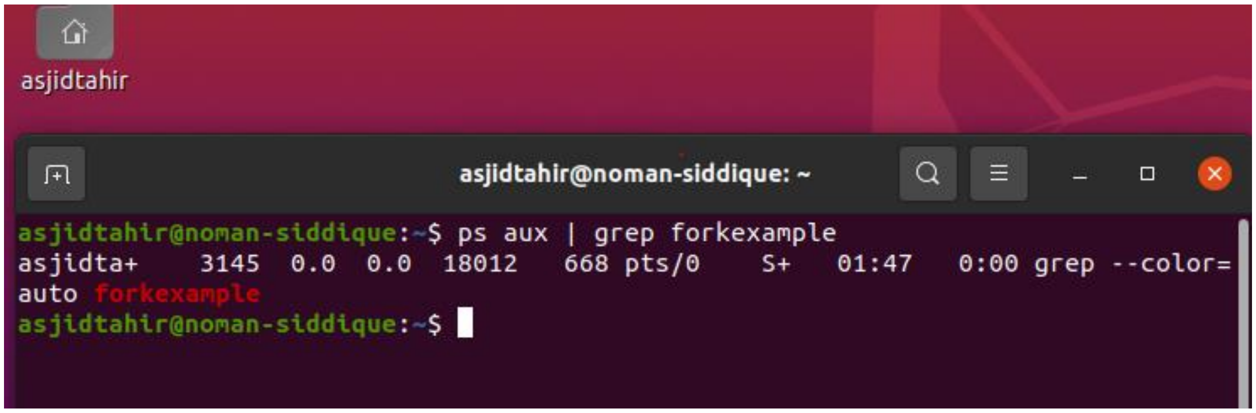
QUESTION NO 03:



```c
/* fork: create a new process */
#include <stdlib.h> /* needed to define exit() */
#include <unistd.h> /* needed for fork() */
#include <stdio.h> /* needed for printf() */
int main(int argc, char **argv) {
fork();
fork();
fork();
sleep(10000);
exit(0);
}
```

```
asjidtahir@noman-siddique:~$ gedit forkexample.c
```

OUTPUT:



```
asjidtahir@noman-siddique:~$ ps aux | grep forkexample
asjidta+    3145  0.0  0.0  18012   668 pts/0    S+   01:47   0:00 grep --color=
auto forkexample
asjidtahir@noman-siddique:~$ 
```