CS342 – Operating Systems Lab

Assignment-4

Tarusi Mittal 1901CS65

1. Write a program in C, which takes n, as an user input, and create n number of zombie processes. Show that the created processes are zombie processes (ADD SCREENSHOT)

Ans:

Compilation: gcc q1.c -o q1

Syntax: ./q1

{the number n }

Here the number n is a user fed input and not the command line argument

Therefore you should press enter after ./q1

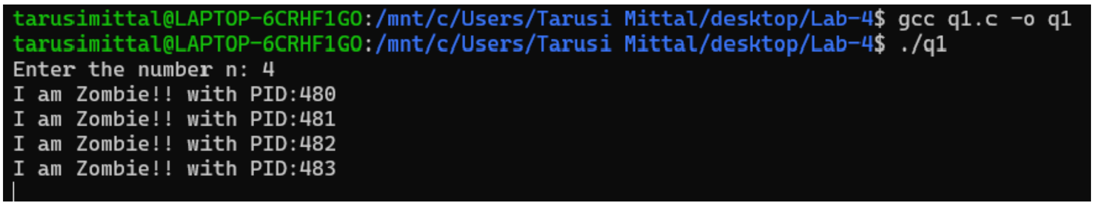
Examples of Execution:

Input: gcc q1.c -o q1

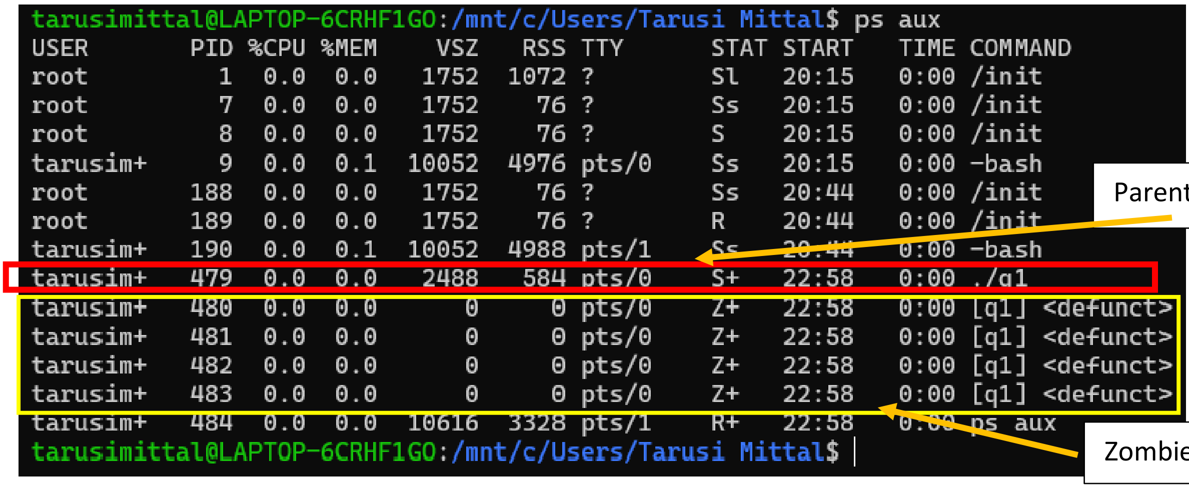
./q1

Enter the number n: 4

Initial Screenshot of execution



```
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ gcc q1.c -o q1
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q1
Enter the number n: 4
I am Zombie!! with PID:480
I am Zombie!! with PID:481
I am Zombie!! with PID:482
I am Zombie!! with PID:483
```

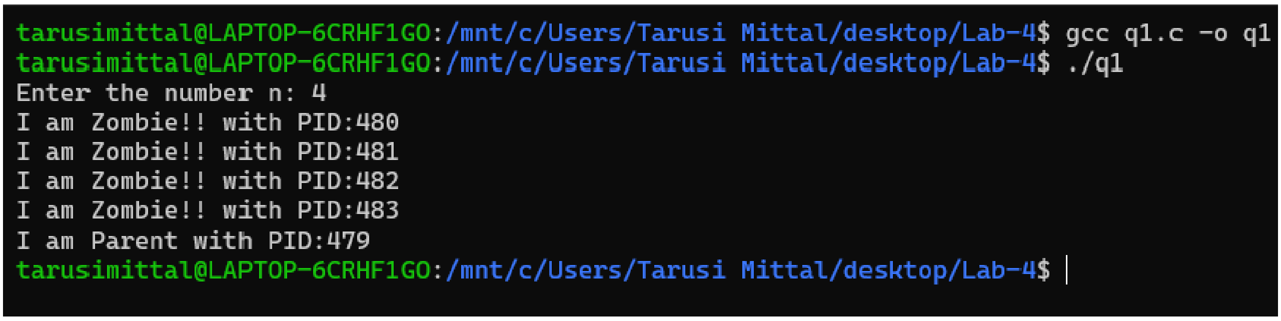


```
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal$ ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   1752  1072 ?        Sl   20:15   0:00 /init
root         7  0.0  0.0   1752    76 ?        Ss   20:15   0:00 /init
root         8  0.0  0.0   1752    76 ?        S    20:15   0:00 /init
tarusim+     9  0.0  0.1  10052  4976 pts/0    Ss   20:15   0:00 -bash
root       188  0.0  0.0   1752    76 ?        Ss   20:44   0:00 /init
root       189  0.0  0.0   1752    76 ?        R    20:44   0:00 /init
tarusim+   190  0.0  0.1  10052  4988 pts/1    Ss   20:44   0:00 -bash
tarusim+   479  0.0  0.0   2488   584 pts/0    S+   22:58   0:00 ./q1
tarusim+   480  0.0  0.0      0     0 pts/0    Z+   22:58   0:00 [q1] <defunct>
tarusim+   481  0.0  0.0      0     0 pts/0    Z+   22:58   0:00 [q1] <defunct>
tarusim+   482  0.0  0.0      0     0 pts/0    Z+   22:58   0:00 [q1] <defunct>
tarusim+   483  0.0  0.0      0     0 pts/0    Z+   22:58   0:00 [q1] <defunct>
tarusim+   484  0.0  0.0  10616  3328 pts/1    R+   22:58   0:00 ps aux
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal$
```

Parent process

Zombie processes

Terminal after 20 Seconds



```
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ gcc q1.c -o q1
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q1
Enter the number n: 4
I am Zombie!! with PID:480
I am Zombie!! with PID:481
I am Zombie!! with PID:482
I am Zombie!! with PID:483
I am Parent with PID:479
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ |
```

---

2. Write a program in C, which takes n, as an user input, and create n number of orphan processes.

Ans:

Compilation: gcc q2.c -o q2

Syntax: ./q2

{the number n }

Here the number n is a user fed input and not the command line argument

Therefore you should press enter after ./q2

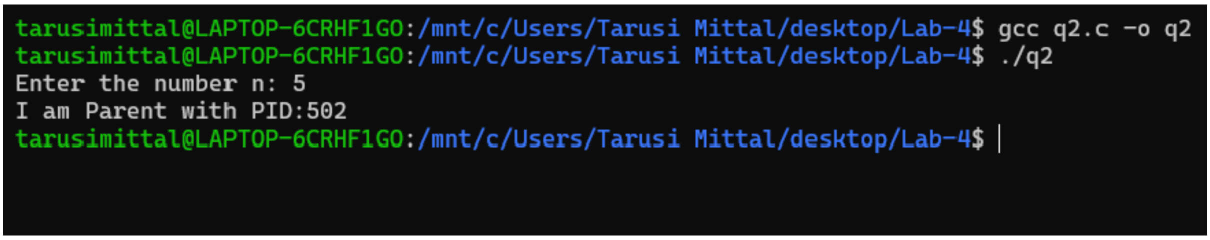
Examples of Execution:

Input: gcc q2.c -o q2

./q2

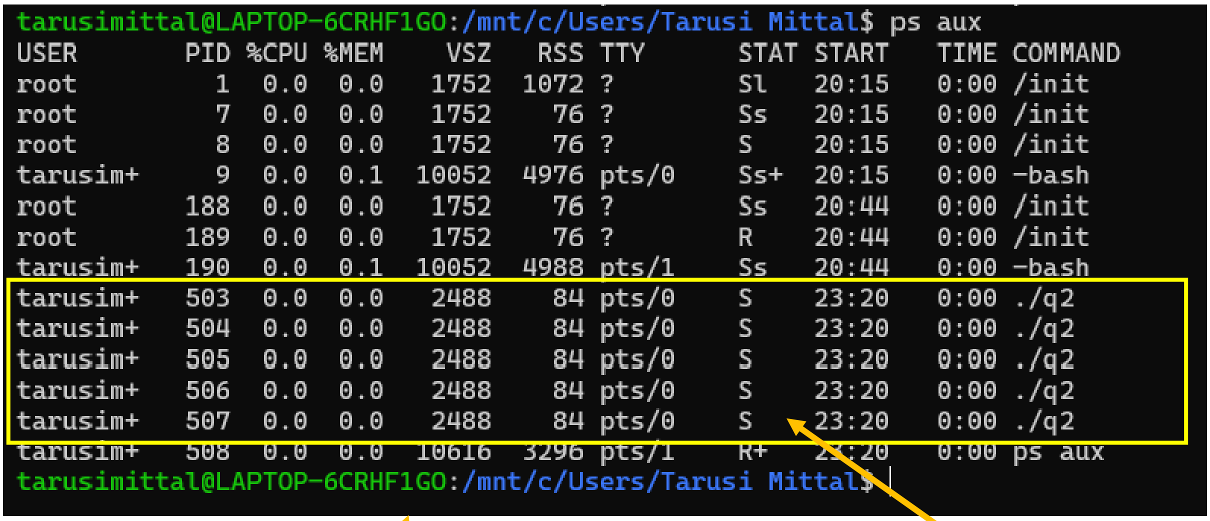
Enter the number n: 5

Initial Screenshot of execution



```
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ gcc q2.c -o q2
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q2
Enter the number n: 5
I am Parent with PID:502
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ |
```

P.T.O

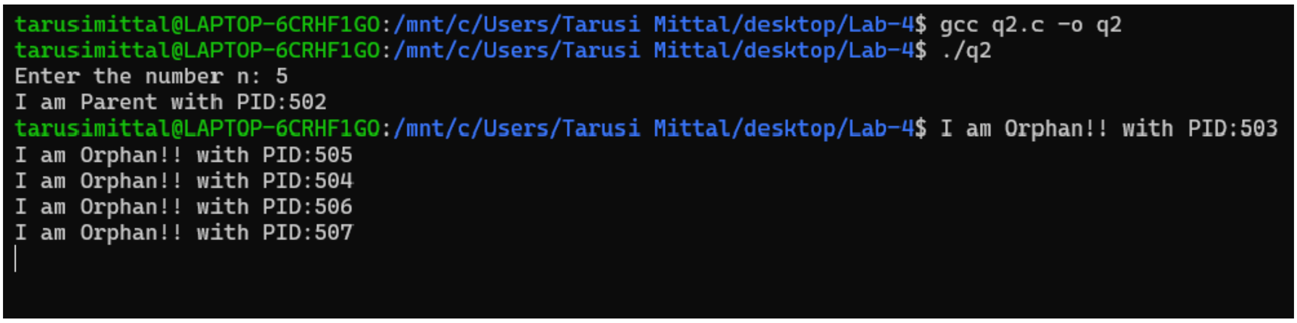


```
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal$ ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   1752  1072 ?        Sl   20:15   0:00 /init
root         7  0.0  0.0   1752    76 ?        Ss   20:15   0:00 /init
root         8  0.0  0.0   1752    76 ?        S    20:15   0:00 /init
tarusim+     9  0.0  0.1  10052  4976 pts/0    Ss+  20:15   0:00 -bash
root       188  0.0  0.0   1752    76 ?        Ss   20:44   0:00 /init
root       189  0.0  0.0   1752    76 ?        R    20:44   0:00 /init
tarusim+   190  0.0  0.1  10052  4988 pts/1    Ss   20:44   0:00 -bash
tarusim+   503  0.0  0.0   2488    84 pts/0    S    23:20   0:00 ./q2
tarusim+   504  0.0  0.0   2488    84 pts/0    S    23:20   0:00 ./q2
tarusim+   505  0.0  0.0   2488    84 pts/0    S    23:20   0:00 ./q2
tarusim+   506  0.0  0.0   2488    84 pts/0    S    23:20   0:00 ./q2
tarusim+   507  0.0  0.0   2488    84 pts/0    S    23:20   0:00 ./q2
tarusim+   508  0.0  0.0  10616  3296 pts/1    R+   23:20   0:00 ps aux
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal$
```

Orphan processes

The parent process is not there as it is already completed

Terminal after 20 seconds:



```
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ gcc q2.c -o q2
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q2
Enter the number n: 5
I am Parent with PID:502
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ I am Orphan!! with PID:503
I am Orphan!! with PID:505
I am Orphan!! with PID:504
I am Orphan!! with PID:506
I am Orphan!! with PID:507
```

Here we can see that the parent process was first completed and after that the command line reappears for taking inputs and commands and after 20 seconds(the time for the child process to sleep) the child processes which were running in the background gets completed and are printed.

P.T.O

3.Write a program, which will-
- Take a user input N
- Make 2 child processes.
- one child process should generate first N LUCAS sequence ([https://en.wikipedia.org/wiki/Lucas_number).](https://en.wikipedia.org/wiki/Lucas_number)
- The 2nd child process should only print the LUCAS sequence (NOT GENERATE) (USE FILE OR SOME OTHER METHOD TO SHARE)

Ans:

Compilation: gcc q3.c -o q3

Syntax: ./q3

{the number n }

Here the number n is a user fed input and not the command line argument

Therefore you should press enter after ./q3

Examples of Execution:

Input: gcc q3.c -o q3

./q3

Enter the number n: 10

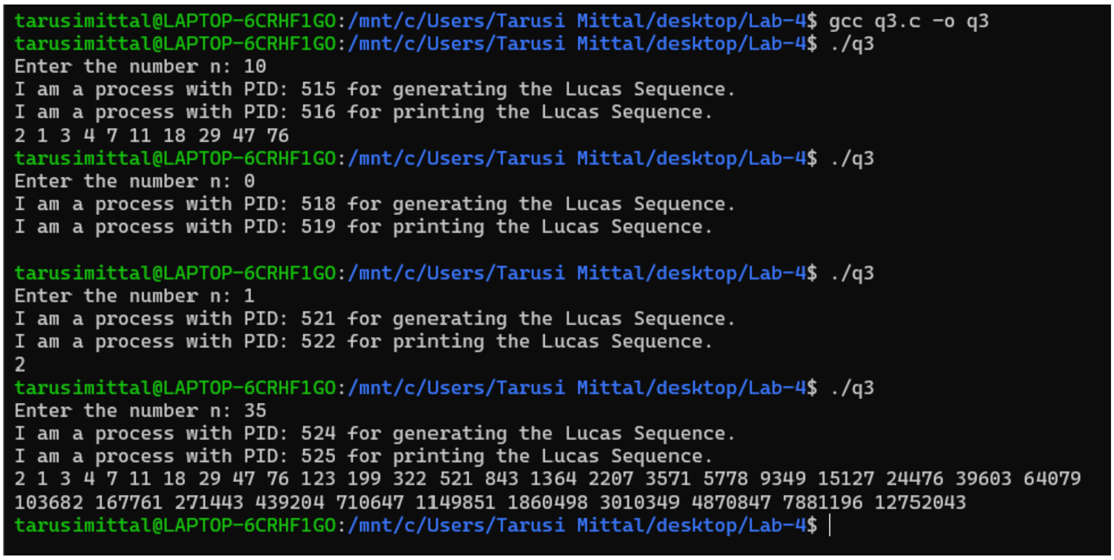
Result:

I am a process with PID: 515 for generating the Lucas Sequence.

I am a process with PID: 516 for printing the Lucas Sequence.

2 1 3 4 7 11 18 29 47 76

For other inputs and results see screenshot attached:



```
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ gcc q3.c -o q3
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q3
Enter the number n: 10
I am a process with PID: 515 for generating the Lucas Sequence.
I am a process with PID: 516 for printing the Lucas Sequence.
2 1 3 4 7 11 18 29 47 76
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q3
Enter the number n: 0
I am a process with PID: 518 for generating the Lucas Sequence.
I am a process with PID: 519 for printing the Lucas Sequence.

tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q3
Enter the number n: 1
I am a process with PID: 521 for generating the Lucas Sequence.
I am a process with PID: 522 for printing the Lucas Sequence.
2
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q3
Enter the number n: 35
I am a process with PID: 524 for generating the Lucas Sequence.
I am a process with PID: 525 for printing the Lucas Sequence.
2 1 3 4 7 11 18 29 47 76 123 199 322 521 843 1364 2207 3571 5778 9349 15127 24476 39603 64079
103682 167761 271443 439204 710647 1149851 1860498 3010349 4870847 7881196 12752043
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ |
```

4. Write a program which will-

- Make 3 threads:

- One will copy the source program to another file f2.

- second will print contents of f2.

- Third will delete the file f2.

Ans:

Compilation: gcc q4.c -o q4

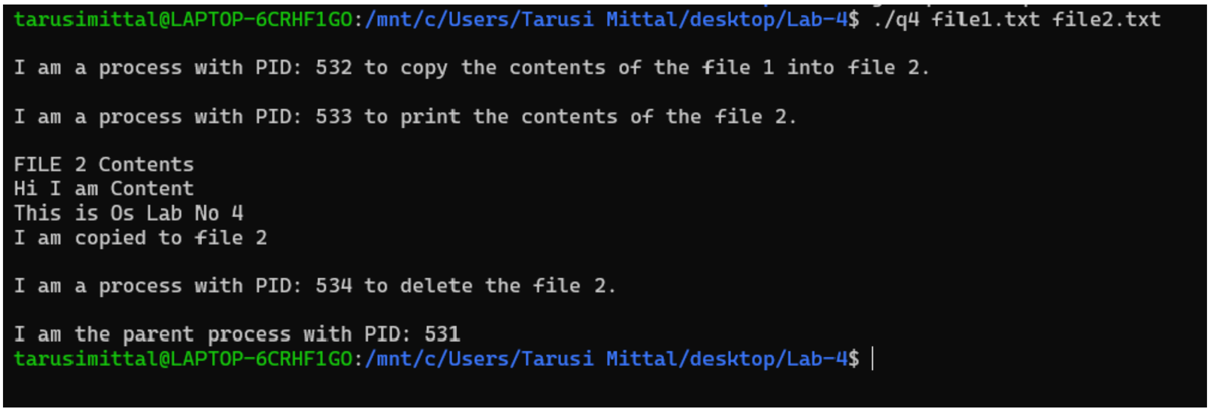
Syntax: ./q4 {name of file 1} {name of file 2 }

Examples of Execution:

Input: gcc q4.c -o q4

./q4 file1.txt file2.txt

Result:



```
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ ./q4 file1.txt file2.txt

I am a process with PID: 532 to copy the contents of the file 1 into file 2.

I am a process with PID: 533 to print the contents of the file 2.

FILE 2 Contents
Hi I am Content
This is Os Lab No 4
I am copied to file 2

I am a process with PID: 534 to delete the file 2.

I am the parent process with PID: 531
tarusimittal@LAPTOP-6CRHF1GO:/mnt/c/Users/Tarusi Mittal/desktop/Lab-4$ |
```

END