

PLAGIARISM STATEMENT

I certify that this assignment/report is my own work, based on my personal study and/or research and that I have acknowledged all material and sources used in its preparation, whether they be books, articles, reports, lecture notes, and any other kind of document, electronic or personal communication. I also certify that this assignment/report has not previously been submitted for assessment in any other course, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that I have not copied in part or whole or otherwise plagiarised the work of other students and/or persons. I pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, I understand my responsibility to report honour violations by other students if I become aware of it.

Name: A. Venkata Sai Mahesh

Date: 03/11/2019

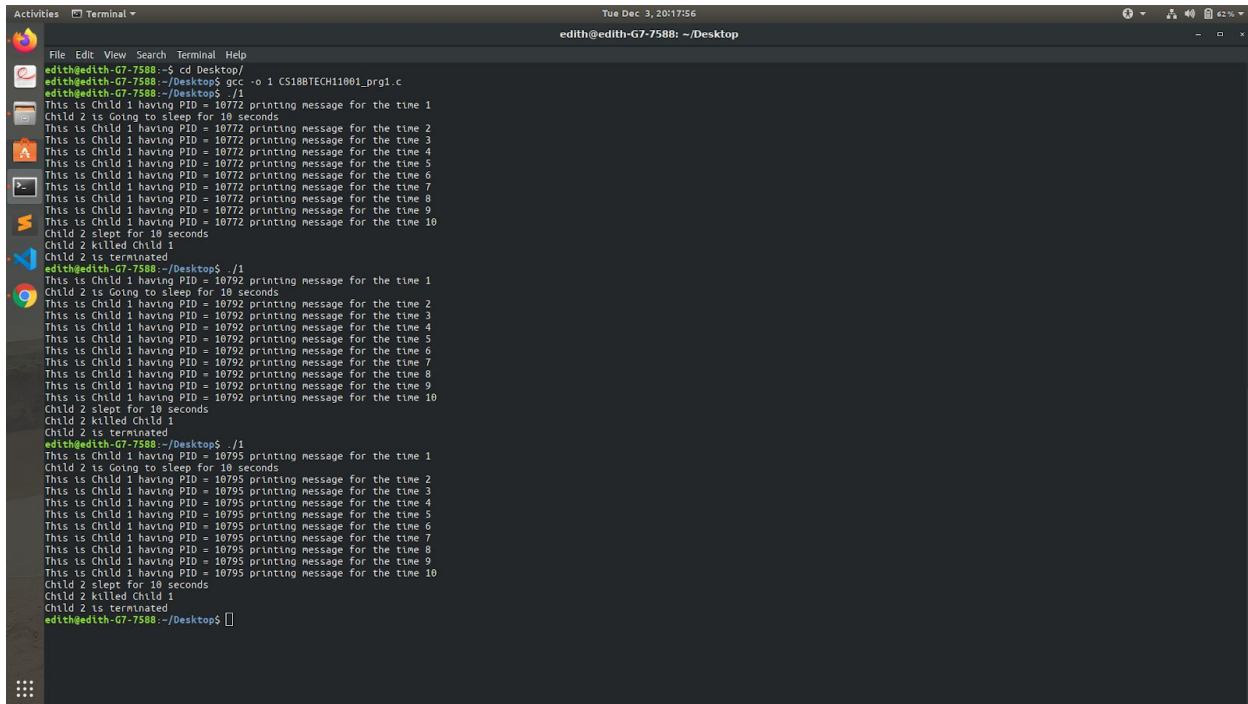
Signature: Venkata Sai Mahesh Abburi

Program 1 :

- First I used a fork() and created a first Child process.
- By checking the pid I used next fork() only to the Parent process such that the Child process will be created only to the Parent.
- Now I use the sleep function for child 2 and the child 1 continues to print the message until child 2 wakes up.
- When the child 2 wakes up I used a kill Signal to kill the Child 1 by passing the pid of child 1 stored in the pid1 variable of Parent.
- Then I again used sleep function for Child 2.
- Then I used waitpid function in Parent with arguments as pid of Child 2 which waits until the child 2 get terminated.
- Then I print message that child 2 has been terminated in the Parent process.
- After that I used return 0 to terminate parent process.

Sample Outputs:

1. This is Child 1 having PID = 6082 printing message for the time 1
Child 2 is Going to sleep for 10 seconds
This is Child 1 having PID = 6082 printing message for the time 2
This is Child 1 having PID = 6082 printing message for the time 3
This is Child 1 having PID = 6082 printing message for the time 4
This is Child 1 having PID = 6082 printing message for the time 5
This is Child 1 having PID = 6082 printing message for the time 6
This is Child 1 having PID = 6082 printing message for the time 7
This is Child 1 having PID = 6082 printing message for the time 8
This is Child 1 having PID = 6082 printing message for the time 9
This is Child 1 having PID = 6082 printing message for the time 10
Child 2 slept for 10 seconds
Child 2 killed Child 1
Child 2 is terminated



```
Activities Terminal
Tue Dec 3, 20:17:56
edith@edith-G7-7588: ~/Desktop
edith@edith-G7-7588:~$ cd Desktop/
edith@edith-G7-7588:~/Desktop$ gcc -o 1 CS18BTECH11001_prg1.c
edith@edith-G7-7588:~/Desktop$ ./1
This is Child 1 having PID = 10772 printing message for the time 1
Child 2 is going to sleep for 10 seconds
This is Child 1 having PID = 10772 printing message for the time 2
This is Child 1 having PID = 10772 printing message for the time 3
This is Child 1 having PID = 10772 printing message for the time 4
This is Child 1 having PID = 10772 printing message for the time 5
This is Child 1 having PID = 10772 printing message for the time 6
This is Child 1 having PID = 10772 printing message for the time 7
This is Child 1 having PID = 10772 printing message for the time 8
This is Child 1 having PID = 10772 printing message for the time 9
This is Child 1 having PID = 10772 printing message for the time 10
Child 2 slept for 10 seconds
Child 2 killed Child 1
Child 2 is terminated
edith@edith-G7-7588:~/Desktop$ ./1
This is Child 1 having PID = 10792 printing message for the time 1
Child 2 is going to sleep for 10 seconds
This is Child 1 having PID = 10792 printing message for the time 2
This is Child 1 having PID = 10792 printing message for the time 3
This is Child 1 having PID = 10792 printing message for the time 4
This is Child 1 having PID = 10792 printing message for the time 5
This is Child 1 having PID = 10792 printing message for the time 6
This is Child 1 having PID = 10792 printing message for the time 7
This is Child 1 having PID = 10792 printing message for the time 8
This is Child 1 having PID = 10792 printing message for the time 9
This is Child 1 having PID = 10792 printing message for the time 10
Child 2 slept for 10 seconds
Child 2 killed Child 1
Child 2 is terminated
edith@edith-G7-7588:~/Desktop$ ./1
This is Child 1 having PID = 10795 printing message for the time 1
Child 2 is going to sleep for 10 seconds
This is Child 1 having PID = 10795 printing message for the time 2
This is Child 1 having PID = 10795 printing message for the time 3
This is Child 1 having PID = 10795 printing message for the time 4
This is Child 1 having PID = 10795 printing message for the time 5
This is Child 1 having PID = 10795 printing message for the time 6
This is Child 1 having PID = 10795 printing message for the time 7
This is Child 1 having PID = 10795 printing message for the time 8
This is Child 1 having PID = 10795 printing message for the time 9
This is Child 1 having PID = 10795 printing message for the time 10
Child 2 slept for 10 seconds
Child 2 killed Child 1
Child 2 is terminated
edith@edith-G7-7588:~/Desktop$
```

Program 3 :

This Case converter program converts all the alphabets in the string to Opposite Case.

- I created 2 pipes(pipe1 and pipe2 which are integer array of size 2).
- Then I created a Child Process for the Parent using fork().
- Then I had taken a string as an input from the user using the fgets function in the Parent Process.
- Then I passed the string to the 1st Child process using pipe1.
- Then I called the case converter function which converts the given string to the Uppercase form (which are in lowercase) and lowercase (for which are in Uppercase) using the ASCII characters(As difference Between lowercase letters and uppercase letters is 32).
- Then I passed the String using(another pipe2) to another child Process created using fork() for the Parent.
- Now this child process prints the converted String to the Standard output.
- Finally The parent taken Input from user and send it to 1st Child process which then converts case and send the converted String to 2nd Child Process which then displays on the terminal.

Sample Outputs :

1. Enter the string to be converted : Hello I am Mahesh
The String after converting to Upper case is : hELLO i AM mAHESH
2. Enter the string to be converted : hbfvlawbn hfvbyas hvba
The String after converting to Upper case is : HBFVLAWBN HFVBYAS HVBA

```
Activities Terminal Tue Dec 3, 23:47:17
edith@edith-G7-7588: ~/Desktop
File Edit View Search Terminal Help
edith@edith-G7-7588:~$ cd Desktop/
edith@edith-G7-7588:~/Desktop$ gcc -o 5 CS180TECH11001_prg3.c
edith@edith-G7-7588:~/Desktop$ ./5
Enter the string to be converted : awhfebyagvy hvdathbtia
The String after case flipping is : AWHFEBYAGVY Hvdathbtia
edith@edith-G7-7588:~/Desktop$ ./5
Enter the string to be converted : My number is 9381306441
The String after case flipping is : mY NUMBER IS 9381306441
edith@edith-G7-7588:~/Desktop$ ./5
Enter the string to be converted : .hjefbyagwc bhygqctq
The String after case flipping is : .HJEFBYAGWC bhygqctq
edith@edith-G7-7588:~/Desktop$ ./5
Enter the string to be converted : hbcgk ckyktqwtvd731tr 850
The String after case flipping is : HBCGK CKYKTQWTD731tr 850
edith@edith-G7-7588:~/Desktop$ ./5
Enter the string to be converted : JHBNAEYVWOYGV YTFevqGV67l238ry0
The String after case flipping is : jhbnaeyvwoygv ycfEVQvgv67l238RY0
edith@edith-G7-7588:~/Desktop$
```