

No E-Mail submissions will be accepted.  
Submission formats and file naming:

File name : firstName\_lastName\_lab\_5

File format: pdf or MS Word format

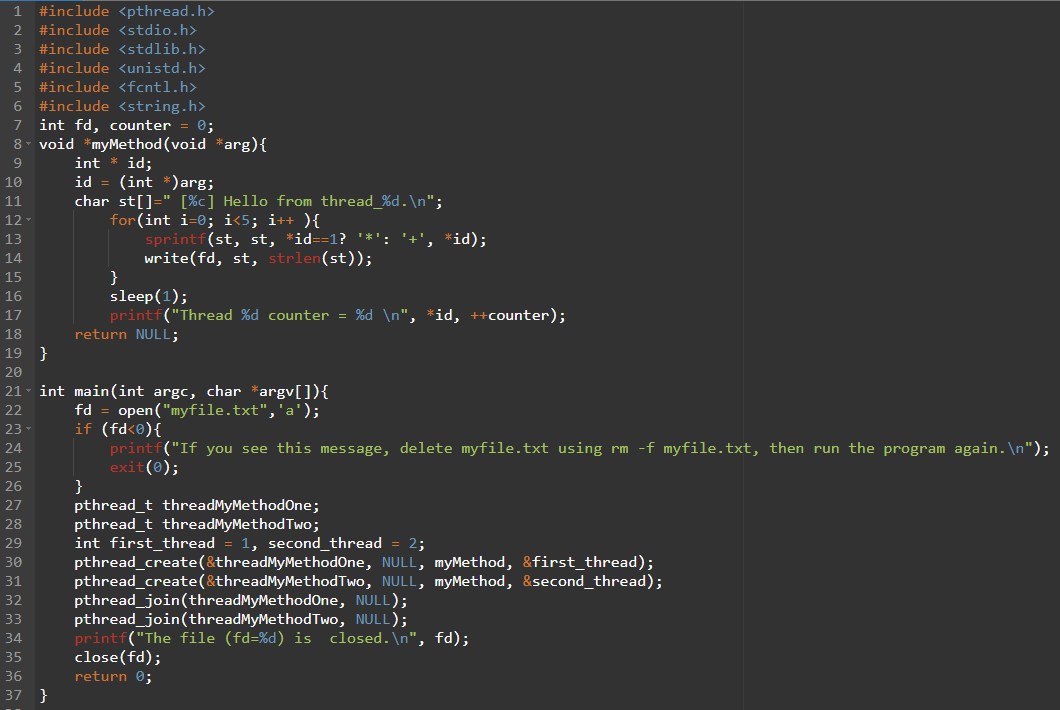
e.g. Donald\_Trump\_lab\_5.pdf

**1)** Use [cocalc.com](https://cocalc.com/) to run the code given below and answer the following questions:





*main1.c*



1. Attach a screenshot of your output.
2. Based on the output of your code, can you provide an explanation on whether the global variable "counter" is shared between threads 1 and 2?
3. After running the program, provide a screenshot showing the contents of myfile.txt. You may need to change the file's permissions using a command like **chmod 444 myfile.txt**.
4. Fromthe contents of myfile.txt, is the file myfile.txt shared between the two threads (yes/no)? Explain your answer.
5. From the output, Obtain the *pid* and *tid* for the process and threads? Attach a screenshot of your output.

|  |  |
| --- | --- |
| Process ID (PID) | ? |
| Thread ID 1 (TID) | ? |
| Thread ID 2 (TID) | ? |

1. Change the sleep time from 1 to 100, then recompile your program and run it using

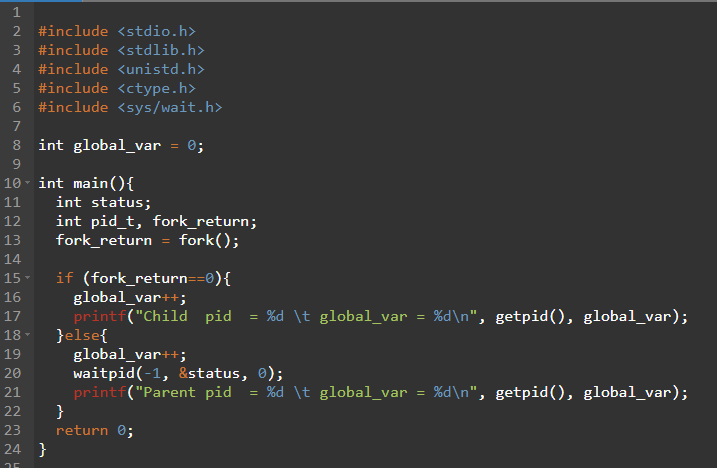


In the second terminal use **kill -9 <tid>** to terminate one of the threads. Does the main process remain alive after the thread is killed? Attach a screenshot of the output after terminating the thread.

**2)** Use the online C compiler

<https://www.onlinegdb.com/online_c_compiler>

to run your code and then answer the following questions.



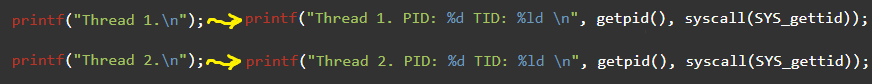
1. From the output, determine the value of the global\_var variable in the parent process.
2. From the output, determine the value of the global\_var variable in the child process.
3. Based on your observations, explain whether the global\_var variable is shared between the parent and child processes.

**3.** Use the online compiler <https://www.onlinegdb.com/online_c_compiler> to run user\_level.c and then answer the following questions.

a) Run the program and observe the output.

1. What do you notice about the interleaving of "Thread 1" and "Thread 2" messages?
2. Which thread starts first?

b) Print the thread IDs inside each thread function.

1. Modify thread\_1() and thread\_2() to print the thread ID using
2. Does each thread have a different thread ID? Why or why not?

c) Change the sleep(1) calls to sleep(100) in both thread functions.

1. What happens to the execution? Does one sleep call block both threads?
2. Explain why this happens in the context of user-level threading.