|  |
| --- |
| // Same thread id is printed in parent and child process. |
|  |  |
|  | #include <stdio.h> |
|  | #include <stdlib.h> |
|  | #include <unistd.h> . |
|  | #include <pthread.h> |
|  |  |
|  | void \*myThread1(void \*vargp) |
|  | { |
|  | int \*myid = (int \*)vargp; |
|  | printf("Thread ID: %d\n", \*myid); |
|  |  |
|  | } |
|  |  |
|  | void \*myThread2(void \*vargp) |
|  | { |
|  | int \*myid = (int \*)vargp; |
|  | printf("Thread ID: %d\n", \*myid); |
|  |  |
|  | } |
|  |  |
|  | int main() |
|  | { |
|  | pthread\_t thread\_1; |
|  | pthread\_t thread\_2; |
|  |  |
|  | printf(" Before Thread 1\n"); |
|  |  |
|  | pthread\_create(&thread\_1, NULL, myThread1, (void \*)&thread\_1); |
|  | pthread\_join(thread\_1, NULL); |
|  |  |
|  | printf("After Thread 1\n"); |
|  |  |
|  | int pid = fork(); |
|  |  |
|  | if(pid==0) |
|  | { |
|  | printf(" In child process \n"); |
|  | printf(" Before Thread 2 \n"); |
|  | printf("pid=%d ,parent=%d\n",getpid(),getppid()); |
|  |  |
|  | pthread\_create(&thread\_2, NULL, myThread2, (void \*)&thread\_2); |
|  | pthread\_join(thread\_2, NULL); |
|  | printf("After Thread 2\n"); |
|  | } |
|  | else{ |
|  | wait(NULL); |
|  | printf(" In Parent process\n"); |
|  | printf(" Before Thread 2 \n"); |
|  | printf("pid=%d ,parent=%d\n",getpid(),getppid()); |
|  |  |
|  | pthread\_create(&thread\_2, NULL, myThread2, (void \*)&thread\_2); |
|  | pthread\_join(thread\_2, NULL); |
|  | printf("After Thread 2\n"); |
|  | } |
|  |  |
|  |  |
|  | return 0; |
|  | } |