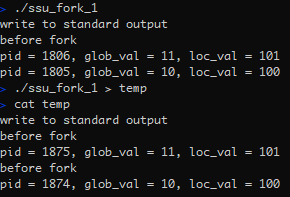
컴퓨터학부 20162448 김병준

1. 결과



1. 소스코드

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <unistd.h>  char glob\_str[] = "write to standard output\n";  int glob\_val= 10;  int main(void)  {  pid\_t pid;  int loc\_val;  loc\_val= 100;  if (write(STDOUT\_FILENO, glob\_str, sizeof(glob\_str)-1) != sizeof(glob\_str) -1) { // 표준출력으로 glob\_str 출력  fprintf(stderr, "write error\n");  exit(1);  }  printf("before fork\n"); // 포킹 전  if ((pid = fork()) < 0) { // 1. 자식의 pid 리턴 및 자식 프로세스 호출  fprintf(stderr, "fork error\n");  exit(1);  }  else if (pid == 0) { // 2. 자식 프로세스는 0을 리턴  glob\_val++;  loc\_val++;  }  else  sleep(3);  printf("pid = %d, glob\_val = %d, loc\_val = %d\n", getpid(), glob\_val, loc\_val); // 3. 자식 프로세스 출력, 4. 부모 프로세스 출력  exit(0);  } |