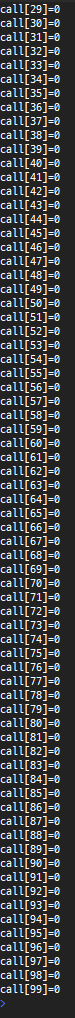
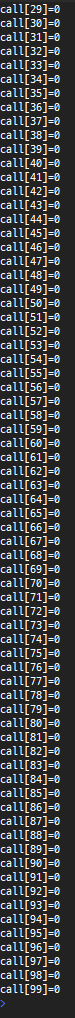
컴퓨터학부 20162448 김병준

1. 결과



1. 소스코드

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <unistd.h>  #include <sys/types.h>  #include <errno.h>  #include <fcntl.h>  #define MAX\_CALL 100  #define MAX\_BUF 20000  void serror(char \*str)  {  fprintf(stderr, "%s\n", str);  exit(1);  }  int main(void)  {  int nread, nwrite, val, i=0;  char \*ptr;  char buf[MAX\_BUF];  int call[MAX\_CALL];  // 표준 입력으로부터 텍스트 읽음  nread = read(STDIN\_FILENO, buf, sizeof(buf));  fprintf(stderr, "read %d bytes\n", nread);  // 표준 출력으로부터 플래그 획득  if ((val = fcntl(STDOUT\_FILENO, F\_GETFL, 0))<0)  serror("fcntl F\_GETFL error");  val|=O\_NONBLOCK; // 플래그 켜기  if(fcntl(STDOUT\_FILENO, F\_SETFL, val)<0) // 플래그 설정  serror("fcntl F\_SETFL error");  for(ptr=buf;nread>0;i++)  {  errno=0;  nwrite=write(STDOUT\_FILENO, ptr, nread); // 표준 출력으로 텍스트 출력  if(i<MAX\_CALL)  call[i]=nwrite;  fprintf(stderr, "nwrite = %d, errno = %d\n", nwrite, errno);  if (nwrite>0)  {  ptr+=nwrite;  nread-=nwrite;  }  }  if ((val=fcntl(STDOUT\_FILENO, F\_GETFL, 0))<0) // 플래그 획득  serror("fcntl F\_GETFL error");  val &= ~O\_NONBLOCK;  if (fcntl(STDOUT\_FILENO, F\_SETFL, val)<0) // 플래그 설정  serror("fcntl F\_SETFL error");  for (i=0;i<MAX\_CALL;i++)  fprintf(stdout, "call[%d]=%d\n", i, call[i]);  exit(0);  } |