Write a C program that creates a processes A, and process A creates a process B. A and B communicate using fifos. Process A sends to process B a number n>50 multiple of 3. Process B divides the number by 3 and sends it back to A . Process A receives the number n and adds to it a value s={0,1,2} such that it is a multiple of 3, then sends it to B. B receives the number n, divides it by 3, and sends it to A and so on until the number is <=5.

#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

#include <sys/stat.h>

#include <sys/wait.h>

#include <sys/types.h>

#include <fcntl.h>

int main(int argc, char\*\*argv)

{

int pidA, pidB;

int A2B, B2A;

pidA = fork();

mkfifo("myfifo", 0666);

if(pidA == 0)

{

pidB = fork();

if(pidB == 0)

{

while(1)

{

int n;

B2A = open("myfifo", O\_RDONLY);

read(B2A, &n, sizeof(int));

close(B2A);

if(n <= 5) break;

int div;

div = n / 3;

B2A = open("myfifo", O\_WRONLY);

write(B2A, &div, sizeof(int));

close(B2A);

}

exit(0);

}

int nr = 124;

A2B = open("myfifo", O\_WRONLY);

write(A2B, &nr, sizeof(int));

close(A2B);

int res;

A2B = open("myfifo", O\_RDONLY);

read(A2B, &res, sizeof(int));

close(A2B);

while(res > 5)

{

if ((res + 1) % 3 == 0)

{

res += 1;

}

else{

if ((res + 2) % 3 == 0)

res += 2;

}

A2B = open("myfifo", O\_WRONLY);

write(A2B, &res, sizeof(int));

close(A2B);

A2B = open("myfifo", O\_RDONLY);

read(A2B, &res, sizeof(int));

close(A2B);

}

A2B = open("myfifo", O\_WRONLY);

write(A2B, &res, sizeof(int));

close(A2B);

printf("res = %d\n", res);

wait(0);

exit(0);

}

wait(0);

return 0;

}