**7A OS 33185**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <fcntl.h>

#include <unistd.h>

#define FIFO1 "pipe1"

#define FIFO2 "pipe2"

int main()

{

char input[1024], output[1024];

mkfifo(FIFO1, 0666);

mkfifo(FIFO2, 0666);

int fd\_write = open(FIFO1, O\_WRONLY);

int fd\_read = open(FIFO2, O\_RDONLY);

if (fd\_write < 0 || fd\_read < 0)

{

printf("Error opening FIFO.\n");

exit(1);

}

printf("Enter a sentence (type 'exit' to stop):\n");

while (1)

{

printf("Input: ");

fgets(input, sizeof(input), stdin);

input[strlen(input) - 1] = '\0';

if (strcmp(input, "exit") == 0)

{

write(fd\_write, input, strlen(input) + 1);

break;

}

write(fd\_write, input, strlen(input) + 1);

read(fd\_read, output, sizeof(output));

printf("Process2 Output: %s\n", output);

}

close(fd\_write);

close(fd\_read);

unlink(FIFO1);

unlink(FIFO2);

return 0;

}

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <fcntl.h>

#include <unistd.h>

#define FIFO1 "pipe1"

#define FIFO2 "pipe2"

#define OUTPUT\_FILE "output.txt"

void countDetails(const char \*str, int \*char\_count, int \*word\_count, int \*line\_count)

{

\*char\_count = \*word\_count = \*line\_count = 0;

int in\_word = 0;

for (int i = 0; str[i] != '\0'; i++)

{

(\*char\_count)++;

if (str[i] == ' ' || str[i] == '\n' || str[i] == '\t')

{

in\_word = 0;

} else if (!in\_word)

{

in\_word = 1;

(\*word\_count)++;

}

if (str[i] == '\n')

{ (\*line\_count)++;

}

}

}

int main()

{

char input[1024], result[1024];

int char\_count, word\_count, line\_count;

mkfifo(FIFO1, 0666);

mkfifo(FIFO2, 0666);

int fd\_read = open(FIFO1, O\_RDONLY);

int fd\_write = open(FIFO2, O\_WRONLY);

if (fd\_read < 0 || fd\_write < 0)

{

printf("Error opening FIFO.\n");

exit(1);

}

FILE \*output\_file;

while (1){

read(fd\_read, input, sizeof(input));

if (strcmp(input, "exit") == 0)

{

break;

}

countDetails(input, &char\_count, &word\_count, &line\_count);

output\_file = fopen(OUTPUT\_FILE, "w");

fprintf(output\_file, "Characters: %d, Words: %d, Lines: %d\n", char\_count, word\_count, line\_count);

fclose(output\_file);

snprintf(result, sizeof(result), "Characters: %d, Words: %d, Lines: %d", char\_count, word\_count, line\_count);

write(fd\_write, result, strlen(result) + 1);

}

close(fd\_read);

close(fd\_write);

unlink(FIFO1);

unlink(FIFO2);

return 0;

}

Test Case 1:

Input (in process1):

Hello, World!

Output (from process2):

Process2 Output: Characters: 13, Words: 2, Lines: 0

Test Case 2:

Input (in process1):

This is a test sentence.

Output (from process2):

Process2 Output: Characters: 24, Words: 5, Lines: 0

Test Case 3:

Input (in process1):

A\nB\nC

Output (from process2):

Process2 Output: Characters: 5, Words: 3, Lines: 2