## **STREAMS message/PIPEs/FIFO:pipe, popenand pcloseFunctions**

**Assignment No:** 13d

**Objectives:**

1.Invoke uppercase/lowercase filter to read commands. (I)

**Theory**:

One thing that popen is especially well suited for is executing simple filters to transform the input or output of the running command. Such is the case when a command wants to build its own pipeline.

Consider an application that writes a prompt to standard output and reads a line from standard input. With the popen function, we can interpose a program between the application and its input to transform the input. The following figure shows the arrangement of processes in this situation.

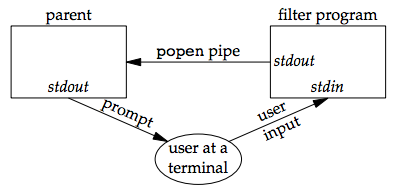


Figure 15.13 Transforming input using popen

The following program is a simple filter to demonstrate this operation:

The filter copies standard input to standard output, converting any uppercase character to lowercase. The reason we’re careful to fflush standard output after writing a newline is discussed in the next section when we talk about coprocesses.

We compile this filter into the executable file myuclc, which we then invoke from the program in the following code using popen:

#include "apue.h"

#include <sys/wait.h>

int

main(void)

{

char line[MAXLINE];

FILE \*fpin;

if ((fpin = popen("myuclc", "r")) == NULL)

err\_sys("popen error");

for ( ; ; ) {

fputs("prompt> ", stdout);

fflush(stdout);

if (fgets(line, MAXLINE, fpin) == NULL) /\* read from pipe \*/

break;

if (fputs(line, stdout) == EOF)

err\_sys("fputs error to pipe");

}

if (pclose(fpin) == -1)

err\_sys("pclose error");

putchar('\n');

exit(0);

}

We need to call fflush after writing the prompt, because the standard output is normally line buffered, and the prompt does not contain a newline.

**Program:**

#include "apue.h"

#include <ctype.h>

int

main(void)

{

int c;

while ((c = getchar()) != EOF) {

if (isupper(c))

c = tolower(c);

if (putchar(c) == EOF)

err\_sys("output error");

if (c == '\n')

fflush(stdout);

}

exit(0);

}

**Input:**

INPut

**Output**:

input

**Conclusion**:

In this way lowercase text is formed.

**References**:

**1.[https://notes.shichao.io/apue/ch15/**