

# 实验四 匿名管道通信

## 实验题目

匿名管道通信

## 实验目的

学习使用匿名管道在两个进程间建立通信。

## 实验软件环境

Ubuntu 21.04

## 实验内容


分别建立名为 Parent 的单文档应用程序和 Child 的单文档应用程序作为父子进程，由父进程创建一个匿名管道，实现父子进程向匿名管道写入和读取数据。

## 实验程序

```
1  #include <sys/wait.h>
2  #include <unistd.h>
3  #include <stdlib.h>
4  #include <string.h>
5  #include <stdio.h>
6  char buf[81];
7  int main( ) {
8      int pipefd[2];
9      int ret = 0;
10     if ((ret = pipe(pipefd)) == -1) {
11         printf("error pipe");
12         exit(-1);
13     }
14     if (ret == 0) {
15         int cpid = fork();
16         if (cpid == -1) {
17             printf("error fork");
18             exit(-1);
19         }
20         if (cpid == 0) {
21             char output[100];
22             close(pipefd[1]);
23             sleep(2);
24             int t = 0;
25             read(pipefd[0], &output, 80);
26             printf("child read success\n");
27             close(pipefd[0]);
28             printf("child close success\n");
29             printf("the data is %s\n", output);
30             exit(0);
31         } else if (cpid > 0) {
32             close(pipefd[0]);
```

```
33     FILE* fd = fopen("Parent.in", "r");
34     fgets(buf, 80, fd);
35     write(pipefd[1], buf, strlen(buf));
36     printf("parent writeline success\n");
37     close(pipefd[1]);
38     printf("parent close success\n");
39     sleep(2);
40 }
41 }
42 return 0;
43 }
```

## 实验截图



```
问题  输出  调试控制台  终端
→ oslab4 gcc oslab4.c
→ oslab4 ./a.out
parent writeline success
parent close success
child read success
child close success
the data is 114514
→ oslab4 □
```