

Laboratório 06 – Comunicação entre processos

Chamada de sistema `pipe()`

A chamada de sistema `pipe()` permite que dois processos possam trocar informações. Para isto são criados 02 (dois) arquivos que servem para enviar informações e são associados aos processos através da chamada de sistema `pipe()`.

A forma de utilização da chamada de sistema `pipe()` é:

```
#include <unistd.h>
int pipe( int fildes[2]);
```

onde `fildes` são descritores de arquivos em que o `fildes[0]` é utilizado como um buffer de leitura e `fildes[1]` é utilizado como buffer de escrita.

O exemplo abaixo cria um processo FILHO através da chamada de sistema `fork()` e cria um buffer para troca de informações entre os processos através da chamada de sistema `pipe()`. O processo FILHO escreve para o processo PAI através do pipe. Observe que o processo FILHO, identificado por `fork() == 0`, fecha o descritor de arquivo `fd[0]` já que não via realizar a leitura. O mesmo acontece com o descritor `fd[1]` para o processo PAI, pois não realizará escrita.

```
#include <stdio.h>
#include <unistd.h>

#define BUFSIZE 80

int main(void)
{
    int fd[2];
    int n=0;
    int i;
    char line[BUFSIZE];

    pipe(fd); /* fd[0] - leitura, fd[1] - escrita*/

    if (fork() == 0) {

        close(fd[0]);

        for (i=0; i < 10; i++) {
            sprintf(line,"%d",n);
            write(fd[1], line, BUFSIZE);
            printf("Processo FILHO escreve: %d\n",n);
            n++;
            sleep(2);
        }
    }
    else {

        close(fd[1]);

        for (i=0; i < 10; i++) {
```

```

pipe\n");

printf("\t\t\t Processo PAI realiza a leitura do

read(fd[0], line, BUFSIZE);
sscanf(line, "%d", &n);
printf("\t\t\t Processo PAI lê: %d\n", n);

}

}

```

Tarefa

Com a chamada de sistema *pipe()* é possível dividir algumas atividades de computação entre processo PAI e processo FILHO, fazendo com que parte da execução de tarefas possam ser divididas em processos distintos, aproveitando os diversos núcleos de um processador para agilizar o processamento.

- 1) Criar funções que realizar a multiplicação de matrizes e calculo da inversa de uma matriz;

$$R = A^{-1} \cdot B^{-1}$$

- 2) Implementar um programa que possa executar a multiplicação da inversa de 02 (duas) matrizes da seguinte forma:

- a. O processo PAI inicia a 01 (um) processo FILHO;
- b. O processo PAI e o processo FILHO calculam a inversa de matrizes, cada um calculando a inversa de matrizes diferentes;
 - i. Processo PAI calcula A^{-1}
 - ii. Processo FILHO calcula B^{-1}
- c. O processo FILHO envia os dados da matriz inversa calculada para o processo PAI;
- d. O processo PAI recebe os dados do processo FILHO;
- e. O processo PAI multiplica a matriz recebida do processo FILHO pela matriz inversa que ele calculou;
 - i. Processo PAI calcula a matriz R
- f. Exibir o resultado da multiplicação.