

**Universidade Federal do Rio de Janeiro**  
**Disciplina: Sistemas Operacionais 2**  
**Trabalho 5**  
**Apresentação em Laboratório: 25/06/2019**

**Exercício Prático - Processos**

Implementar um *daemon*(\*) que, de *n* em *n* segundos (argumento do programa) acorda e escreve em um arquivo próprio de *log* informações sobre os processos ZOMBIEs do sistema.

Exemplo: `$ przombies 3`

Log:

PID	PPID	Nome do Programa
223	220	progl
321	220	xcount
223	220	progl
321	220	xcount
400	105	xpto

...

Existem pelo menos duas maneiras de identificar os processos ZOMBIEs. Uma delas seria examinar a saída do utilitário **ps**, executando-o como filho do *daemon* e obtendo-a através de um *pipe*. A outra consiste em varrer o diretório **/proc** (com as funções **opendir**, **readdir** e **closedir** já estudadas), que possui um subdiretório para cada processo do sistema (o nome do subdiretório é o próprio PID do processo). Dentro de cada um destes subdiretórios, existe um arquivo regular chamado **status**, de onde podem ser retiradas as informações desejadas. Há também o comando **stat**, que apresenta as mesmas informações de forma mais compacta. O *daemon*, que trabalha em *background*, finaliza sua execução quando recebe um **SIGTERM**, que deve ser interceptado e gerar mensagens de finalização no arquivo de *log*. Aos demais sinais (exceto obviamente o **SIGKILL**), o *daemon* deve ser invulnerável.

Para produzir os processos ZOMBIEs, você pode utilizar o seguinte programa:

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

#define EVER ;;

int
main (int argc, char **argv)
{
    int n;

    if ((n = (argc == 1) ? 1 : atol (argv[1])) <= 0)
    {
        fprintf (stderr, "Use: %s [<n>]\n", argv[0]);
        return (1);
    }

    if (fork ())
        /* Passa para background */
        exit (0);

    while (n-- > 0)
    {
        if (fork () == 0)
            exit (0); /* Filhos falecem ... */
    }

    for (EVER)
        pause (); /* ... e o pai nem liga */

    return (0);
}
```

(\*) – processo a ser iniciado em background