

Criando um Zumbi

Preâmbulo

A nomenclatura de diversos componentes de sistemas operacionais possuem uma característica muito peculiar, uma delas é o conceito de processos PAIS e FILHOS, que se assemelham ao que temos na vida real. Outros conceitos curiosíssimos são os processos **ZUMBI**.

Os processos **ZUMBI**, tal como na ficção, não podem ser mortos! E em sistemas operacionais o que ocasiona este tipo de processo é o fato do processo pai não recolher o *exit status* do processo filho.

A existência de processos **ZUMBI**, tal como nos quadrinhos, não é desejada! Mas hoje vamos fazer um experimento muito interessante, que é a criação, em um ambiente controlado, de um processo zumbi!

O seu objetivo é conseguir criar um único processo **ZUMBI** e depois conseguir encerrá-lo. Para alcançar esse objetivo você deve seguir os seguintes passos:

1. Primeiramente o seu processo deve ser capaz de tratar sinais SIGUSR1 ou SIGUSR2
2. Depois de instalar os tratadores de sinais o seu processo deve aguardar algum dos sinais acima;
3. Assim que receber o sinal, o seu processo deve criar um processo **ZUMBI** e aguardar por outro sinal (qualquer um dos dois do passo 1)
4. Depois que seu processo receber mais um sinal, ele deverá ser capaz de eliminar o zumbi e deverá novamente esperar por outro sinal (qualquer um do passo 1)
5. Depois que seu processo receber esse último sinal ele deverá sair com *exit status* 0.

Resumindo:

- Inicia; Aguarda Sinal; Cria processo zumbi; aguarda sinal; destrói processo zumbi; aguarda sinal; encerra com sucesso.
 - Você tem que estar pronto para receber, ao todo, 3 sinais (podendo ser SIGUSR1 ou SIGUSR2).

NADA deve ser impresso na saída padrão.

Exemplos

Exemplo 1

Seu processo poderá receber a seguinte ordem de sinais:

```
SIGUSR1
SIGUSR1
SIGUSR1
```

Exemplo 2

Seu processo poderá receber a seguinte ordem de sinais:

```
SIGUSR1
SIGUSR2
SIGUSR1
```

Exemplo 3

Seu processo poderá receber a seguinte ordem de sinais:

```
SIGUSR2
SIGUSR2
SIGUSR2
```

Exemplo 4

Seu processo poderá receber a seguinte ordem de sinais:

```
SIGUSR2  
SIGUSR1  
SIGUSR2
```

Author: Bruno Ribas