



## Trabalho 2 Fundamentos de Sistemas Operacionais - Questão 1

Alunos:

Eduardo Q. Gomes 14/0137068

Miguel Pimentel 14/0156143

### Material Usado

- Sistema Operacional Linux, distribuição Ubuntu 16.04 com a interface Gnome;
- Compilador gcc (5.4.0);
- Editor de texto Sublime.
- Valgrind versão 3.11.0

### Conteúdo Utilizado

- Conteúdo do Moodle disponível em <http://aprender.ead.unb.br/course/view.php?id=3806>
- Livro Autor: Autor: SILBERSCHATZ, A.; GAGNE, G.; GALVIN, P.B. Obra: Operating System Concepts

### Questão 1

**Escreva um programa em C que implemente uma aplicação que simula o alarme de relógio. A aplicação alarme.c deverá disparar um processo filho que, logo após sua execução, dorme por 5 segundos antes de enviar um SIGALRM ao processo pai. O processo pai, por sua vez, prepara-se para receber o sinal do tipo SIGALRM, aguarda por esse sinal; quando o sinal é recebido, imprime na tela a ocorrência do evento e, em seguida, se encerra. Faça o uso da função pause(void) para aguardar pelo sinal enviado pelo processo filho.**

O programa foi desenvolvido na plataforma linux, desta forma, a utilização deste programa é apenas possíveis nesse sistema operacional (SO). Por exemplo, a utilização do fork() utilizado para duplicar um processo.

Para a execução do programa é necessário realizar o seguinte procedimento:

1. Entre no diretório q01 onde estará disposta o código fonte e o Makefile para a execução do programa;
2. Abra um terminal( ctrl + alt + t para a maioria das distros linux), e digite:
  - a. make - para compilar o programa;
  - b. make run - para executar o programa;
  - c. make clean - para deletar o executável do programa.

Limitações conhecidas do programa: desconhecido.

Dependências: Depende das bibliotecas <stdio.h> <unistd.h> <signal.h>.

O programa não necessita de entradas do usuário, por sua vez, o programa cria um processo filho a partir do `fork()`, e então cria um processo filho, este é interrompido pelo sinal de um alarme que é disparado cinco segundos a partir da criação do processo filho.

O programa tem dois fluxos de execução, que dependem do retorno da função `fork()`:

1. O programa cria um processo filho com sucesso, então o programa irá imprimir: "Child Process", e então disparar o alarme em 5 segundos. Segue a representação:

```
eduardo@Nautilus2:~/Documentos/GitE/S0_Trabalhos/T2/q01$ make
gcc alarme.c -o q01 -W -Wall -pedantic -lm
eduardo@Nautilus2:~/Documentos/GitE/S0_Trabalhos/T2/q01$ make run
./q01
Child Process
Alarme de tempo
```

2. O programa não consegue criar um processo filho, então a tela imprime a mensagem: ERROR: Child Process has not been created.