

**Professor:** João Tavares

**Data da entrega:** 16/9/19 19h (entregar via moodle - .zip único contendo os fontes com os nomes de todos os integrantes – em comentário)

**Peso:** 20% GA

**Observação:** Podem ser formados grupos entre 2 e 3 alunos.

### **Definição do contexto**

Desenvolva em C um programa que desempenhe o papel de menu de disparo e controle de aplicações, denominado **applications\_menu**. Inicialmente o programa deve oferecer ao usuário um menu em modo texto de forma que este possa selecionar a aplicação a ser executada, vide quadro de exemplo a seguir.

Exemplo:

```
<<<< Applications Menu >>>>
  1) Web Browser           (executando, pid=1234)
  2) Text Editor           (executando, pid=3456)
  3) Terminal              (falhou)
  4) Finalizar processo    (concluído)
  5) Quit
Opção: _
```

Uma vez que tenha sido selecionada uma aplicação a ser executada, o **applications\_menu** deve dispará-la em um novo processo e registrar o status de sua execução e seu PID (ex.: executando, pid=1234). Ao término da aplicação, o menu deve registrar o tipo de terminação:

- **concluída** – em caso de terminação normal com sucesso
- **falhou** – em caso de terminação normal, porém com fracasso
- **abortado** – em caso de terminação anormal.

Dependendo da aplicação selecionada pelo usuário, o programa **applications\_menu** deve solicitar ao usuário parâmetros adicionais que serão repassados à aplicação a ser disparada:

- Caso seja selecionada opção 1 (Web Browser) deverá ser também lida da entrada padrão a URL a ser exibida no navegador web.
- Caso seja selecionada a opção de finalizar processo, deverá ser lida da entrada padrão qual das aplicações em execução (1, 2 ou 3) deverá receber o sinal SIGTERM. Nesse caso, a partir do tipo da aplicação selecionada, a aplicação **applications\_menu** deve consultar suas estruturas de dados internas para identificar o PID do processo que deverá receber o sinal de terminação e enviá-lo com a chamada a função **kill()**.

Como requisito final, o estado do menu deve ser atualizado a cada 5 segundos ou sempre que o usuário digitar Ctrl+C, ou ao término de um processo filho (o que pode ser percebido tratando-se o sinal SIGCHLD).

Por simplificação, considere que haverá apenas uma instância de cada aplicação executando.

**RECOMENDAÇÃO:**

O arquivo contendo o código fonte C do programa, deve ser devidamente formatado e documentado com comentários complementares ao código. Um bom comentário no código deve esclarecer o propósito ou justificativa do comando no contexto da lógica (algoritmo) da aplicação, e não ser uma simples tradução literal das palavras.

**AVISO IMPORTANTE:**

É permitido e benéfico discutir e trocar ideias com os colegas ou mesmo utilizar os fóruns para resolver dúvidas. Porém, em nenhuma circunstância cópias serão admitidas, mesmo que parciais, sendo penalizadas com a nota zero a todos os envolvidos. Se emprestar diretamente a sua solução a um colega, tenha isso em mente.

**Bom trabalho!**