
 INSTITUTO DE INFORMÁTICA UFG	Universidade Federal de Goiás Instituto de Informática Sistemas Operacionais	 UFG
Professor: Diego Américo Guedes		Data: 24/09/2020
Lista de Exercícios - Processos <u>Vale nota (64h e 96h) e presença (96h)</u>		

1. Analise o código do programa a seguir e, sem o compilar e executar, anote as ocorrências previstas de cada um dos caracteres indicados na Tabela 1. Em seguida, compile, execute o programa e anote as ocorrências observadas de cada caractere. Entendeu como a chamada de sistema `fork()` funciona?

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

int main()
{
    pid_t pid;
    pid=fork();

    printf("A\n");
    if(pid==0)
    {
        printf("B\n");
    }

    printf("C\n");

    if(pid>0)
    {
        printf("D\n");
    }

    printf("E\n");

    return 0;
}
```

Caractere	Ocorrências previstas	Ocorrências observadas
A		
B		
C		
D		
E		

Tabela 1

2. Implemente um programa que crie N processos, todos filhos do mesmo pai. N é um valor fornecido pelo usuário. Todos os processos devem imprimir seu PID e PPID.

3. Implemente um programa que crie uma cadeia de processos N processos (além do pai). A cadeia deve obedecer a seguinte sequência: 1º processo cria o 2º processo, 2º processo cria o 3º processo, ..., Nº – 1 processo cria o Nº processo. Cada processo deve imprimir seu PID e o PID do seu pai (ou PPID). Garanta que a informação exibida na tela ocorrerá na ordem inversa da criação dos processos, ou seja, inicialmente aparece as informações do Nº processo (PID e PPID), depois do Nº – 1, ..., depois do 2º e por fim do 1º.