**ATIVIDADE 4 – Processos - FORK**

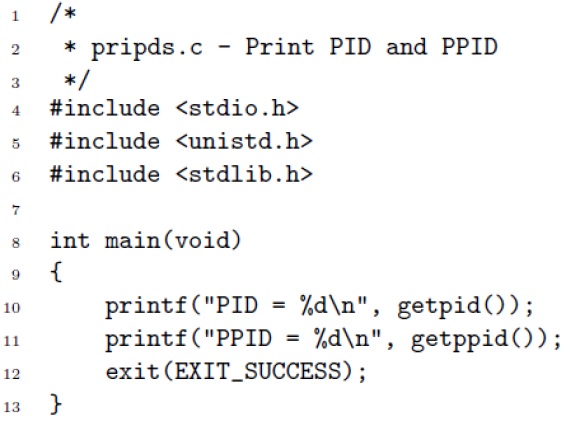
**Identificadores de um Processo**

Um processo é identificado pelo seu PID. Adicionalmente, o processo tem a informação do PID do seu pai, o PPID.

* PID (*Process* ID) - número natural que identifica um processo;
* PPID (*Parent Process* ID) - número natural que identifica o PID do processo seu processo pai.

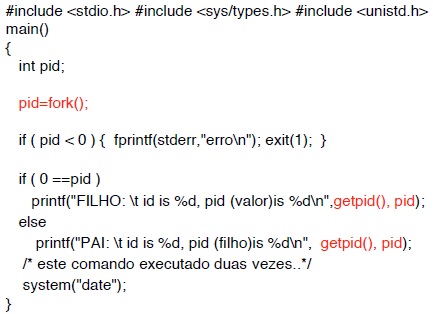
O PID pode ser utilizado pelo processo para a criação de um arquivo temporário ou escrever o PID em um arquivo de log. O PPID é utilizado pelo processo para mandar sinais ou outras mensagens para seu processo pai.

O código abaixo obtém o PID e o PPID do processo. As chamadas de sistema *getpid* e *getppid* são utilizadas (use o *man* para mais informações).



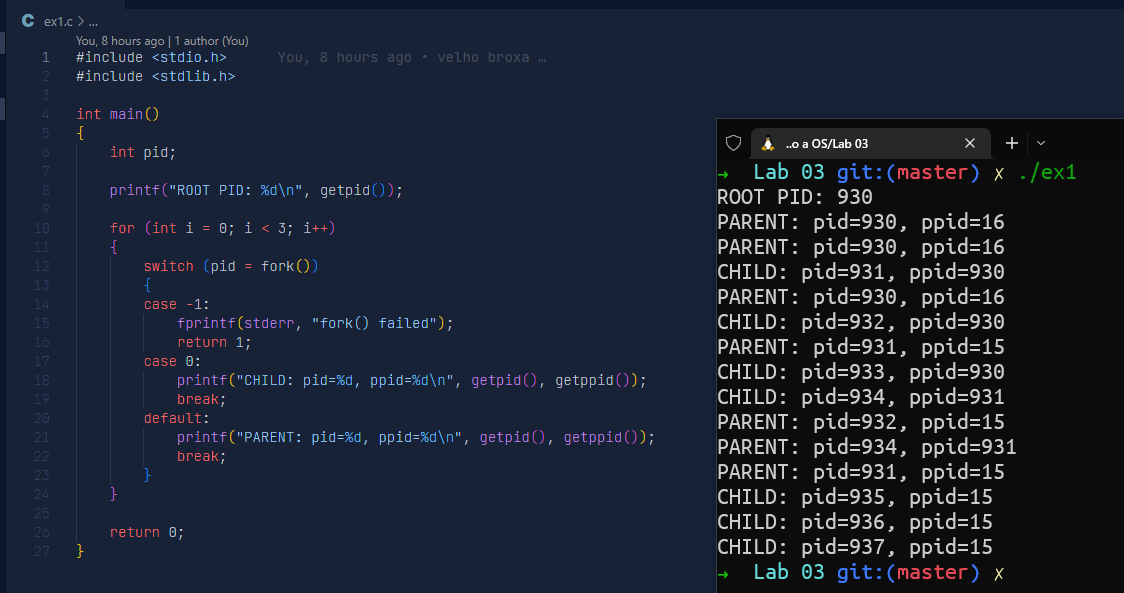
**Comando *Fork***

O comando *fork()* permite a criação dinâmica de processos, os quais serão filhos do processo que executa este comando, veja o exemplo abaixo:



Execute o código acima e analise a sua saída.

**PARTE 1** - Após entender o exemplo, crie outro programa que chame o método *fork()* várias vezes seguidas, após cada *fork*, imprima o PID do processo e de seu pai. Analise a saída.



**PARTE 2** - Utilizando o comando Linux para apresentar os processos em árvore, identifique os processos criados por seu programa (elaborado no item anterior). Explique a partir da árvore de processos apresentada a relação entre os processos criados no seu programa.

Cada chamada de **fork()** cria um novo processo que roda em paralelo com o processo original, rodando o mesmo código que vem depois da chamada da função.