**Processos**

**Marcos Vinícius Simão de Souza – 1591916**

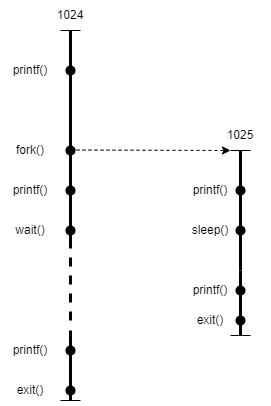
**Exercício 1:**

fork() – Função responsável por criar novos processos no sistema, ela retorna 0 se for um processo filho, 1 se for um processo pai e -1 se ocorrer algum erro.

getpid() – Mostra qual é o ID associado a um processo criado e rodando. A ID é única para cada processo.

wait() – Função que gera uma espera de ms no sistema.

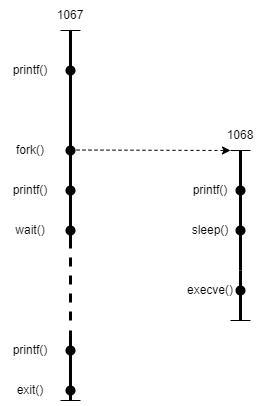
O programa foi escrito para mostrar a ID do processo filho que é criado pelo fork() e também de seu processo pai. Embora um processo não afete como um ou outro irá operar, a função wait() faz com que gere uma interrupção e o processo pai aguarde pela resposta do processo filho e só após o encerramento do processo filho que o programa encerra o processo pai.



**Exercício 2:**

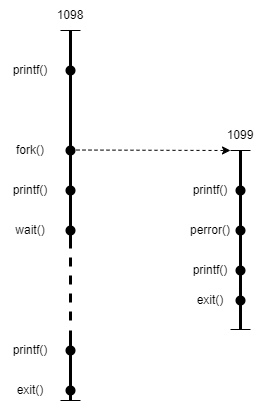
execve() – Substitui o processo atual por um novo processo ou programa, ao completar com sucesso a nova chamada é substituído o texto, dado e pilha do processo original pelos do novo processo e em caso de falha é retornado -1.

O programa exibirá a ID (nesse caso 1067) do processo corrente e cria um processo filho, cujo qual serão apresentados os dados de ID (1068) e processo pai na tela, em seguida chama-se a função execve() que executa uma interrupção e exibe a data atual e ao encerrar o processo da data o programa encerra também o processo pai pois o processo filho foi substituído pelo processo da data.



**Exercício 3:**

O programa realiza o fork() criando assim um processo filho (1099) para o processo pai (1098), o processo pai e filho exibindo um valor de x = 0, é incrementado 1 a x e mostrado na tela que agora x do processo filho vale 1 e encerra-se o mesmo, em seguida mostra na tela que x do processo pai ainda vale 0 e encerra o mesmo.



**Terminal:**

