

Lista de Exercícios

1. Construir um programa MPI no qual o *master* passa um número inteiro diferente para cada processo escravo/*worker*. Esses, por sua vez, imprimem o número recebido junto com seu *rank*.
2. Altere o programa anterior de modo que o escravo altere o número recebido (somando duas unidades ao valor recebido) e devolva o vetor alterado para o *master*, que deve imprimi-lo.
3. Construir um programa MPI no qual o master passa um vetor de inteiros diferentes para cada processo escravo. Esses, por sua vez, imprimem o vetor recebido junto com o seu *rank*.
4. Altere o programa anterior de modo que os escravos subtraíam uma unidade de cada posição no vetor recebido e devolvam o vetor alterado para o mestre/*master*, que será encarregado por imprimir cada vetor recebido.
5. Construir um programa MPI no qual os processos escravos sinalizem para o processo master que estão vivos (“Oi, Estou vivo!”). O master, por sua vez, deve imprimir a mensagem recebida junto com o *rank* do escravo que enviou.
6. Elaborar um programa MPI que faça o repasse de um token entre processos residentes em hosts distintos, de forma circular. Ou seja, o token começa no token zero, vai para o rank 1 e assim sucessivamente até voltar de novo para o rank 0. Por sua vez, cada processo deve imprimir mensagens de sinalização de recepção e/ou envio do token para o próximo nó. Em função da sua posição na lista circular, cada processo deve imprimir um dos seguintes tipos de mensagem exemplificados: Rank[0/'cm1']: enviando token para Rank[1/'cm2']: recebeu token e enviou para 2. Obs.: Opcionalmente, pode-se usar a função `MPI_Get_processor_name(hostname, &hostname_len)` para recuperar o nome do host. Neste caso, `hostname` é um vetor de char e `hostname_len` é do tipo int.