

Professor: Antonio Augusto Rocha

Disciplina: Sistemas Distribuídos

Período: 2016/2

Alunos: _____



Prova Surpresa 4

Obs.: Essa prova consiste de um projeto de implementação que pode ser feito individualmente ou em dupla. Cada solução deve ser implementada e compreendida pelo(s) autor(es). Não será aceita qualquer tipo de cópia parcial ou integral da solução. A violação dessa regra pelo(s) aluno(s) poderá resultar em nota zero para esta avaliação. Embora essa prova seja opcional, lembre-se que todo aluno deverá fazer ao menos três das provas surpresas ao longo do semestre.

- 1) Implementar duas versões de um mesmo programa, um sequencial e outro paralelo utilizando primitivas MPI, para checar quais números de uma lista são primos. Essa lista deve ser gerada aleatoriamente, contendo um tamanho específico de elementos, cujo os elementos devem ser números inteiros entre 1 e um determinado valor máximo. O programa deve receber como parâmetros de entrada: o tamanho da lista de números a ser, o valor máximo para esses números, uma semente geradora dos números aleatórios, e o nome do arquivo de saída.

Ao iniciar, o programa deverá gerar uma lista aleatória de números utilizando a semente especificada. Ou seja, a execução do mesmo programa duas vezes, tendo os mesmo parâmetros de entrada, deve necessariamente gerar as mesmas listas e, obviamente, os mesmo números primos.

Devem ser implementadas uma versão sequencial e uma versão paralela, onde a versão paralela deve usar o modelo mestre escravo, onde o mestre atribui aos escravos a tarefa de checagem dos números. Os escravos devem processar todos os elementos da lista, retornando se o número é primo ou não. A lista dos números primos deve ser escrita em um arquivo de saída. O processamento da versão paralela deve enviar para checagem um número para cada escravo por vez, porém com diversos escravos executando em paralelo. A versão paralela pode ser bloqueante ou não bloqueante, a decisão fica à critério do grupo. O número de escravos deve ser flexível, a depender da configuração do ambiente de execução. Por fim, o programa deve imprimir também no arquivo de saída o tempo total de execução do programa.