

Disciplina DCE540 - Computação Paralela e Distribuída	Método de entrega Moodle da disciplina	Data de entrega 31/08/2021 às 14h00
Professor Iago Augusto de Carvalho (iago.carvalho@unifal-mg.edu.br)		

### Prova 01

Cada aluno deverá submeter um único arquivo .pdf com a resolução da prova.

A prova pode ser realizada de duas maneiras:

- Com papel e caneta, sendo posteriormente escaneada e enviada
- Digitada em algum editor de texto, e.g., Word ou LaTeX

A prova deverá ser entregue no Moodle da disciplina até a data limite.

- Atrasos não serão tolerados

### Exercício 1 (15 %)

Uma das características básicas de sistemas distribuídos é a transparência. Em um sistema distribuído transparente, espera-se que seus recursos, embora fisicamente distantes entre si, sejam exibidos de uma forma coesa e clara aos usuários e desenvolvedores, de tal forma que estes não precisem se preocupar com a localização espacial dos recursos que estão utilizando. Existem alguns importantes tipos de transparência que devem ser garantidos em sistemas distribuídos. Explique, com suas próprias palavras, as definições de

- a) Transparência de acesso
- b) Transparência de localização
- c) Transparência a falhas

### Exercício 2 (20%)

Uma *Remote Procedure Call* (RPC) é uma forma de fazer chamadas a métodos de componentes remotos.

- a) Explique o funcionamento de uma RPC
- b) Uma RPC, no geral, se baseia em chamadas transientes e síncronas. Explique as vantagens e desvantagens deste tipo de chamada

### Exercício 3 (10%)

O que é um *middleware*? Qual é a importância de *middlewares* para sistemas distribuídos?

### Exercício 4 (10%)

Apresente as definições de

- a) Comunicação transiente
- b) Comunicação persistente
- c) Comunicação síncrona
- d) Comunicação assíncrona

## Exercício 5 (20%)

Dê exemplos de aplicações que utilizem

- a) Comunicação transiente e síncrona
- b) Comunicação transiente e assíncrona
- c) Comunicação persistente e síncrona
- d) Comunicação persistente e assíncrona

## Exercício 6 (25%)

Considere um sistema computacional que possui 4 processadores independentes, cada uma com uma *thread*. Calcule o *speed-up* de algoritmos com as seguintes características:

- a) Algoritmo tem uma fração paralela de 50% (decomposição de domínio)
- b) Algoritmo tem uma fração paralela de 50% (decomposição funcional), sendo que a maior das quatro sub-tarefas corresponde a 50% da fração paralela
- c) Algoritmo possui duas frações paralelas. A primeira é uma decomposição de domínio e corresponde a uma fração de 20% do algoritmo. A segunda é uma decomposição funcional e corresponde a 30% do algoritmo, sendo que as quatro sub-tarefas levam o mesmo tempo para serem executadas

# Gabarito

## Exercício 1

- a) Oculta diferenças na representação de dados e no modo de acesso a um recurso, todos conseguem acessar independentemente de como representa os dados e acessa o recurso
- b) Oculta o lugar onde o recurso está localizado, ou seja, não precisa saber onde o recurso está para acessá-lo
- c) Oculta quando ocorre falha e a recuperação de um recurso é feita de forma que o usuário não percebe que processo falhou

## Exercício 2

- a) Um processo A chama procedimento em uma máquina remota B e é suspenso. A execução do procedimento ocorre em B, sendo que o chamador envia os parâmetros do procedimento ao chamado, que retorna o resultado ao procedimento chamador através de troca de mensagens, que não são visíveis ao programador
- a) Transiente significa que o receptor e remetente devem estar ativos durante toda a comunicação, então mensagens podem ser perdidas. E síncrona significa que o remetente necessita ficar bloqueado esperando uma resposta do receptor e muitas vezes isso não é necessário e diminui a eficiência dos processos.

## Exercício 3

*Middleware* é o software distribuído que coordena o sistema distribuído enquanto fornece transparência ao sistema. *Middlewares* são as peças centrais de sistemas distribuídos pois possibilita que

- programadores construam aplicações sem ser necessário criar interfaces personalizadas para acessar diferentes componentes (serviços ou microserviços), bancos de dados ou dispositivos de computação
- usuários consigam utilizar um sistema distribuído de maneira simples e transparente, sem se preocupar com a localização física dos aplicativos e recursos do sistema

## Exercício 4

- a) Esquema de comunicação no qual mensagens só existem no sistema enquanto os componentes (ou aplicações) envolvidos estejam ativos
- b) Esquema de comunicação no qual mensagens *persistem* no sistema, mesmo quando todos os componentes envolvidos na comunicação estejam inativos
- c) Comunicação na qual tanto o remetente quanto o(s) destinatário(s) ficam bloqueados até que a comunicação tenha sido finalizada
- d) Comunicação na qual nem o remetente nem o(s) destinatário(s) ficam bloqueados durante o processo

## Exercício 5

- a) RPC: receptor e remetente devem estar ativos durante a comunicação e ficam bloqueados até que a comunicação solicitada tenha sido realizada
- b) MPI assíncrona: receptor e remetente devem estar ativos durante a comunicação e não ficam bloqueados até que a comunicação solicitada tenha sido realizada
- c) Certas implementações de filas de mensagens: receptor e remetente não necessitam estar ativos durante a comunicação e ficam bloqueados até que a comunicação solicitada tenha sido realizada

- d) Caixa de e-mails: receptor e remetente não necessitam estar ativos durante a comunicação e não ficam bloqueados até que a comunicação solicitada tenha sido realizada

## Exercício 6

- a) Aplicação direta da Lei de Amdhal:

$$S(n) = \frac{1}{(1 - B) + \frac{1}{n} B} = \frac{1}{(1 - 0,5) + \frac{1}{4} 0,5} = 1,6$$

- b) Na decomposição funcional, existe um ponto de sincronia ao fim da realização de todas as sub-tarefas. Assim, devemos considerar somente o tempo da maior das sub-tarefas. Neste exercício, temos que 50% do algoritmo permite decomposição funcional e os outros 50% devem ser executados de forma serial. Além disto, a maior sub-tarefa corresponde a 50% do tempo da fração paralela, isto é, 25% do tempo total do algoritmo. Então, o speed-up é igual a 50% do tempo da fração paralela. Seja  $x$  o tempo do programa serial. Temos que o speed-up é

$$S(n) = \frac{x}{0,75 x} = \frac{1}{0,75} = 1,333$$

- c) Neste exercício, temos que todas as sub-tarefas da decomposição funcional gastam exatamente o mesmo tempo para serem executadas. Desta forma, pode-se fazer o calculo como se a decomposição funcional fosse uma decomposição de domínio. Somando as duas frações paralelas, temos que 50% do algoritmo pode ser paralelizado. Deste modo, temos que

$$S(n) = \frac{1}{(1 - B) + \frac{1}{n} B} = \frac{1}{(1 - 0,5) + \frac{1}{4} 0,5} = 1,6$$