**P1 2017**

**Parte Teórica**

**1 - O projeto e desenvolvimento de sistemas distribuídos suscita diversos desafios, dentre os quais oferecer um sistema distribuído aberto. Neste contexto, explique o que significa ser “aberto”. (1,5)**

**R: (Capítulo 1 - 1.5.2 Sistemas Abertos - páginas 17 e 18)** Diz-se que um sistema computacional é aberto quando ele pode ser estendido e reimplementado de várias maneiras. O fato de um sistema distribuído ser ou não um sistema aberto é determinado principalmente pelo grau com que novos serviços de compartilhamento

de recursos podem ser adicionados e disponibilizados para uso por uma variedade

de programas clientes.

Resumindo:

• Os sistemas abertos são caracterizados pelo fato de suas principais interfaces serem

publicadas.

• Os sistemas distribuídos abertos são baseados na estipulação de um mecanismo de

comunicação uniforme e em interfaces publicadas para acesso aos recursos compartilhados.

• Os sistemas distribuídos abertos podem ser construídos a partir de hardware e software

heterogêneo, possivelmente de diferentes fornecedores. Para que um sistema

funcione corretamente, a compatibilidade de cada componente com o padrão publicado

deve ser cuidadosamente testada e verificada.

**2 - Sobre a representação de dados e empacotamento, assinale à alternativa incorreta (1,0):**

**à) Um método que pode ser utilizado para que dois computadores troquem valores de dados binários é o emissor converter os valores para um formato externo, acordado antes da transmissão, e o receptor converter os valores para o formato local na recepção.**

**b) O marshalling é o procedimento de pegar um conjunto de itens de dados e montá-los em uma forma conveniente para transmissão de uma mensagem.**

**c) A representação de dados em XML não inclui informações de tipos de dados.**

**d) Em Java o processo que desserializa um objeto pode verificar se ele tem a versão correta da classe**

**e) No Java, quando um objeto é serializado, todos os objetos que ele referencia são serializados junto.**

**3 - Em sistemas distribuídos os modelos de programação são diversos e incluem: à invocação remota de objetos, à notificação remota de eventos, o acesso remoto ao banco de dados e o processamento de transação distribuído. Diferencie o modelo de invocação remota de objetos do modelo de notificação remota de eventos. (1,5)**

O modelo de invocação remota de objetos oferece para o programdor o acesso a interface do programa onde pode localmente acessar as funções que são distribuidas, como por exemplo o Java RMI. O modelo de notificação remota de eventos é quando a comunicação é realizada, como o nome diz, por eventos, onde é realizado um cast para o evento e é aberto um listening onde espera o acontecimento, um exemplo é o Publish-Subscribe

**4 - Um protocolo de requisição-resposta deve ser implementado em um serviço de comunicação com falhas por omissão para fornecer semântica de invocação RMI *pelo menos uma vez*. Explique como deve funcionar o protocolo. (1,5)**

Falhas por omissão garante para usuário que a mensagem irá chegar fazendo mais de um envio no caso de falha. Pelo menos uma vez é realizado o envio da mensagem sem a preocupação de duplicatas, pois é necessário somente a garantia que chegue a mensagem, ao contrário de no máximo uma vez, que deve garantir a chegada de somente uma mensagem.

**Parte Prática**

**5 - O uso de técnicas e mecanismos de segurança num sistema distribuído pode ser feito de maneira similar a construção de um middleware? Explique sua resposta. (1,5)**

O middleware vai ser a camada mais superficial para o usuário, trazendo o máximo de transparência, por efeito é necessário que tenha mecanismos de proteção, uma vez que o middleware é o caminho entre o usuário e a funcionalidade desejada. Criptografia de dados é um mecanismo que deve ser implementado por um middleware para garantir segurança na troca de mensagem.

**6 - O objetivo de um serviço de nomes é mapear nomes aos seus respectivos endereços. Na internet, o serviço de nomes mais conhecido é o DNS. Qual a técnica utilizada pelo DNS para poder atender uma rede muito grande tipo à Internet? (1,5)**

O serviço de nomes da internet em seu início era centralizado, porém com a explosão da internet foi necessário a criação de algo mais robusto, então se originou o DNS, um serviço de nomes distribuídos, onde cada entidade é responsável pelo seu domínio e subdomínios localmente, sendo ela responsavel pela inclusão e exclusão de novos nomes

**7 - Dê um exemplo de como um alto grau de transparência pode trazer um efeito desfavorável em um sistema distribuído. (1,5)**

A transparência é desejada para facilitar ao máximo a utilização do sistema, um middleware sempre busca ter o máximo de transparência pois seu objetivo é facilitar a comunicação de qualquer funcionalidade em um sistema. O excesso de transparência pode engessar o middleware, tornando muito específico e de difícil utilização para casos adversos porém pertencentes ao mesmo contexto.

**P1 2018**

**Parte Teórica**

**1 - Em sistemas distribuídos os modelos de programação são diversos e incluem: invocação remota de objetos, notificação remota de eventos, chamada remota de procedimentos e orientado à mensagens. Diferencie o modelo de invocação remota de objetos do modelo de notificação remota de eventos. (1,5)**

A invocação remota de objetos é quando é oferecido uma interface onde pode ser acessadas a suas funções localmente, como por exemplo o Java RMI, ja a notificação remota de eventos é como um publish subscribe, onde eventos são ouvidos e realizados conforme sua ocorrência.

**2 - O termo middleware se aplica à uma camada de software que fornece uma abstração de programação. Esta abstração oculta (torna transparente) detalhes do sistema distribuído. Dê um exemplo de como um programador pode se beneficiar de uma transparência criada por outro programador. (1,0)**

A transparência sempre é o objetivo do middleware, pois ele é feito para suprir uma necessidade do sistema, um middleware transparente faz com que o programador não se preocupe na forma de que é feita tal funcionalidade, somente se preocupa em utilizar-la.

**3 - Diferencie as arquiteturas de sistema distribuído cliente-servidor e peer-to-peer (1,5)**

Uma arquitetura cliente servidor é bem como o nome diz, vários usuários clientes irão fazer solicitações para um servidor centralizado que irá respondê-las, já em um sistema distribuído peer-to-peer é retirada essa centralização e todos envolvidos são clientes e servidores, onde cada um pode realizar uma requisição para outro tanto como responder.

**4 - em um serviço de comunicação com ??? por omissão para fornecer semântica de invocação RMI pelo menos uma vez. Explique como deve funcionar o protocolo. (1,5)**

Comunicação com falhas por omissão significa que a solução para uma falha de uma mensagem será resolvida com o envio de outra mensagem, em uma semântica de pelo menos uma vez será realizado o envio de mensagens até a acusação do recebimento, sem a preocupação de duplicatas.

**Parte Prática**

**5 - Faça um comparativo entre algoritmos de resumo e algoritmos de criptografia. Cite as principais diferenças entre eles. (1,5)**

**6 - O objetivo de um serviço de nomes é mapear nomes aos seus respectivos endereços. Na internet, o serviço de nomes mais conhecido é o DNS. Quais as estratégias/técnicas utilizadas pelo DNS para poder atender uma rede muito grande tipo à Internet? Justifique sua resposta. (1,5)**

O serviço de nomes da internet no início era centralizado, porém com a explosão da internet isso se tornou impossível, por isso a implementação do DNS que torna distribuído o serviço de nomes, onde cada indivíduo é responsável pelo domínio e a inclusão e remoção de novos.

**7 - Considere que você está implementando um middleware de comunicação orientado a mensagens. O que mudaria em sua implementação se a especificação mudasse de comunicação transiente orientada à mensagem para comunicação persistente orientada à mensagem? (1,5)**

Comunicação persistente tira a necessidade do usuário estar online para a comunicação, desta forma sendo necessário uma implementação onde armazena as mensagens para que possam ser acessadas posteriormente.

**P1 2019**

**Parte Teórica**

**1 - Explique as funções do stub no RMI. (1,5)**

**R(slide 23 - [Slides aula sobre Objetos Distribuídos e Java RMI](https://ead06.proj.ufsm.br/mod/resource/view.php?id=810954)**

**[Arquivo)](https://ead06.proj.ufsm.br/mod/resource/view.php?id=810954):** Stub (lado cliente):

* Transforma parâmetros (serializados)
* Envia requisição pela rede (sockets);

O stub esconde a serialização dos parâmetros e toda a comunicação a nível de rede, com o objetivo de simplificar o mecanismo de realização da invocação do método.

Resultado: TRANSPARÊNCIA

O stub é responsavel pela comunicação do RMI, ele serializa os dados e envia a requisição via rede. seu principal objetivo é simplificar a realização da invocação de um método.

**2 - Explique o que é preciso fazer para que um programa cliente escrito em Java possa trocar dados via mensagens com um servidor escrito em C++. (1,5)**

**R:** É necessário estabelecer uma representação de dados externa comum entre o cliente e o servidor, como CORBA, XML, JSON ou Protobuf. Com a representação de dados estabelecida é necessário buscar/criar um serviço para empacotamento/serialização e desempacotamento/deserialização das mensagens.

**3 - Conceitue o paradigma de “comunicação indireta” (0,75) e dê dois exemplos de uso em sistemas distribuídos (0,75).**

**R (Capítulo 6 - 6.1 Introdução - página 230):** A comunicação indireta é definida como a comunicação entre entidades de um sistema distribuído por meio de um intermediário, sem nenhum acoplamento direto entre o remetente e o destinatário (ou destinatários).

Possui duas propriedades que resultam do uso de um intermediário:

* *Desacoplamento espacial*, no qual o remetente não sabe ou não precisa saber a identidade do destinatário (ou destinatários) e vice-versa.
* *Desacoplamento temporal*, no qual o remetente e o destinatário (ou destinatários)

podem ter tempos de vida independentes.

Exemplos: a comunicação indireta é frequentemente usada em sistemas distribuídos

em que são previstas alterações, por exemplo:

1. em ambientes móveis, onde os usuários podem se conectar e desconectar rapidamente da rede global – e devem ser gerenciadas para fornecer serviços mais confiáveis.
2. muito usada para disseminação de eventos em sistemas distribuídos, em que os destinatários podem ser desconhecidos e estar propensos à mudança – por exemplo, no gerenciamento da disseminação de eventos (feeds) em sistemas financeiros, conforme apresentado no Capítulo 1

**4 - Quais as estratégias/técnicas utilizadas pelo DNS para poder atender uma rede muito grande tipo a Internet? Justifique sua resposta. (1,5)**

A principal estratégia do DNS é a distribuição do serviço de nomes, onde cada indivíduo cuida de seu conjunto de domínio e subdomínios e também sua inserção e remoção de novos domínios, descentralizando o serviço de nomes e tornando possível a constante expansão.

**5 - Todos os modelos de arquitetura para sistemas distribuídos, apesar de bastante diferentes, apresentam algumas propriedades fundamentais idênticas. Em particular, todos são compostos de processos que se comunicam por meio do envio de mensagens através de uma rede de computadores. Mas há outras propriedades fundamentais que precisam ser esclarecidas. Neste contexto, descreva à função do modelo de interação. (1,0)**

A principal função do modelo de interação é cuidar da consistência das mensagens, sendo o principal método a sincronização de relógios, que possuem várias implementações porém todas buscam sincronizar as mensagens de todos computadores para manter consistência e ordem nas mensagens.

**6 - Um protocolo de requisição-resposta deve ser implementado em um serviço de comunicação com falhas por omissão para fornecer semântica de invocação RMI pelo menos uma vez. Explique como deve funcionar o protocolo. (1,5)**

Um serviço que suporta falhas a omissão significa que caso a falha de uma mensagem ocorra será enviado outra mensagem para suprir a falha, na semântica de pelo menos uma vez a mensagem será enviada novamente até que seja recebido pelo usuário, sem o tratamento de duplicatas.

**7 - Um dos requisitos de segurança para tornar operações de compra pela Internet seguras é a autenticação do vendedor para o comprador, para que este possa ter confiança de que está em contato com um servidor operado pelo vendedor com quem pretende negociar. Explique o que são certificados digitais (0,75) e como eles ajudam nesta garantia para ofertar segurança à um comprador (0,75)**

**R:** Certificados digitais usam criptografia assimétrica e uma entidade certificadora para confirmar a validade dessa chave. Eles garantem a identidade de ambas partes de uma troca, dando a segurança de que nenhuma das partes esteja perdendo seus recursos para terceiros.

Certificados digitais usam criptografia assimétrica e uma entidade certificadora para confirmar a validade dessa chave. Eles garantem a identidade de ambas partes de uma troca, dando a segurança de que nenhuma das partes esteja perdendo seus recursos para terceiros.

**P2 2017**

**Parte Teórica**

**1 - O NTP pode ser utilizado para sincronizar relógios de computador. Explique porque, mesmo com este serviço, nenhum limite garantido é dado para à diferença entre relógios de computadores distintos. (1,0)**

Pois o NTP utiliza de dados da internet para fazer sua sincronização e tais dados são sensíveis a falhas, uma vez que vários fatores podem contribuir para uma comunicação errada, como latência, perda entre outros.

**2 - O relógio lógico de Lamport captura numericamente à relação “aconteceu antes” e o relógio vetorial permite capturar à relação de ordem causal. Explique a "lógica" por trás do relógio vetorial. (1,5)**

O relógio vetorial utiliza de um vetor para manter o estado do clock de todos os processos. Cada processo possui seu vetor local e em cada comunicação é repassado para cada um dos processos por piggyback para serem ser atualizados, dessa forma é possível sincronizar todos os processos e ainda garantir a permanência de mensagens.

**3 - Alcançar o acordo em sistemas distribuídos tipicamente requer que os processos tenham conhecimento sobre o valor proposto por outros processos. Um algoritmo de consenso busca um tipo de acordo onde qualquer processo participante pode propor um valor, mas o resultado acordado é apenas um valor. Descreva à propriedade de VALIDADE para este algoritmo. (1,5)**

**4 - Em muitos algoritmos de acordo distribuído um dos processos faz o papel de coordenador. Explique porque esta estratégia é interessante. (1,5)**

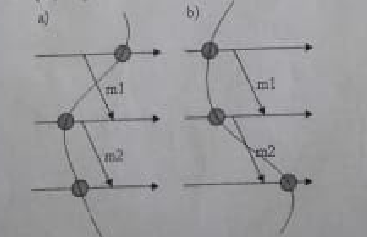
Um coordenador é interessante pois diminui significamente a complexidade das eleições, pois, é feito pelo coordenador um multicast para todos envolvidos e depois os envolvidos respondem somente para ele, diminuindo as mensagens necessárias para o acordo.

**Parte Prática**

**5 - Considere dois relógios C1 e C2, onde C1 adianta e C2 atrasa. À taxa máxima de adianto ou atraso é p, ou seja, após t? segundos C1 e C2 divergem em 2.p.t segundos. Se o projetista do sistema distribuído espera garantir que C1 E C2 não se distanciam mais de que 8 segundos, os relógios devem ser sincroniza dos no máximo a cada quantos segundos? (1,5)**

Para encontrar a sincronização máxima temos:

ð/(2.p) → 8/2.p → 4/p

**6 - Considere um sistema com três processos e canal confiável, como na figura abaixo. Um estado global consistente para este sistema é aquele que corresponde à um corte consistente. Indique se as linhas de corte das figuras à) e b) correspondem à um estado consistente ou não (0,75). Justifique sua resposta (0,75).**

**7 - O número de mensagens costuma ser uma das métricas de desempenho para avaliar algoritmos distribuídos. Demonstre porque na ausência de concorrência o algoritmo baseado em anel de Chang e Roberts troca 3N-1 mensagens (1,0). Destaque o significado da ausência de concorrência em sua demonstração (0,5).**

**P2 2018**

**Parte Teórica**

**1 - Explique para que serve o algoritmo distribuído “sobreposição de roteamento” nos middlewares peer-to-peer (1,0) e indique qual à principal tecnologia usada para implementá-lo. (1,0)**

**R:** Localiza nós e objetos. Assume a forma de uma camada responsável por direcionar requisições de qualquer cliente para um host que contenha o objeto para o qual a requisição é endereçada. Os objetos de interesse podem ser alocados e, subsequentemente, movidos para qualquer nó da rede, sem envolvimento do cliente.

A principal tecnologia usada são os GUIDs, auxiliando no roteamento de requisição de objetos, inserção e remoção de objetos, adição e remoção de nós

**2 - O NTP pode ser utilizado para sincronizar relógios de computador. Explique porque, mesmo com este serviço, nenhum limite garantido é dado para à diferença entre relógios de computadores distintos. (1,0)**

**R:** O NTP usa comunicação UDP multicast para fazer as requisições ao *time server*, podendo ter atraso não resposta ou até perdas para um dos computadores que fez a requisição. Fora isso, mesmo que o NTP tente realizar correção de delay, a ausência de um relógio global caracteriza uma possível imprecisão no protocolo para sincronizar relógios diferentes.

**3 - O relógio lógico de Lamport captura numericamente à relação “aconteceu antes” e o relógio vetorial permite capturar à relação de ordem causal. Explique à “lógica” por trás do relógio vetorial. (1,0)**

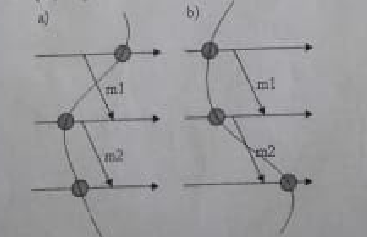
**4 - Em muitos algoritmos de acordo distribuído um dos processos faz o papel de coordenador. Explique porque esta estratégia é interessante. (1,5)**

**Parte Prática**

**5 - Considere dois relógios C1 e C2, onde C1 adianta e C2 atrasa. À taxa máxima de adianto ou atraso é p, ou seja, após t? segundos C1 e C2 divergem em 2.p.t segundos. S e o projetista do sistema distribuído espera garantir que C1 E C2 não se distanciam mais de que 8 segundos, os relógios devem ser sincronizados no máximo a cada quantos segundos? (1,5)**

8 = 2pt

t = 4/p

**6 - Considere um sistema com três processos e canal confiável, como na figura abaixo. Um estado global consistente para este sistema é aquele que corresponde à um corte consistente. Indique se as linhas de corte das figuras à) e b) correspondem à um estado consistente ou não (0,75). Justifique sua resposta (0,75).**

1. É um corte consistente, o envio e recebimento de cada mensagem está do mesmo lado de cada corte.
2. Não é um corte consistente, o corte está antes do envio e recebimento de m2.

**7 - O número de mensagens costuma ser uma das métricas de desempenho para avaliar algoritmos distribuídos. Demonstre porque na ausência de concorrência o algoritmo baseado em anel de Chang e Roberts troca 3N-1 mensagens (1,0). Destaque o significado da ausência de concorrência em sua demonstração (0,5).**

**R: (642-643)** A ausência de concorrência indica que apenas um processo dispara uma eleição. O pior caso corresponde ao início da eleição pelo sucessor do maior identificador.

* N mensagens ELEICAO para chegar ao futuro coordenador, que não dispara a eleição por que ainda não completou o anel.
* N-1 mensagens ELEICAO para percorrer o anel e chegar ao coord.
* N mensagens ELEITO para concluir a eleição
* TOTAL: 3N-1.

**P ÚNICA 2020**

**1 - Um multicast confiável definido pelas definições abaixo de integridade, acordo e validade atende grupos abertos ou fechados (um grupo é fechado se somente membros do grupo podem enviar mensagem para ele, caso contrário é aberto)? Justifique sua resposta.**

**INTEGRIDADE - um processo correto *p* entrega uma mensagem *m* no máximo uma vez.**

**ACORDO - se um processo correto entrega a mensagem *m*, então todos os outros processos corretos em group(*m*) distribuirão *m.***

***VALIDADE - se um processo correto executa um multicast da mensagem m, então, ele distribuirá m.***

**2 - Sugira como fazer para adaptar o algoritmo valentão para tratar com particionamentos temporários de rede (comunicação lenta) e processos lentos.**

**3 - Em MMOG (massively multiplayer online games) três estratégias de implementação são utilizadas: 1) adotar a arquitetura cliente-servidor e através de um único servidor central (cluster) atender os jogadores (clientes); 2) adotar arquitetura distribuída com múltiplos servidores, alocando jogadores a servidor em particular de acordo com critérios previamente definidos; 3) adotar uma arquitetura peer-to-peer, promovendo o uso de recursos do jogador. Em particular, quais vantagens você vê em adotar a estratégia de servidor único (1) para representar o estado do jogo para vários jogadores? Quais problemas você consegue identificar e como eles poderiam ser resolvidos?**

**4 - Um serviço é implementado por vários servidores. Explique por que recursos poderiam ser transferidos entre eles. Seria satisfatório para os clientes fazer multicast (difusão seletiva) de todos os pedidos para o grupo de servidores, como uma maneira de proporcionar transparência de mobilidade para os clientes?**

**5 - Explique o que significa "representação externa de dados" no contexto de sistemas distribuídos e o processo de empacotamento/desempacotamento de dados.**