

Preparação para PG2 de NF2

2º semestre de 2025

Fixação dos principais conceitos

Este questionário revisa os tópicos abordados na Avaliação 2: clusterização de servidores, balanceamento de carga, simulação de redes com Mininet, segurança em Kubernetes com NGINX Ingress Controller, orquestração de containers com Docker e Amazon ECS, além de algoritmos MPI para comunicação e problemas clássicos de sincronização, como o *Dining Philosophers*.

1. Qual é o principal benefício de usar um cluster de servidores?
 - (a) Reduzir custo de hardware
 - (b) Alta disponibilidade e escalabilidade
 - (c) Simplificar o código da aplicação
 - (d) Eliminar a necessidade de rede
2. Em um balanceamento estático, a distribuição de requisições:
 - (a) É ajustada em tempo real
 - (b) Não considera variação da carga
 - (c) Usa métricas de CPU e memória
 - (d) Exige monitoramento contínuo
3. O Mininet é utilizado para:
 - (a) Simular redes virtualmente
 - (b) Criar containers em produção
 - (c) Substituir Kubernetes
 - (d) Monitorar uso de CPU
4. Qual comando cria um container em background expondo a porta 8080?
 - (a) `docker ps`
 - (b) `docker run -d -p 8080:8080 imagem`
 - (c) `docker exec -it imagem bash`
 - (d) `docker build -t imagem .`
5. O que acontece ao desativar o Ingress Controller em um cluster?
 - (a) Melhora a performance

- (b) Interrompe roteamento de tráfego externo
 - (c) Aumenta a segurança sem riscos
 - (d) Remove todos os pods do cluster
6. Qual vantagem do uso de *task definitions* no ECS/K8s?
- (a) Execução apenas em uma máquina local
 - (b) Suporte a Auto Scaling e integração com load balancers
 - (c) Eliminação da rede entre containers
 - (d) Dispensa uso de imagens Docker
7. A função `MPI_Bcast` é usada para:
- (a) Enviar mensagem ponto a ponto
 - (b) Enviar dados do processo raiz para todos
 - (c) Criar processos paralelos
 - (d) Sincronizar todos os processos
8. O algoritmo em árvore binária no broadcast faz:
- (a) Raiz envia para todos ao mesmo tempo
 - (b) Raiz envia para 2 filhos, que repassam até todos receberem
 - (c) Mensagem percorre processos sequencialmente
 - (d) Cada processo envia para si mesmo
9. O que pode causar *deadlock* no jantar dos filósofos?
- (a) Todos pegarem primeiro o garfo da esquerda
 - (b) Usar comunicação assíncrona
 - (c) Sincronizar com `MPI_Barrier`
 - (d) Ordenar a aquisição de recursos
10. O `MPI_Barrier` serve para:
- (a) Evitar deadlock
 - (b) Sincronizar o ponto de execução de todos os processos
 - (c) Substituir `MPI_Send`
 - (d) Eliminar starvation