

Grupo Handbook

Handbook de Questões de TI

comentadas para CONCURSOS

Além do gabarito

2ª Edição

Volume 4

*Questões FCC
Parte 2*

Fundação Carlos Chagas

Prefácio

Este é o volume 4 da série *Handbook de Questões de TI Comentadas para Concursos – Além do Gabarito* que complementa o volume 3 ao trazer para você 50 questões elaboradas pela banca da Fundação Carlos Chagas, todas comentadas além do gabarito.

A Fundação Carlos Chagas (FCC) é uma instituição privada sem fins lucrativos, reconhecida como de utilidade pública nos âmbitos federal, estadual e municipal. Em quarenta e cinco anos de existência, com mais de 2.600 concursos realizados e mais de 33 milhões de candidatos em todo o território nacional, a FCC encontra-se plenamente habilitada a realizar todo tipo de concurso.

Atualmente, muitos dos principais concursos na área de TI são organizados por essa banca. Entre os concursos estão os do Banco Central, da Câmara dos Deputados e de inúmeros tribunais espalhados por todo país.

A primeira característica dos concursos de TI da FCC é o fato das questões serem diretas, sem que isso implique, no entanto, em uma prova mais fácil. A segunda principal característica dos concursos de TI da FCC é a abrangência multidisciplinar de seus editais, o que reflete a demanda dos órgãos públicos por profissionais qualificados e generalistas.

Levando em consideração tais características, a equipe Handbook de TI preparou os volumes 3 e 4 da série para você se preparar ainda melhor para os concursos de TI que estão por vir, especialmente os organizados pela FCC.

Bons estudos,

Grupo Handbook de TI

Direitos Autorais

Este material é registrado no Escritório de Direitos Autorais (EDA) da Fundação Biblioteca Nacional. Todos os direitos autorais referentes a esta obra são reservados exclusivamente aos seus autores.

Os autores deste material não proíbem seu compartilhamento entre amigos e colegas próximos de estudo. Contudo, a reprodução, parcial ou integral, e a disseminação deste material de forma indiscriminada através de qualquer meio, inclusive na Internet, extrapolam os limites da colaboração. Essa prática desincentiva o lançamento de novos produtos e enfraquece a comunidade concurseira Handbook de TI.

A série *Handbook de Questões de TI Comentadas para Concursos – Além do Gabarito* é uma produção independente e contamos com você para mantê-la sempre viva.

Grupo Handbook de TI

Canais de Comunicação

A equipe Handbook de TI disponibiliza diversos canais de comunicação para seus clientes.

Loja Handbook de TI

<http://www.handbookdeti.com.br>

Serviço de Atendimento

Comunicação direta com a Equipe Handbook de TI pode ser feita em
<http://www.handbookdeti.com.br/contacts>

Twitter do Handbook de TI

Que acompanhar de perto o trabalho do Grupo Handbook de TI. Cadastre-se no twitter e comece a seguir o grupo Handbook de TI em <http://twitter.com/handbookdeti>

1. Assuntos relacionados: *Programação, Algoritmos de Ordenação,***Banca:** *FCC***Instituição:** *TRT 15a Região***Cargo:** *Analista Judiciário - Tecnologia da Informação***Ano:** *2009***Questão:** *21*

São algoritmos de classificação por trocas apenas os métodos

- (a). SelectionSort e InsertionSort.
- (b). MergeSort e BubbleSort.
- (c). QuickSort e SelectionSort.
- (d). BubbleSort e QuickSort.
- (e). InsertionSort e MergeSort.

Solução:

Os algoritmos de ordenação são uns dos principais objetos de estudo na área de computação. Tais algoritmos tem por objetivo colocar os elementos de uma sequência em uma determinada ordem, sendo as mais utilizadas as ordens numéricas e alfabéticas.

Uma das principais razões para se ordenar uma sequência é permitir que os seus elementos sejam acessados de forma mais eficiente. Os métodos de ordenação mais conhecidos e utilizados são os seguintes:

- Ordenação por Troca
 - BubbleSort (Método da Bolha)
 - QuickSort (Método da Troca e Partição)
- Ordenação por Inserção
 - InsertionSort (Método da Inserção Direta)
 - BinaryInsertionSort (Método da Inserção Direta Binária)
- Ordenação por Seleção
 - SelectionSort (Método da Seleção Direta)
 - HeapSort (Método da Seleção em Árvore)
- Outros métodos
 - MergeSort (Método da Intercalação)
 - BucketSort (Método da Distribuição de Chave)
 - CountingSort (Método da Ordenação por Contagem)

Como podemos ver, dentre os métodos apresentados nas alternativas da questão, os únicos baseados na ordenação troca são o BubbleSort e o QuickSort. Portanto, a resposta da questão é a alternativa D. Tais algoritmos são classificados como de troca por que seus processos de ordenação se dão através de trocas entre pares de elementos do vetor. Embora o BubbleSort e o QuickSort se baseiem em trocas, seu funcionamento geral e o seu desempenho diferem substancialmente.

O BubbleSort é, talvez, o método simples de ordenação, que funciona através de sucessivas trocas entre pares de elementos do vetor. O método realiza varreduras no vetor, trocando pares adjacentes de elementos sempre que o próximo elemento for menor que o anterior.

Após uma varredura, o maior elemento está corretamente posicionado no vetor e não precisa mais ser comparado. Dessa forma, após a i -ésima varredura, os i maiores elementos estão ordenados. A complexidade algorítmica de tempo do BubbleSort é $O(n^2)$.

O Quicksort adota uma estratégia conhecida como divisão e conquista. Por isso, usualmente, o Quicksort é implementado utilizando-se o paradigma recursivo. No primeiro passo do QuickSort, um elemento da lista é escolhido como pivô. Em seguida, a lista é reorganizada (por meio de trocas de posição entre os elementos) de modo que todos os elementos anteriores ao pivô sejam menores que ele, e todos os elementos posteriores ao pivô sejam maiores que ele. Ao fim desse processo, que denomina-se particionamento, o pivô estará em sua posição final e haverá duas sublistas não ordenadas.

Em seguida, o algoritmo QuickSort é reaplicado a cada uma das sublistas, sendo a base da recursão as sublistas de tamanho zero ou um que, por definição, estão sempre ordenadas. O processo é finito, pois a cada iteração pelo menos um elemento é posto em sua posição final e não será mais manipulado na iteração seguinte. A complexidade algorítmica de tempo do QuickSort é $O(n \lg n)$.

A seguir, exemplos de implementação do bubble sort e do quicksort usando a linguagem de programação ruby.

```
#BubbleSort em Ruby
def bubblesort(list)
  return list if list.size <= 1 # already sorted
  loop do
    swapped = false
    0.upto(list.size-2) do |i|
      if list[i] > list[i+1]
        list[i], list[i+1] = list[i+1], list[i] # swap values
        swapped = true
      end
    end
    break unless swapped
  end
  return list
end

#QuickSort em Ruby
def quicksort (array)
  return array if array.size <= 1
  pivot = array[0]
  return quicksort (array.select {|y| y < pivot })
  + array.select { |y| y == pivot } +
  quicksort (array.select {|y| y > pivot })
end
```

2. Assuntos relacionados: *Programação, Estruturas de Dados, Tabela Hash,***Banca:** FCC**Instituição:** TRT 15a Região**Cargo:** Analista Judiciário - Tecnologia da Informação**Ano:** 2009**Questão:** 22

Uma estrutura de dados especial de armazenamento de informações, cuja ideia central é utilizar uma função que, quando aplicada sobre uma chave de pesquisa, retorna o índice onde a informação deve ser armazenada denomina-se

- (a). vetor de dispersão.
- (b). matriz de dispersão.
- (c). tabela hash.
- (d). árvore binária.
- (e). lista encadeada

Solução:

A resposta da questão é a alternativa C, tabela hash, que também é conhecida como tabela de espalhamento ou de dispersão.

A implementação de uma tabela hash se baseia na escolha de uma função, chamada função hash, que associe uma chave de pesquisa a um índice em uma estrutura de dados. O requisito mais importante de uma função hash é o de distribuir uniformemente as chaves pelos vários índices, de forma eliminar ou minimizar a chance de que mais de uma chave seja mapeada para o mesmo índice, problema conhecido como colisão.

Por mapearem uma chave de pesquisa a um determinado índice de forma direta, as tabelas hash proporcionam um tempo médio de busca constante, ou seja, $O(1)$. Em virtude de seu alto desempenho, as tabelas hash são tipicamente utilizadas na indexação de grandes volumes de informações em bancos de dados e na criação de esquemas associativos de acesso às memória cache. A Figura 1 mostra, de forma básica, como se dá o esquema de mapeamento de chaves em índices usando uma tabela hash.

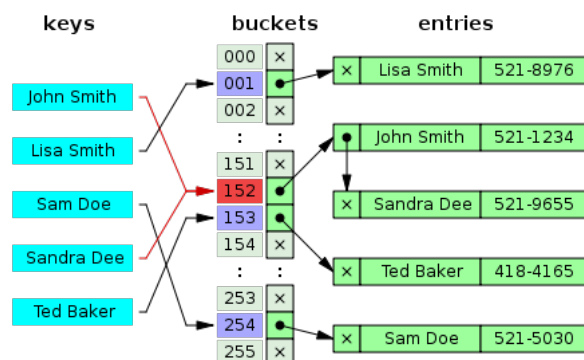


Figura 1: Tabela Hash.

No entanto, a definição de uma boa função hash nem sempre é uma tarefa simples, sendo um trabalho profundamente relacionado à estatística. Portanto, para otimizar a

função hash, é necessário conhecer a natureza da chave a ser utilizada e como os valores que a chave pode assumir se distribuem ao longo do domínio. Os métodos de implementação de funções hash mais comuns são o método da **divisão**, o método da **dobra**, e o método da **multiplicação**.

Com relação às limitações das tabelas hash, valem as seguintes observações. As tabelas hash são estruturas de dados que não permite armazenar elementos repetidos, recuperar elementos sequencialmente, nem recuperar antecessores ou sucessores de um determinado elemento já encontrado. Caso essas operações tenham muita importância no sistema, vale a pena considerar a utilização de estruturas de dados como listas encadeadas, árvores, entre outras.

3. **Assuntos relacionados:** *Segurança da Informação, ISO 27001, SGSI,*

Banca: FCC

Instituição: TRT 15a Região

Cargo: Analista Judiciário - Tecnologia da Informação

Ano: 2009

Questão: 27

Implementar e operar a política, os controles, os processos e os procedimentos do Sistema de Gestão de Segurança da Informação, segundo a norma ABNT 27001, são atividades associadas no modelo PDCA à fase

- (a). D = Do.
- (b). P = Plan.
- (c). C = Check.
- (d). C = Control.
- (e). A = Act

Solução:

A norma ISO 27001 especifica um conjunto de requisitos para o estabelecimento, implantação, monitoração, revisão, manutenção e melhoria de um Sistema de Gestão de Segurança da Informação, ou simplesmente, SGSI. Um SGSI pode também ser definido como um framework de políticas, procedimentos e controles de várias naturezas (físicos, lógicos, legais etc) que, por sua vez, fazem parte do processo de gerenciamento de riscos da organização como um todo.

Como todo sistema de gestão, o SGSI também é descrito em termos de processos, os quais são administrados por meio da metodologia PDCA (Plan-Do-Check-Act). Em última instância, o uso do PDCA na implementação de sistemas de gestão tem como objetivo principal garantir a melhoria contínua da gestão da segurança da informação, minimizando os riscos associados à segurança da informação.

O PDCA pode ser utilizado em qualquer atividade da organização. É ideal que todas as pessoas da organização utilizem a ferramenta de gestão no dia-a-dia de suas atividades.

Desta forma, elimina-se a cultura “tarefeira” que muitas empresas insistem em perpetuar e que incentiva a se realizar o trabalho sem antes planejar, desprezando o autocontrole, o uso de dados gerados pelas medições por indicadores e a atitude preventiva, para que os problemas dos processos nunca ocorram.

As etapas do ciclo PDCA são as seguintes:

- **PLAN:** O primeiro passo para a aplicação do PDCA é o estabelecimento de um plano. Essa etapa envolve o estabelecimento dos objetivos e do método que deve ser utilizado para conseguir alcançá-los. A elaboração de um bom plano evita falhas e retrabalho nos ciclos subsequentes;
- **DO:** O segundo passo do PDCA é a execução do plano. Essa etapa envolve o treinamento dos envolvidos e a execução propriamente dita do plano;
- **CHECK:** O terceiro passo do PDCA é a verificação dos resultados alcançados. Nessa fase podem ser detectados erros ou falhas;

- ACT: A última fase do PDCA é a realização das ações corretivas para as falhas encontradas no passo anterior. Após realizada a investigação das causas das falhas ou dos desvios no processo, deve-se repetir o ciclo PDCA para a correção das falhas, de modo que o sistema o processo possa melhorar cada vez mais.

Portanto, no caso da ISO 27001, “Implementar e operar a política, os controles, os processos e os procedimentos” corresponde a etapa **Do** do ciclo PDCA, pois tal atividade se trata da execução de algo foi definido na etapa de **Plan** da implantação do SGSI. Logo, a alternativa A deve ser marcada.