

Cap 2 - Questionário

Responda às seguintes perguntas em 2-3 frases cada:

martinson.freitas@gmail.com [Mudar de conta](#)



Salvando...

* Indica uma pergunta obrigatória

Enviar por e-mail *



Registrar **martinson.freitas@gmail.com** como o e-mail a ser incluído na minha resposta

1 - Quais são as principais vantagens das estruturas de dados que usam arrays como base? * 10 pontos

As estruturas baseadas em arrays permitem acesso direto e rápido aos elementos por índice, o que garante alta performance em leituras. Além disso, aproveitam melhor a memória por serem armazenadas de forma contígua. São ideais para buscas e iterações previsíveis.

2 - Por que as operações `add(i,x)` e `remove(i)` em um `ArrayStack` têm tempos de execução que dependem de `n` e `i` (ignorando `resize`)? * 10 pontos

Porque ao adicionar ou remover em uma posição "`i`", todos os elementos após "`i`" precisam ser deslocados para manter a ordem, o que pode levar o custo da operação para até $O(n)$ operações no pior caso. O custo das operações `add` e `remove` dependem de `i` (posição) e `n` (tamanho total).



3 - Explique o conceito de custo amortizado no contexto de redimensionamento de array.

* 10 pontos

O custo amortizado é uma média do custo de operações ao longo do tempo. No caso de redimensionamento de arrays, a maioria das inserções tem custo constante $O(1)$, mas ocasionalmente, quando o array está cheio, ele é redimensionado, o que custa $O(n)$. Mesmo assim, esse custo alto é distribuído entre várias inserções anteriores, resultando em um custo médio por operação ainda constante, ou seja, $O(1)$ amortizado.

4 - Como a aritmética modular é usada na implementação de um ArrayQueue?

* 10 pontos

A aritmética modular é útil para simular um array infinito, porque " $i \% \text{tam}(a)$ " sempre dá um valor no intervalo $[0, \text{tam}(a)-1]$. Isso trata o array "a" como um array circular em que os índices de array $> \text{tam}(a) - 1$ retornem para o início do array.

5 - Qual é a principal desvantagem de usar um ArrayStack para implementar uma fila FIFO?

* 10 pontos

A principal desvantagem é que remover do início de um ArrayStack exige deslocar todos os elementos restantes. Isso torna a operação de remoção $O(n)$ em vez de $O(1)$. Como uma fila FIFO depende de remoções eficientes na frente, o desempenho se torna inadequado.

6 - Descreva a estratégia do ArrayDeque para otimizar as operações $\text{add}(i, x)$ e $\text{remove}(i)$ para índices perto das extremidades.

* 10 pontos

O ArrayDeque escolhe a direção de deslocamento com base na posição i : se $i < n/2$, ele desloca os elementos para a esquerda; se não, desloca para a direita, obtendo custo de operação máxima de $O(1 + \min\{i, n-i\})$ para deslocar elementos.



7 - Como o DualArrayDeque combina dois ArrayStacks para implementar uma lista? * 10 pontos

O DualArrayDeque é uma estrutura inteligente que usa dois ArrayStacks, "front" e "back", para fornecer operações rápidas em ambos os extremos da lista, rebalanceando quando necessário para manter a eficiência.

8 - Explique o objetivo do método balance() em um DualArrayDeque. * 10 pontos

O método balance() redistribui os elementos entre o "front" e o "back", garantindo que cada um tenha ao menos $n/4$ elementos, se uma das extremidades ficar muito grande (ou pequena), a operação move os elementos entre as partes, afim de que contenham $n/2$ elementos.

9 - Qual é o principal problema de espaço que o RootishArrayStack tenta resolver? * 10 pontos

Resolve o problema de desperdício de espaço em arrays como o ArrayStack. Ele organiza os dados em blocos de tamanho variável, reduzindo o desperdício para $O(\sqrt{n})$, mantendo a eficiência em operações ($O(1)$ amortizado) enquanto otimiza o uso de memória.

10 - Como o RootishArrayStack distribui os elementos da lista entre seus blocos de array? * 10 pontos

Ele distribui os elementos em blocos de tamanho crescente (1, 2, 3, ..., r), onde cada bloco "i" contém "i + 1" elementos. Portanto, todos os blocos "r" contêm um total de : $1 + 2 + \dots + r = r(r + 1)/2$ elementos, garantindo acesso eficiente $O(1)$ por operação.

Enviar

[Limpar formulário](#)

Nunca envie senhas pelo Formulários Google.

Este formulário foi criado em FEN UERJ.
Este formulário parece suspeito? [Relatório](#)

