

Lista de Exercício Complexidade Big O

1. **Soma dos Pares:** Dado um array de números inteiros e um valor alvo, determine se existem dois números no array que somem o valor alvo.
2. **Elemento Mais Frequente:** Encontre o elemento mais frequente em um array de inteiros.
3. **Substrings Únicas:** Dado uma string, encontre o número de substrings únicas.
4. **Cruzamento de Listas:** Dado duas listas vinculadas, determine se elas se cruzam em algum ponto.
5. **Multiplicação de Matrizes:** Implemente um algoritmo para multiplicar duas matrizes.
6. **N-ésimo Elemento:** Dado uma lista vinculada, encontre o n-ésimo elemento a partir do final.
7. **Sequência Crescente:** Determine se é possível obter uma sequência crescente removendo apenas um elemento de um array.
8. **Permutações de Strings:** Escreva um algoritmo para imprimir todas as permutações possíveis de uma string.
9. **Caminho em Matriz:** Dado uma matriz de booleanos representando um labirinto, encontre um caminho do canto superior esquerdo ao canto inferior direito.
10. **Divisores de um Número:** Liste todos os divisores de um número inteiro dado.

Após implementar soluções para os problemas acima, os alunos devem analisar e determinar a complexidade de tempo de cada algoritmo em termos de notação big O. É importante lembrar que a complexidade pode variar com base na implementação específica escolhida pelo aluno. Coletar os tempos de várias execuções de cada uma das soluções, com entradas de tamanhos diferentes e colocá-los em uma tabela com o objetivo de gerar um gráfico. Comparar o formato do gráfico