

- Faça os exercícios à mão, identificando devidamente cada exercício.
- Submeta ao SIGAA (seção de tarefas) um único arquivo **PDF** com as suas soluções **escaneadas** (dica: aplicativo CamScanner no Android).
- Seja o mais formal e claro possível em todas as respostas.
- Os envolvidos em qualquer tipo de cola serão automaticamente reprovados (conceito F).
- Essa lista vale **até 2 pontos**, que serão somados à nota da Prova 1. A nota máxima possível na Prova 1 é 10.

1. Utilizando as definições de notação assintótica, prove os seguintes itens:
 - (a) $27 \log n + \sqrt{n} = \Theta(\sqrt{n})$
 - (b) $\log(\sqrt{n}) = \Omega(100 \log(n^{5/4}))$
 - (c) $2^{an} \neq O(2^n)$, com $a \geq 1$
 - (d) $\pi \log(n^{60}) = O(\sqrt{n})$
 - (e) $7n^2 \neq O(n)$
 - (f) $(n+a)^b = \Theta(n^b)$, com $a, b \geq 1$
2. Considere os seguintes algoritmos recursivos que resolvem um mesmo problema em uma entrada de tamanho n :

Algoritmo A: Divide o problema em 2 partes de tamanho $n/4$ cada e gasta tempo adicional n .

Algoritmo B: Divide o problema em 9 partes de tamanho $n/3$ cada e gasta tempo adicional n .

Algoritmo C: Divide o problema em 6 partes de tamanho $n/5$ cada e gasta tempo adicional \sqrt{n} .

No que segue, assuma $T(1) = 1$. Para cada um dos três algoritmos, monte uma recursão para seu tempo de execução $T(n)$ e a resolva com o melhor limitante possível (notação O ou Θ) usando:

- (a) **Algoritmo A:** Método da substituição.
 - (b) **Algoritmo B:** Método iterativo.
 - (c) **Algoritmo C:** Método mestre.
3. Seja $A[1..n]$ um vetor com n elementos quaisquer. Descreva e explique o funcionamento de um algoritmo que verifica se A é um heap máximo em tempo $\Theta(n)$ no pior caso. Justifique porque seu algoritmo executa nesse tempo.
 4. Considere o problema de verificar se um dado grafo G é conexo.
 - (a) Crie um algoritmo que recebe dois vértices u e v de G e verifica se existe caminho entre u e v .
 - (b) Crie um algoritmo que recebe o grafo G e utiliza o algoritmo acima para verificar se G é conexo.
 - (c) Analise o tempo de execução dos seus algoritmos.
 5. Os gatos Pi, Mi e Sigma desenvolveram, respectivamente, algoritmos X , Y , e Z para resolver o **problema de ordenação**.
 - Pi afirma que seu algoritmo é eficiente pois o tempo de execução do algoritmo X é $\Omega(n)$ no pior caso. Explique por que essa afirmação não faz sentido.
 - Sabe-se que os tempos de execução de X , Y e Z no pior caso são, respectivamente, $O(n \log n)$, $O(n\sqrt{n})$, e $\Theta(n)$. Com essas informações é possível saber qual dos três algoritmos é assintoticamente mais eficiente no pior caso? **Justifique sua resposta.**