

Lista Avaliativa 5

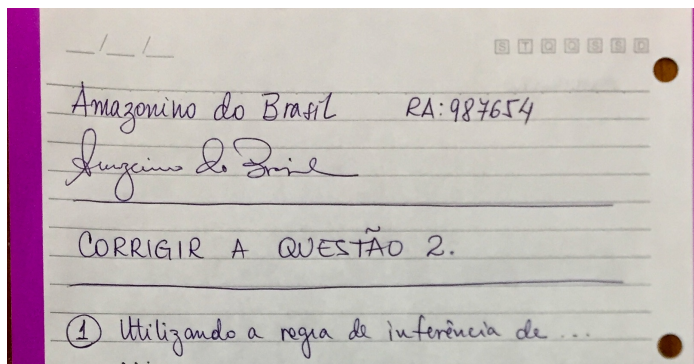
MC458 — Projeto e Análise de Algoritmos I

Prof. Pedro J. de Rezende

2º Semestre de 2020

Instruções

1. Por se tratar de avaliação de conhecimentos adquiridos por cada aluno, a resolução desta Lista Avaliativa deve ser um trabalho individual sem consulta direta ou indireta a outras pessoas.
2. Como a interpretação dos enunciados é parte integrante da resolução, tentativas de obtenção de auxílio para essas interpretações serão consideradas fraudes e será aplicado o que é previsto no item 3. abaixo.
3. QUALQUER TENTATIVA DE COLA OU FRAUDE ACARRETARÁ NOTA ZERO NESTA LISTA PARA TODOS OS IMPLICADOS, ALÉM DAS SANÇÕES PREVISTAS NO REGIMENTO GERAL DA UNICAMP (EM PARTICULAR, O ART. 227, INCISO VII, E OS ART. 228 A 231). SE VOCÊ PENSA QUE TALVEZ UMA AÇÃO SEJA CONSIDERADA FRAUDE, PROVAVELMENTE SERÁ.
4. Das três questões desta Lista, apenas duas serão corrigidas e valerão um total de 10 pontos.
 - Indique **exatamente UMA** das questões para ser corrigida pelo PED, a qual valerá nota de 0 a 5.
 - A segunda questão a ser corrigida será escolhida pelo PED, a qual também valerá nota entre 0 e 5. Se alguma questão estiver em branco, esta será a escolhida pelo PED.
5. O prazo **final** para submissão das resoluções se encerrará às 23hs do dia indicado no Google Classroom. Envios realizados após esse horário serão considerados atrasados. Se o atraso for de **até 2hs** após o encerramento do prazo regular de submissão, as resoluções submetidas serão corrigidas e receberão nota integral. Resoluções enviadas com **mais de 2hs de atraso**, mas tardias em **não mais do que 8hs** ainda serão corrigidas e receberão nota, mas com **50% de penalidade**. Submissões com atraso superior a 8hs automaticamente receberão nota zero.
6. **Importante:** note que a submissão não se completa apenas com o *upload* de um arquivo, mas **requer** o acionamento do botão “Entregar”. Como é essa ação que determina a hora do encaminhamento, sem ela a submissão não está efetivada e será desconsiderada.
7. **Justifique cuidadosamente todas as respostas.**
8. **Só serão aceitas** submissões de resoluções desta Lista Avaliativa na plataforma Google Classroom, e elas devem seguir **estritamente** o seguinte formato:
 - (a) As resoluções devem ser **manuscritas**, sem rasuras, escaneadas, formando **um único documento PDF** cujo nome deve ser **LEA#NNNNNN.pdf**, onde # é o número da presente LEA e NNNNNN é o seu R.A.
 - (b) No topo da primeira página das suas resoluções, coloque seu nome e RA de forma bem legível e, em seguida, a sua assinatura conforme esta consta em seu RG ou CNH. Veja modelo abaixo:



- (c) É sua responsabilidade **garantir** que o arquivo escaneado seja **claramente legível**. Para isso, recomenda-se o uso de um aplicativo para celular (**Android** ou **iOS**) como **Adobe Scan** (ou **CamScanner** ou **Office Lens** ou similar) para escanear as páginas manuscritas e, em seguida, fazer os devidos ajustes de contraste. Esses Apps facilitam a inclusão de múltiplas páginas em um único PDF. Todas as páginas devem estar na posição “retrato”.
- (d) Submissões constituídas meramente de arquivos de fotos (**jpg**, **png**, etc.), serão desconsideradas e receberão nota zero.

-
1. Dadas três cadeias de caracteres de um alfabeto Σ : X de tamanho n , Y de tamanho m e Z de tamanho $n + m$, deseja-se determinar se Z é uma intercalação de X e Y .

Diz-se que Z é uma intercalação de X e Y se Z pode ser decomposta em duas subcadeias disjuntas, sendo uma delas X e a outra Y . Por exemplo, $n = 11$, $X = \text{'programacao'}$, $m = 8$, $Y = \text{'dinamica'}$, $Z = \text{'pr**do**ingram**am**ac**ia**oca'}$. Vê-se que, neste caso, Z é uma intercalação de X e Y (os elementos de X em Z estão destacados em negrito).

Nesta questão, você deve projetar um algoritmo de programação dinâmica recursivo **com memorização** (*top-down*) que, dados n , m e as cadeias X de tamanho n , Y de tamanho m e Z de tamanho $n + m$, retorna 1 se Z for uma intercalação de X e Y e 0 caso contrário.

Sequência obrigatória:

- (a) (50% do valor da nota): Prove que o problema tem subestrutura ótima, deduzindo uma fórmula de recorrência que poderá ser usada para projetar seu algoritmo. Explique o que é cada termo da recorrência. Indique os casos base e seus valores.
- (b) (30% do valor da nota) Depois (e só depois de fazer o item (a)), descreva em alto nível (ou em pseudo-código) o seu algoritmo.
- (c) (20% do valor da nota) Em seguida, faça uma análise de complexidade de tempo e de espaço (em função de n e m) de seu algoritmo.

Se você quiser incluir um exemplo (apenas para fins didáticos), que ele seja **pequeno** e ilustrativo, mas **NÃO** se refira a ele nem na demonstração de que o problema tem subestrutura ótima, nem na descrição do algoritmo, nem na análise de complexidade.

-
2. Descendo-se o Rio Amazonas, encontram-se n aldeias, A_1, A_2, \dots, A_n . Você está na aldeia A_1 e precisa chegar à aldeia A_n para tomar seu voo de volta para Campinas. Na ausência de estradas, é preciso fazer o trajeto em canoas que podem ser alugadas nas várias aldeias. Antes de iniciar sua jornada rio abaixo, você recebe uma tabela com os preços dos alugueis das canoas que se podem alugar nas várias cidades. Para cada i, j , $1 \leq i < j \leq n$, a tabela indica o custo $t_{ij} > 0$ para se alugar uma canoa na aldeia A_i a ser devolvida na aldeia A_j . Os valores são arbitrários, i.e., não relacionados à distância entre a origem e o destino, nem ao tempo de descida pelo rio, ou qualquer outro critério que você possa imaginar. Por exemplo, é possível que $t_{1,3} = 4$, $t_{3,4} = 16$, $t_{1,4} = 15$, $t_{1,2} = 7$ e $t_{2,4} = 3$.

Nesta questão, você deve projetar um algoritmo de programação dinâmica **sem recursão** (*bottom-up*) de complexidade de tempo $O(n^2)$ que determina o **custo mínimo** de alugueis de canoas com que você precisa arcar para conseguir descer o Rio Amazonas da aldeia A_1 até a aldeia A_n .

Sequência obrigatória:

- (a) (50% do valor da nota): Prove que o problema tem subestrutura ótima, deduzindo uma fórmula de recorrência que poderá ser usada para projetar seu algoritmo. Explique o que é cada termo da recorrência. Indique os casos base e seus valores.
- (b) (30% do valor da nota) Depois (e só depois de fazer o item (a)), descreva em alto nível (ou em pseudo-código) o seu algoritmo.
- (c) (20% do valor da nota) Em seguida, faça uma análise de complexidade de tempo e de espaço (em função de n) de seu algoritmo.

Se você quiser incluir um exemplo (apenas para fins didáticos), que ele seja **pequeno** e ilustrativo, mas **NÃO** se refira a ele nem na demonstração de que o problema tem subestrutura ótima, nem na descrição do algoritmo, nem na análise de complexidade.

-
3. Descreva como modificar o seu algoritmo da Questão 2. para se obter a sequência de aldeias onde você deverá devolver canoas alugadas durante sua descida pelo Rio Amazonas.
-