

Universidade Federal de Ouro Preto - UFOP
Departamento de Ciência da Computação - DECOM

Relatório atividade 11 - Adventures in Moving - Part IV

BCC402 - ALGORITMOS E PROGRAMACAO AVANCADA

Kayo Xavier Nascimento Cavalcante Leite - 21.2.4095

Professor: Rafael Alves Bonfim

Ouro Preto
31 de março de 2025

Sumário

1	Código e enunciado.	1
2	Problema 10201: Adventures in Moving - Part IV	1
2.1	Descrição do Problema	1
2.2	Entrada e Saída	1
2.3	Estratégia de Solução	1
2.4	Análise Matemática	2
3	Casos teste - Input e output esperado.	2

Lista de Códigos Fonte

1	Pseudocódigo do problema.	2
---	-----------------------------------	---

1 Código e enunciado.

Na Atividade 11 - o problema selecionado foi Adventures in Moving - Part IV. O objetivo é calcular o custo mínimo de combustível para uma viagem de caminhão de Waterloo até uma cidade, considerando restrições de capacidade do tanque (200 litros) e exigências de combustível no início (meio tanque) e no final (pelo menos meio tanque). O caminhão consome 1 litro por quilômetro, e o desafio é abastecer estrategicamente nos postos ao longo da rota para minimizar o gasto total posicionadas para um dado N . O código comentado e documentado, casos de teste e executável pré compilado se encontram no .zip da atividade. O código foi feito com base na referência encontrada no site:

<https://github.com/Diusrex/UVA-Solutions/blob/master/10201%20Adventures%20in%20Moving.cpp>

Caso queira, para rodar e compilar o código, é necessário ter o compiler g++ e utilizar o seguinte comando no terminal dentro do diretório da pasta da atividade específica:

Compilando e rodando o exercício

```
para compilar:
g++ moving.cpp -o executavel

e para rodar basta utilizar .\executavel no cmd.

para utilizar os cenários de teste:
.\executavel < sampleinput.txt
.\executavel < testinput.txt
```

2 Problema 10201: Adventures in Moving - Part IV

2.1 Descrição do Problema

O objetivo é calcular o custo mínimo de combustível para uma viagem de caminhão de Waterloo até uma cidade, considerando restrições de capacidade do tanque (200 litros) e exigências de combustível no início (meio tanque) e no final (pelo menos meio tanque). O caminhão consome 1 litro por quilômetro, e o desafio é abastecer estrategicamente nos postos ao longo da rota para minimizar o gasto total.

2.2 Entrada e Saída

- **Entrada:**
 - Número de casos de teste T .
 - Para cada caso: distância total (km) e lista de postos (distância de Waterloo e preço por litro em décimos de centavo).
- **Saída:** Custo mínimo em décimos de centavo ou "Impossible" se não for viável.

2.3 Estratégia de Solução

Utiliza-se **Programação Dinâmica (PD)** para modelar os estados de combustível em cada posto:

1. Definição de Estados:

- $bestCost[i][j]$: menor custo para chegar ao posto i com j litros restantes.

2. Recursão com Memoização:

- Para cada posto i e combustível j , calcula-se o custo mínimo abastecendo quantidades viáveis para alcançar o próximo posto.
- Considera-se apenas postos válidos (distância \leq destino final).

3. Condições de Contorno:

- Ao chegar no destino, o combustível deve ser ≥ 100 litros.
- Estados inválidos (combustível negativo) são descartados.

2.4 Análise Matemática

Seja d_i a distância do posto i e p_i o preço no posto. O custo para ir do posto i ao $i + 1$ com j litros é:

$$\text{Custo}(i, j) = \min_{\substack{k \geq \max(j, d_{i+1} - d_i) \\ k \leq 200}} (\text{Custo}(i + 1, k - (d_{i+1} - d_i)) + (k - j) \cdot p_i)$$

onde k é a quantidade de combustível após abastecer no posto i .

```
1 Algoritmo CustoMinimoViagemResumido
2
3 // --- Objetivo Geral ---
4 // Calcular o custo minimo de uma viagem ate um destino, considerando:
5 // - Distancia a percorrer.
6 // - Postos de gasolina com precos diferentes.
7 // - Capacidade limitada de combustivel (max 200).
8 // - Condicao inicial (comeca com 100) e final (termina com >= 100).
9
10 // --- Funcao Principal de Calculo (usando DP/Memoizacao) ---
11 // Funcao CalcularCusto(indice_posto_atual, combustivel_atual):
12 //   - Determina o menor custo para chegar ao destino final a partir do '
13     posto_atual',
14 //     chegando nele com 'combustivel_atual'.
15 //   - Considera o custo de abastecer (ou nao) no posto atual e o custo
16 //     recursivo de continuar a viagem a partir do proximo posto.
17 //   - Utiliza memoria para armazenar resultados ja calculados e evitar
18     redundancia.
19 //   - Trata casos base: chegar ao destino (com combustivel suficiente ou nao)
20 //     ou ficar sem combustivel no caminho.
21 // Fim Funcao CalcularCusto
22
23 // --- Rotina Principal de Execucao ---
24 // Funcao Principal():
25 //   - Le os dados da viagem (destino, postos, precos).
26 //   - Prepara as estruturas de dados (lista de postos validos, memoria DP).
27 //   - Calcula o estado inicial (combustivel ao chegar no primeiro posto).
28 //   - Chama a 'Funcao CalcularCusto' para o estado inicial.
29 //   - Imprime o resultado final (custo minimo encontrado ou "Impossible").
30 // Fim Funcao Principal
```

Código 1: Pseudocódigo do problema.

3 Casos teste - Input e output esperado.

Para os casos de teste do problema, foi disponibilizado junto a pasta do mesmo os seguintes arquivos :sampleinput.txt e testinput.txt, sendo o primeiro o próprio caso de teste disponibilizado pelo exercício e o segundo caso de teste encontrado na plataforma <https://www.udebug.com/>. Além disso, encontra-se também o arquivo com os outputs esperados para cada input. Ambos os resultados foram validados e tiveram o output esperado.

sampleinput.txt

```
1
500
100 999
150 888
200 777
300 999
400 1009
450 1019
500 1399
```

output esperado

```
450550
```

Os demais testes se encontram no diretório da atividade