

TRABALHO PRÁTICO 02 IMPLEMENTAÇÃO DO ALGORITMO SIMPLEX

Objetivo.

Este trabalho tem como objetivo desenvolver as competências práticas na implementação de algoritmos relacionados à Pesquisa Operacional, especificamente para a resolução de problemas de programação linear. Este trabalho possibilitará ao aluno aplicar os conhecimentos teóricos, desenvolvendo habilidades de programação e análise de algoritmos, além de aprofundar o entendimento sobre o funcionamento do algoritmo Simplex.

Descrição.

Você deverá implementar o algoritmo Simplex para resolver problemas de programação linear. O programa deverá ser capaz de ler um arquivo texto contendo a especificação de um problema de programação linear (um problema de minimização escrito na forma padrão) e, a partir da especificação do problema, executar o algoritmo Simplex para encontrar a solução ótima.

Requisitos do Programa.

1. O caminho para o arquivo texto com a especificação do problema será passado para o programa como um argumento de linha de comando.
2. A implementação do algoritmo Simplex não precisará lidar com casos especiais: soluções degeneradas (que podem levar a um ciclo infinito), solução ilimitada (em que o valor da função objetivo pode diminuir infinitamente) e problemas inviáveis (problemas com solução inexistente).
3. A cada iteração do algoritmo Simplex, o programa deverá apresentar o número da iteração, o valor da função objetivo e o tempo transcorrido (em segundos) até o momento.
4. Ao final da execução do algoritmo, o programa deve informar que a solução ótima foi encontrada, apresentando o tempo total de execução, o valor da função objetivo para a solução ótima e os valores das variáveis de decisão da solução ótima.
5. Os valores de função objetivo, tempo transcorrido e variáveis de decisão devem ser apresentados com quatro casas decimais.
6. Poderão ser utilizadas as linguagens de programação C, C++, Java ou Python para a implementação do algoritmo Simplex. Para utilizar uma linguagem de programação diferente das citadas anteriormente, o professor a disciplina deverá ser consultado, podendo aprovar ou rejeitar a solicitação após análise.
7. É permitido o uso de bibliotecas fornecidas pela própria linguagem de programação ou bibliotecas de terceiros para manipulação de matrizes e resolução de sistemas lineares. No entanto, não é permitido o uso de implementações prontas do algoritmo Simplex.

Estrutura do Arquivo de Entrada.

O arquivo texto de entrada deverá obedecer à seguinte estrutura:

- A primeira linha do arquivo contém dois valores inteiros, n e m , separados por um espaço. O valor n representa o número de variáveis de decisão e o valor m representa o número de restrições (excluindo as restrições de não negatividade das variáveis).
- A segunda linha do arquivo contém n números reais, separados por espaços, representando os coeficientes das variáveis de decisão na função objetivo.
- As m linhas seguintes representam as restrições. Cada linha terá $n + 1$ números reais, onde os primeiros n números são os coeficientes das variáveis na restrição, e o último número é o termo independente da restrição.

Exemplo de Arquivo de Entrada e Saída Esperada.

Considere o problema de programação linear de minimização e já escrito na forma padrão:

$$\begin{aligned} \text{Minimizar } & -180x_1 - 300x_2 \\ \text{sujeito a: } & x_1 + x_3 = 60 \\ & x_2 + x_4 = 50 \\ & x_1 + x_2 + x_5 = 120 \\ & x_1, \dots, x_5 \geq 0 \end{aligned}$$

O quadro abaixo apresenta o arquivo texto de entrada que descreve o problema de programação linear acima e um exemplo da saída esperada que deverá ser apresentado pelo programa.

Arquivo de entrada	Saída esperada
5 3 -180 -300 0 0 0 1 0 1 0 0 60 0 1 0 1 0 50 1 1 0 0 1 120	Iteração: 1 Tempo(s): 0.2130 Objetivo: 0.0000 Iteração: 2 Tempo(s): 0.4144 Objetivo: -15000.0000 Iteração: 3 Tempo(s): 0.6159 Objetivo: -25800.0000 Solução ótima encontrada em 0.6200 segundos! Função objetivo é -25800.0000. Solução: x[1] = 60.0000 x[2] = 50.0000 x[3] = 0.0000 x[4] = 0.0000 x[5] = 10.0000

Critérios de Avaliação.

- Corretude do algoritmo Simplex implementado.
- Desempenho da implementação do algoritmo implementado.
- Capacidade do programa de ler o arquivo de entrada corretamente.
- Capacidade do programa de apresentar a saída corretamente.
- Qualidade de código, incluindo a legibilidade (indentação, nomes de variáveis, ...) e comentários.

Formato de Entrega.

Deve ser entregue um único arquivo compactado no formato ZIP. O nome do arquivo compactado deve seguir o padrão `nome-sobrenome-matricula.zip`, por exemplo `andre-maravilha-123456.zip`. Esse arquivo deverá ter o seguinte conteúdo:

- Um diretório `src` com o código-fonte.
- Um arquivo `LEIAME.txt` com instruções de compilação (no caso de linguagens compiladas) e execução.