

MC859 - Projeto em Teoria da Computação

SEGUNDO SEMESTRE DE 2023

TURMA A: PROF. FLÁVIO K. MIYAZAWA, fk@ic.unicamp.br

Ementa. Implementação de um projeto prático na área de Teoria da Computação.

Aulas e Atendimento. As aulas serão nas sextas-feiras das 14:00 às 18:00. O atendimento será feito logo após as aulas.

Avaliação. Nesta disciplina, serão oferecidos alguns problemas NP-difíceis (pelo menos dois). Cada aluno deverá escolher um deles. Alguns dos problemas poderão ter restrições de roteamento, ou de classificação/particionamento.

Independente do problema escolhido, a resolução do problema envolverá a resolução por duas abordagens, uma heurística/metaheurística e outra exata.

Parte Heurística: Fazer pelo menos 2 heurísticas construtivas aleatorizadas (como gulosos que foram aleatorizados) e pelo menos 1 heurística de melhoria (como busca local), além de uma heurística final que deve chamar as heurísticas construtivas e a heurística de melhoria, para obter soluções viáveis boas para o problema (o aluno é responsável por montar a combinação da heurística final através das heurísticas construtivas e de busca local). Alternativamente, pode-se implementar uma metaheurística que envolva uma abordagem de busca mais sofisticada.

I_H (IMPLEMENTAÇÃO - PESO 3): Será considerado o desempenho da(s) implementações desta parte em relação às entradas disponibilizadas, grau de complexidade de implementação, documentação e comentários no código. Junto com todos os códigos fonte da implementação, necessários para compilar os códigos, o aluno deverá entregar as entradas utilizadas nos testes computacionais divulgados no relatório.

R_H (RELATÓRIO - PESO 1): Relatório da implementação desta parte. Deve conter a descrição das rotinas implementadas, com explicações em alto nível e também descrição em pseudo-código (considerando passos de nível mais alto, sem descrever a descrição de rotinas simples ou bem conhecidas nos cursos de projeto e análise de algoritmos), descrição das estruturas de dados utilizadas, relato da complexidade computacional dos algoritmos implementados. O relatório deve conter uma seção de experimentos computacionais, onde o desempenho do código implementado deve ser apresentado para várias entradas disponibilizadas, mostrando os tipos de entrada para as quais a implementação obtém bons resultados e em quais entradas a implementação realizada não obteve bons resultados.

V_H (VÍDEO - PESO 1): O vídeo precisa conter uma apresentação do relatório e da execução do programa, por exemplo, feito na plataforma Google Meet, de até 15 minutos. O vídeo deve conter uma apresentação com explicações sobre cada uma das partes mencionadas no item anterior.

Parte Exata: Fazer algoritmo exato, por programação linear inteira para resolver o problema. O programa deve necessariamente utilizar alguma heurística, possivelmente as que foram utilizadas na primeira parte, para obter limitantes para as soluções viáveis, ou mesmo utilizar métodos heurísticos para obter soluções viáveis baseados nos programas lineares. Para a implementação do algoritmo exato, vamos usar o Gurobi, que é livre para a área acadêmica e facilita a implementação de programas lineares.

I_E (IMPLEMENTAÇÃO - PESO 3): Será considerado o desempenho da(s) implementações desta parte em relação às entradas disponibilizadas, grau de complexidade de implementação,

documentação e comentários no código. Junto com todos os códigos fonte da implementação, necessários para compilar os códigos, o aluno deverá entregar as entradas utilizadas nos testes computacionais divulgados no relatório.

R_E (RELATÓRIO - PESO 1): Relatório da implementação desta parte. Deve conter a descrição das rotinas implementadas, com explicações em alto nível e também descrição em pseudo-código (considerando passos de nível mais alto, sem descrever a descrição de rotinas simples ou já conhecidas), descrição das estruturas de dados utilizadas, relato da complexidade computacional dos algoritmos implementados. O relatório deve conter uma seção de experimentos computacionais, onde o desempenho do código implementado deve ser apresentado para várias entradas disponibilizadas, mostrando os tipos de entrada para as quais a implementação obtém bons resultados e em quais entradas a implementação realizada não obteve bons resultados.

V_E (VÍDEO - PESO 1): O vídeo precisa conter uma apresentação do relatório e da execução do programa, por exemplo, feito na plataforma Google Meet, de até 15 minutos. O vídeo deve conter uma apresentação com explicações sobre cada uma das partes mencionadas no item anterior.

Experimentos. Serão disponibilizadas algumas entradas. Algumas entradas serão de pequeno porte, para direcionar o formato de entrada e ter os primeiros resultados das implementações. E entradas de médio e grande porte, para testar e comparar melhor as implementações exatas e heurísticas. Para as entradas de pequeno porte, é esperado que todos os programas obtenham bons resultados sobre elas. Implementações que não obtiverem bons resultados nestas instâncias, terão nota de implementação na correspondente parte limitado a 3.0. Programas que não compilarem ou não executarem para entradas de pequeno porte, terão nota de implementação na correspondente parte igual a 0.0.

Bônus. Pelo menos as duas melhores implementações da turma, de cada parte, ganham bônus de 0.5 ou 1 ponto na parte correspondente a Implementação.

Prazos. Para cada parte, heurística/metaheurística ou exata, será entregue o enunciado do trabalho com data de entrega com prazo de pelo menos 3 semanas (a partir da data de entrega do enunciado) para sua preparação e entrega por email para o professor. Para cada parte, o material enviado para o professor deve conter: código fonte da implementação, relatório e vídeo.

Média Final. A média final, MF , será calculada como

$$MF \leftarrow \frac{3I_H + R_H + V_H + 3I_E + R_E + V_E}{10}$$

O aluno terá se aprovado se $MF \geq 5.0$, caso contrário terá se reprovado.

Observações. Qualquer tentativa de fraude acarretará nota zero na disciplina para todos os implicados.