

# Trabalho Prático - GCC218

---

04 Outubro 2019 / Semestre 02

## Introdução

O planejamento de horários em instituições de ensino constitui um desafio para diretores e coordenadores pedagógicos. De fato, trata-se de uma tarefa passível a erros, devido ao seu caráter combinatório, e que exige demasiado tempo de elaboração. O problema consiste em designar horários a turmas já pré-estabelecidas, levando em conta as seguintes restrições obrigatórias:

- não é permitida a alocação de duas aulas para um mesmo professor no mesmo horário;
- não pode haver duas aulas para uma mesma turma no mesmo horário;
- todas as aulas devem ser alocadas ao longo dos dias de funcionamento da escola e turnos de aula pré-determinados;
- professores não devem ser alocados em horários nos quais eles não podem estar presentes.

Algumas características são consideradas desejáveis, variando de acordo com as recomendações pedagógicas e de cunho pessoal. Tais particularidades são listadas a seguir:

- uma mesma turma não deverá ter três ou mais aulas geminadas da mesma disciplina, em horários sequenciais;
- não deverá haver horários de aula separados por grandes janelas entre aulas, para uma mesma turma. Exemplo: suponha que durante um dia, os alunos tivessem 5 horários de aulas.

Então, considere que no 1° e 2° horários houvesse aula, depois no 3° horário houvesse uma janela, e nos 4° e 5° horários os alunos voltassem a ter aulas. Logo, a solução sofreria uma penalidade. O ideal seria ter 4 aulas sequenciais, e o 5° horário livre.

- deve-se buscar atender às preferências de cada professor em relação a dias ou horários em que possa lecionar.

A atividade em questão pode ser automatizada, e você, detentor de conhecimentos em Algoritmos e em Teoria de Grafos, pode contribuir para auxiliar os responsáveis pela elaboração de horários.

### Objetivo

Desenvolva um algoritmo eficiente que encontre a distribuição de horários para cada turma, minimizando o número de restrições desejáveis não atendidas. Para tanto, implemente uma abordagem baseada em **coloração de vértices**.

### Prêmio

- Uma premiação extra será dada aos três melhores trabalhos entregues (caso haja mais de 3 submissões).
  - Primeiro lugar: 3 pontos extras.
  - Segundo lugar: 2 pontos extras.
  - Terceiro lugar: 1 ponto extra.
- Caso haja apenas 3 submissões de trabalhos, apenas o primeiro lugar será o premiado.
- Critérios de avaliação: (a) organização do código; (b) qualidade da solução; (c) tempo de execução do algoritmo;

(d) criatividade na elaboração do algoritmo; (e) qualidade do relatório interativo (dinâmico) entregue.

### Regras

- O código será implementado em C++ ou Python.
- O grupo será formado por até 3 alunos.
- Os grupos deverão apresentar sua implementação em um único arquivo, cujo nome deve seguir o seguinte padrão: "nomeMembroSubmeteTrabalho-MatriculaDesseAluno.py"(se for em Python). Exemplo: "mayron-123456.py". O nome do membro deve estar com todas as letras minúsculas.
- O código fonte deve conter em seu cabeçalho o nome de todos os membros do grupo, com suas respectivas matrículas.
- O grupo também deverá apresentar sua solução em termos de um relatório dinâmico, utilizando ferramentas tais como Jupyter Notebooks (<https://jupyter.org/>) ou RMarkdown (<https://rmarkdown.rstudio.com/>).
- Os dois arquivos (relatório dinâmico e arquivo fonte) devem ser compactados e submetidos por apenas um membro do grupo no Campus Virtual.
- Como foi exigido apenas um arquivo, sejam claros e objetivos no código.
- O grupo que não seguir alguma regra será penalizado em ao menos 1 ponto por cada infração cometida.
- Data de entrega: 07/12/19, às 23h55.
- Trabalhos entregues no dia 08/12/19 passarão a valer no máximo 14 pontos; caso o grupo entregue a atividade

avaliativa no dia 09/12/19, o trabalho passará a valer 8 pontos; trabalhos após o dia 09/12/19 não serão aceitos.

- Qualquer cópia ou plágio acarretará em nota nula da atividade para todos os membros do grupo, e consequências em órgãos competentes da UFLA.
- Os arquivos de teste para o trabalho serão disponibilizados no Campus Virtual.