

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE CIÊNCIAS EXATAS E DA TERRA
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA

DIM0549 - GRAFOS - TURMA 2
PROF. BRUNO MOTTA DE CARVALHO

Trabalho 3 - 14/10/2025 - Trabalho em grupos com até 3 componentes

Entrega: 10/12/2025 Horário: Até 23:59 Local: SIGAA

Preâmbulo: O algoritmo tem que ser totalmente implementados por vocês utilizando C/C++, Java ou Python. Você pode usar a linguagem de descrição DOT dos trabalhos anteriores ou simplesmente codificar usando a representação matricial do grafo da Figura 1 dado na Tabela 1.

1. Coloração em Grafos

A coloração de grafos é a atribuição de “cores” aos vértices ou arestas de um grafo de tal forma que dois vértices adjacentes (conectados por uma aresta) ou arestas adjacentes nunca tenham a mesma cor. O objetivo principal é encontrar a coloração que utiliza o menor número possível de cores, chamado número cromático ($\chi(G)$). Esse problema é NP-difícil, o que significa que não existe um algoritmo eficiente para encontrar a solução ótima para todos os grafos.

Aqui vocês vão trabalhar com um algoritmo heurístico que usa a técnica gulosa para obter uma solução aproximada. Ele não garante que encontraremos o valor exato de $\chi(G)$. O algoritmo DSATUR (Degree of Saturation) escolhe o próximo vértice a ser colorido baseado nos graus de saturação dos vértices não coloridos e prioriza vértices com mais vizinhos em caso de empate.

Sua tarefa consiste em implementar o DSATUR e o utilizar para calcular a coloração no grafo mostrado na Figura 1. A representação matricial do grafo pode ser vista na Tabela 1.

Escreva um relatório, explicando como executar seu programa, e explicando as escolhas de sua implementação. Fale sobre a complexidade de sua implementação do algoritmo DSATUR.

```

Input: Um grafo  $G = (V, E)$ 
Output: Uma coloração válida  $C$  para todos os vértices  $v \in V$ 
// Initialização das estruturas de dados
1 forall  $v \in V$  do
2    $C[v] \leftarrow 0$ ; // Identifica um vértice sem cor associada
3    $grau\_saturacao[v] \leftarrow 0$ ;
4    $grau[v] \leftarrow$  Número de vizinhos em  $G$ ;
5  $U \leftarrow V$ ; // Conjunto de vértices sem cor
6 while  $U$  não for vazio do
7   // Selecione um vértice não colorido  $u$  com grau de
    // saturação máximo
8    $u \leftarrow$  SelecionaVerticeSaturacaoMaxima( $U, grau\_saturacao, grau$ );
9   // Encontre a menor cor disponível (menor inteiro
    // positivo)
10   $cores\_usadas \leftarrow \{cores dos vizinhos coloridos de u\}$ ;
11   $cor \leftarrow 1$ ;
12  while  $cor \in cores\_usadas$  do
13     $cor \leftarrow cor + 1$ ;
14     $C[u] \leftarrow cor$ ; // Associe a cor e atualize
15     $U \leftarrow U \setminus \{u\}$ ; // Remova  $u$  do conjunto de não coloridos
    // Atualize os graus de saturação dos vizinhos não
    // coloridos
16  forall  $v \in Vizinhos de u em U$  do
17    if nova cor  $cor$  é diferente das cores dos vizinhos existentes then
       $grau\_saturacao[v] \leftarrow grau\_saturacao[v] + 1$ ;
17 return  $C$ ;

```

Algorithm 1: Algoritmo de Coloração de Grafos DSATUR

2. Planaridade

Pesquise sobre planaridade em grafos e realize as seguintes tarefas:

- (a) Defina grafos planares.
- (b) Descreva a fórmula de Euler e mostre graficamente porque ela funciona.
- (c) Descreva o Teorema de Kuratowski.

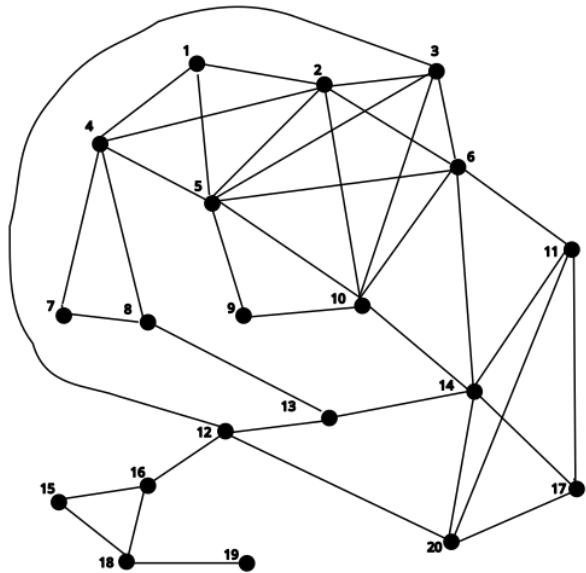


Figura 1: Grafo a ser colorido usando o algoritmo DSATUR.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	1	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
3	0	1	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0
4	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
5	1	1	1	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0
6	0	1	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0
7	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0
9	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
10	0	1	1	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0
11	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	1
12	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
13	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0
14	0	0	0	0	0	1	0	0	0	1	1	0	1	0	0	0	1	0	0	1
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0
17	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
20	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	1	0	0	0

Tabela 1: Matriz de adjacência do grafo da Figura 1