

# Analiza Algoritmilor

## Problema colorării nodurilor unui graf

Arina Emanuela Turcu

Universitatea Politehnica București  
Facultatea de Automatica și Calculatoare  
Grupa 323CA  
arina.turcu@stud.acs.upb.ro

15 November 2020

**Rezumat** Proiectul are ca scop analizarea unor algoritmi care rezolvă problema colorării nodurilor unui graf din punct de vedere al complexității lor. Se va propune câte o implementare în C++ pentru fiecare algoritm, urmând să fie testate, și se vor prezenta concluzii privind performanța și resursele folosite de fiecare.

**Cuvinte cheie:** Graf, Număr cromatic, Complexitate, Adiacență

## 1 Introducere

### 1.1 Descrierea problemei rezolvate

Problema colorării nodurilor unui graf este o problemă NP-completă care își propune să găsească numărul minim de culori necesare pentru a colora nodurile unui graf în așa fel încât oricare două noduri adiacente să nu aibă aceeași culoare. Numărul minim de culori de care este nevoie pentru a colora graful în acest fel este numit *număr cromatic* și este notat  $\chi(G)$ .

### 1.2 Exemple de aplicații practice pentru problema aleasă

Problema colorării nodurilor unui graf poate reprezenta programarea examenelor într-o facultate astfel: fiecare materie este reprezentată de un nod, iar o muchie între două noduri (materii) arată că există cel puțin un student care va da examen la ambele materii. Se încearcă găsirea unor intervale orare în care se vor da examenele astfel încât un student să nu aibă două examene în același timp.

Rezolvarea unui joc Sudoku poate fi asemănată cu problema colorării nodurilor unui graf în care nodurile reprezintă celulele. Există o muchie între două noduri dacă două celule sunt pe aceeași linie, același rând sau se află în același bloc. În acest caz, numărul cromatic este 9.

## 1.3 Specificarea soluțiilor alese

Dându-se un graf neorientat se poate găsi numărul cromatic prin mai multe metode, după cum urmează.

### 1.3.1 Breadth-first search

Se verifică dacă nodurile grafului pot fi colorate cu un număr de culori cuprins între 1 și  $V - 1$ . Pentru fiecare număr de culori, inițial se colorează toate nodurile cu prima culoare. Se realizează apoi BFS și pentru fiecare nod adiacent al nodului curent se verifică dacă are aceeași culoare cu nodul curent, caz în care i se asignează următoarea culoare. Dacă la un moment dat culorile se epuizează, verificarea se va termina cu valoarea *false*. Dacă se reușește să se coloreze toate nodurile corespunzător, verificarea se va termina cu valoarea *true*. Când găsește primul număr cu care se pot colora nodurile grafului, îl returnează.

### 1.3.2 Greedy algorithm

Acest algoritm va aproxima o soluție astfel: primului nod din graf i se va da o prima culoare, iar următoarelor noduri li se va da cea mai mică culoare nefolosită de niciunul dintre nodurile adiacente colorate anterior. Dacă toate culorile folosite anterior nu pot fi asignate, i se va da o culoare nouă. Când se ajunge la ultimul nod, algoritmul returnează numărul de culori folosite. Problema acestui algoritm este că pentru unele grafuri rezultatul va fi mult diferit de  $\chi(G)$ , de exemplu, în cazul unui graf bipartit care poate fi colorat cu doar 2 culori, algoritmul greedy va da un număr egal cu  $V/2$ , unde  $V$  este numărul de noduri ale grafului.

## 1.4 Specificarea criteriilor de evaluare alese pentru validarea soluțiilor

Pentru evaluarea algoritmului cu BFS va fi realizat un set de teste care verifică răspunsul dat de algoritm cu răspunsul exact. Pentru algoritmul greedy se vor da teste care verifică aproximarea rezultatului. Testele vor avea diverse grafuri neorientate cu una sau mai multe componente conexe și număr variabil de noduri și muchii. Vor fi măsurate timpul și resursele necesare fiecărui algoritm pentru a putea fi făcută o comparație între acestea.

## Bibliografie

- [1] GeeksForGeeks. <https://www.geeksforgeeks.org/graph-coloring-set-2-greedy-algorithm/?ref=lbp>.
- [2] GeeksForGeeks. <https://www.geeksforgeeks.org/m-coloring-problem-backtracking-5/?ref=lbp>.
- [3] GeeksForGeeks. <https://www.techiedelight.com/greedy-coloring-graph/>.