Universidade Federal de Uberlândia

Análise de Algoritmos

Relatório apresentado ao
Professor Paulo Henrique, da disciplina de
Análise de Algoritmos do
curso de Ciência da Computação.

Alunos: Brenner de Souza
Danilo Augusto Nunes
Danilo Vieira França

Matrículas: 11421BCC013
11611BCC021
11521BCC027

Uberlândia – Junho

Problema escolhido: Coloração de vértices.
O problema de coloração de grafos consiste em atribuir cores a certos elementos de um grafo sujeito a certas restrições. A **coloração de vértices** é o problema de coloração mais comum: dadas m cores, encontre uma maneira de colorir os vértices de um grafo de forma que dois vértices adjacentes não sejam coloridos com a mesma cor. Nesse contexto, o menor número de cores necessárias para colorir um grafo G é chamado de número cromático.

O problema para encontrar o número cromático de um dado grafo é NP-Difícil, de modo que se faz necessário o projeto de um algoritmo de aproximação para resolvê-lo. O Algoritmo 1 adota a estratégia gulosa para a atribuição de cores. Esse algoritmo não garante o uso do menor número de cores; em vez disso, ele provê um limite superior no número de cores. Nesse caso, ele nunca usa mais do que $d + 1$ cores, sendo que d é o grau máximo do grafo G.

Algoritmo 1: Algoritmo guloso básico de coloração de vértices.

Entrada: Um grafo G e uma lista de cores C
Saída: Uma coloração aproximada de G

para cada *vértice* $v_i \in V(G)$ **faça**
  atribua a v_i a **primeira** cor disponível em C que ainda não foi atribuída a nenhum de seus vizinhos já coloridos de v_i;
fim

No trabalho, como exemplo de execução do algoritmo, foram utilizados 5 grafos com diferentes dimensões, são eles: *petersen*, *chvatal*, *grotzsch*, *heawood* e *robertson*. Esses grafos são bastante conhecidos e estão disponíveis na biblioteca "igraph": [igraph Reference Manual]

2. Descrição dos experimentos realizados:

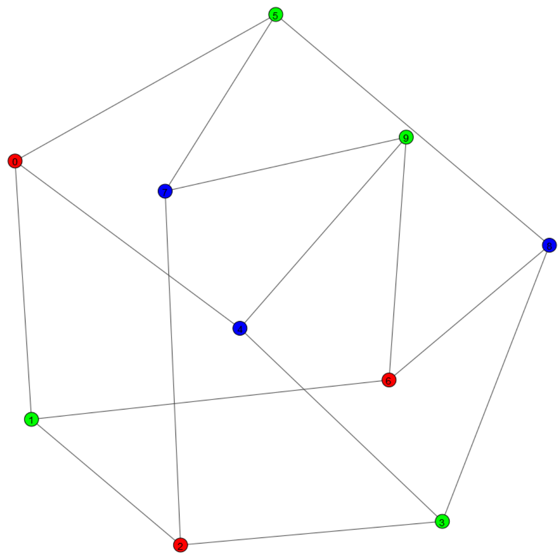
(a) Qual linguagem de programação foi adotada e por quê?
A linguagem adotada pelo grupo foi python, uma vez que ela é conhecida pela sua facilidade, já que permite o uso de estruturas de forma simples e ágil, além de possuir um conjunto grande de bibliotecas úteis. No trabalho, foi utilizada a biblioteca "igraph" para criar e manipular os grafos.

(b) Quais os grafos que foram gerados? Quais as dimensões de cada um? (Mostre as matrizes de adjacências desses grafos.)

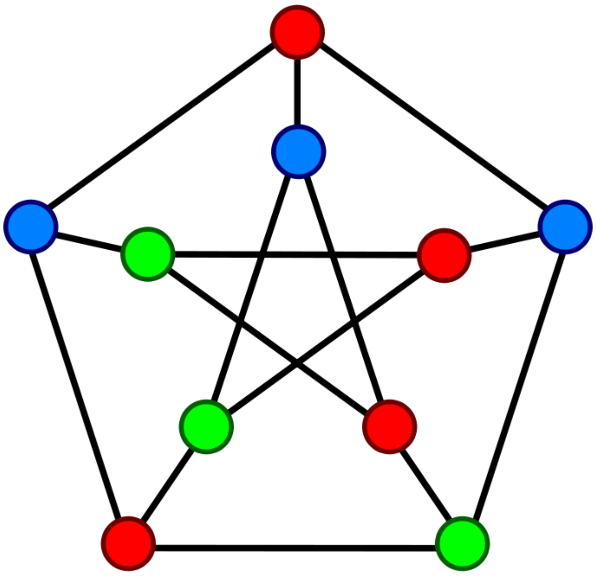
Petersen

Grafo 3-regular com 10 vértices e 15 arestas.
É o menor grafo não hamiltoniano que quando removido um único vértice qualquer, o grafo se torna hamiltoniano.

Solução do programa: 3 cores



Solução ótima: 3 cores



Matriz de adjacência:

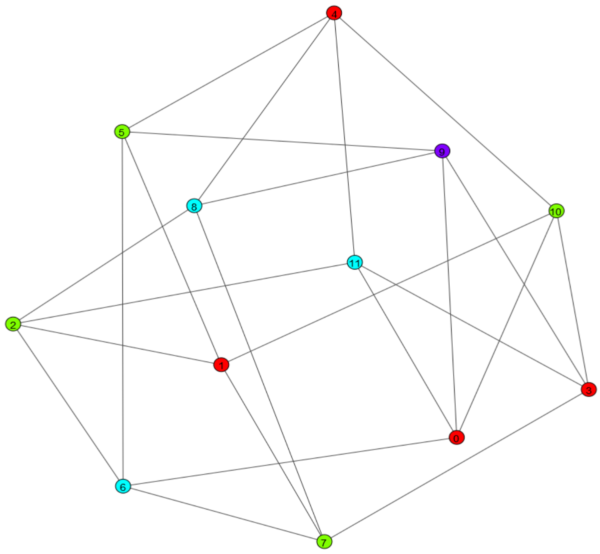
```
0 1 0 0 1 1 0 0 0 0
1 0 1 0 0 0 1 0 0 0
0 1 0 1 0 0 0 1 0 0
0 0 1 0 1 0 0 0 1 0
1 0 0 1 0 0 0 0 0 1
1 0 0 0 0 0 0 1 1 0
0 1 0 0 0 0 0 0 1 1
0 0 1 0 0 1 0 0 0 1
0 0 0 1 0 1 1 0 0 0
0 0 0 0 1 0 1 1 0 0
```

Link: Grafo de Petersen – Wikipédia, a enciclopédia livre (wikipedia.org)

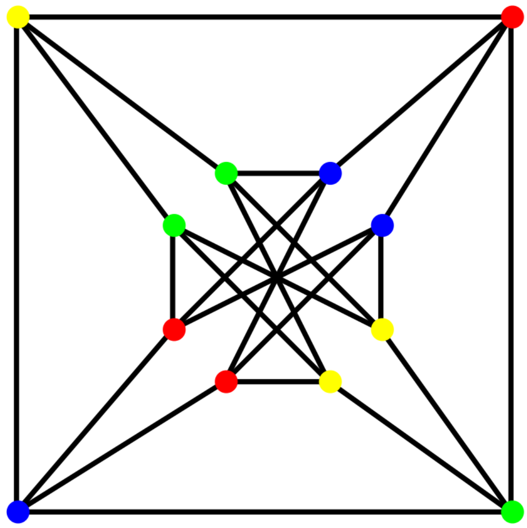
Chvatal

Grafo com 12 vértices e 24 arestas.
É o menor grafo livre de triângulos que é 4-cromático e 4-regular. De acordo com a conjectura de Grunbaum, existe um grafo m-regular e m-cromático com n vértices para cada m > 1 e n > 2. O grafo de Chvatal é um exemplo para m = 4 e n = 12.

Solução do programa: 4 cores



Solução ótima: 4 cores



Matriz de adjacência:

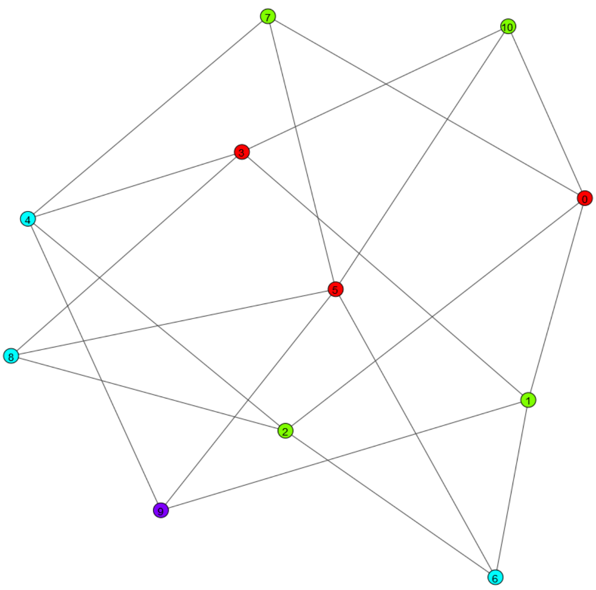
```
0 0 0 0 0 0 1 0 0 1 1 1
0 0 1 0 0 1 0 1 0 0 1 0
0 1 0 0 0 0 1 0 1 0 0 1
0 0 0 0 0 0 0 1 0 1 1 1
0 0 0 0 0 1 0 0 1 0 1 1
0 1 0 0 1 0 1 0 0 1 0 0
1 0 1 0 0 1 0 1 0 0 0 0
0 1 0 1 0 0 1 0 1 0 0 0
0 0 1 0 1 0 0 1 0 1 0 0
1 0 0 1 0 1 0 0 1 0 0 0
1 1 0 1 1 0 0 0 0 0 0 0
1 0 1 1 1 0 0 0 0 0 0 0
```

Link: [Grafo de Chvátal – Wikipédia, a enciclopédia livre (wikipedia.org)]

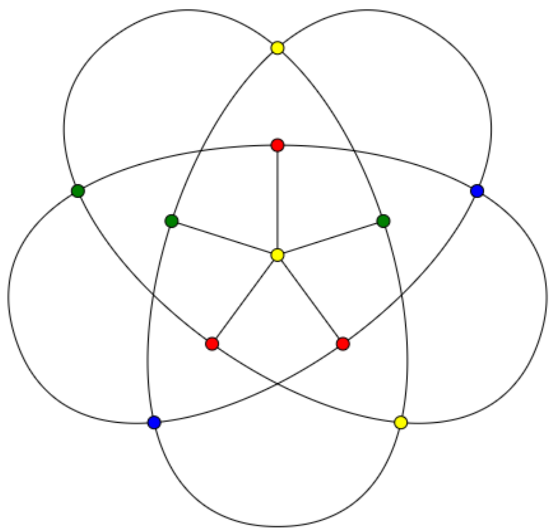
Grötzsch

Grafo com 11 vértices e 20 arestas.
É um grafo livre de triângulos, e sua existência demonstra que a suposição de planaridade é necessária no teorema de Grötzsch de que todo grafo planar livre de triângulo pode ser colorido com 3 cores.

Solução do programa: 4 cores



Solução ótima: 4 cores



Matriz de adjacência:

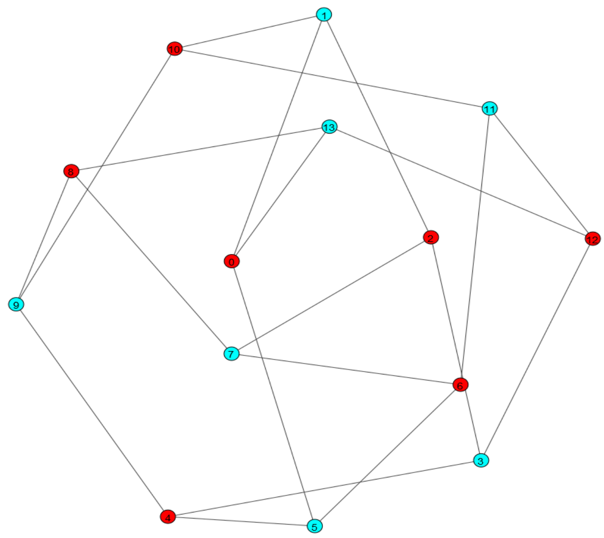
```
0 1 1 0 0 0 0 1 0 0 1
1 0 0 1 0 0 1 0 0 1 0
1 0 0 0 1 0 1 0 1 0 0
0 1 0 0 1 0 0 0 1 0 1
0 0 1 1 0 0 0 1 0 1 0
0 0 0 0 0 0 1 1 1 1 1
0 1 1 0 0 1 0 0 0 0 0
1 0 0 0 1 1 0 0 0 0 0
0 0 1 1 0 1 0 0 0 0 0
0 1 0 0 1 1 0 0 0 0 0
1 0 0 1 0 1 0 0 0 0 0
```

Link: [Grötzsch graph - Wikipedia]

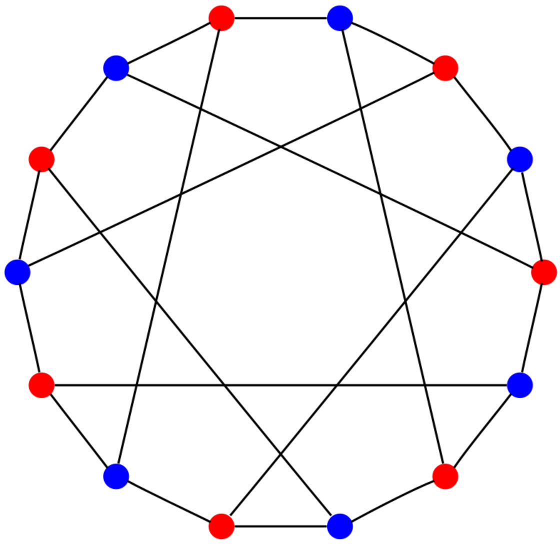
Heawood

Grafo com 14 vértices e 21 arestas.
É um grafo cúbico e todos os seus ciclos têm seis ou mais arestas.

Solução do programa: 2 cores



Solução ótima: 2 cores



Matriz de adjacência:

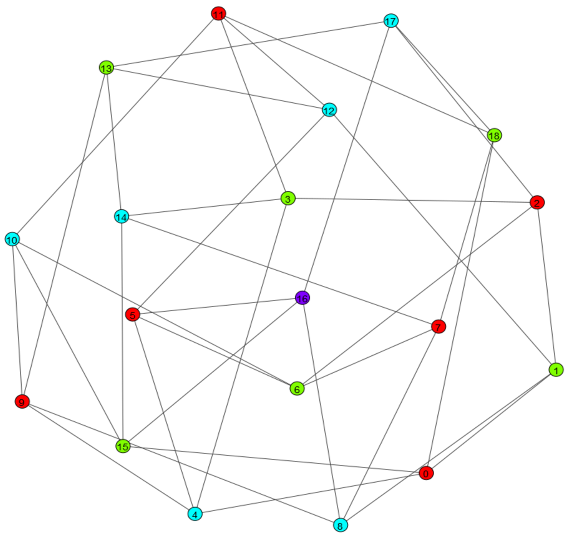
```
0 1 0 0 0 1 0 0 0 0 0 0 0 1
1 0 1 0 0 0 0 0 0 0 1 0 0 0
0 1 0 1 0 0 0 1 0 0 0 0 0 0
0 0 1 0 1 0 0 0 0 0 0 0 1 0
0 0 0 1 0 1 0 0 0 1 0 0 0 0
1 0 0 0 1 0 1 0 0 0 0 0 0 0
0 0 0 0 0 1 0 1 0 0 0 1 0 0
0 0 1 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 0 1 0 1 0 0 0 1
0 0 0 0 1 0 0 0 1 0 1 0 0 0
0 1 0 0 0 0 0 0 0 1 0 1 0 0
0 0 0 0 0 0 1 0 0 0 1 0 1 0
0 0 0 1 0 0 0 0 0 0 0 1 0 1
1 0 0 0 0 0 0 0 1 0 0 0 1 0
```

Link: [Grafo de Heawood – Wikipédia, a enciclopédia livre (wikipedia.org)]

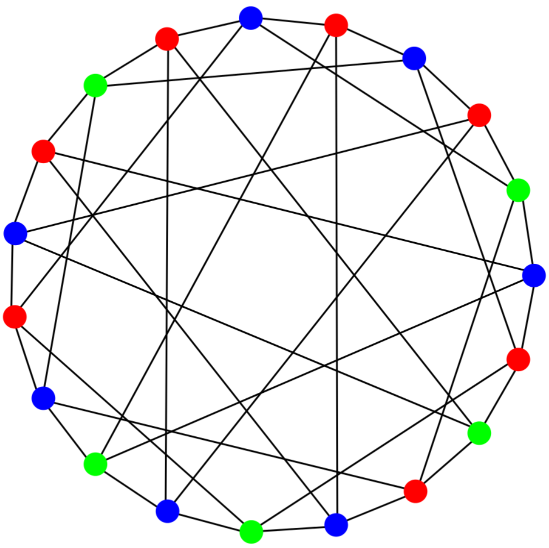
Robertson

Grafo com 19 vértices e 38 arestas.
É o único grafo 4-regular de cintura 5.

Solução do programa: 4 cores



Solução ótima: 3 cores



Matriz de adjacência:

```
0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1
1 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0
0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0
0 0 1 0 1 0 0 0 0 0 0 1 0 0 1 0 0 0 0
1 0 0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 1 0 0 0 0 0 1 0 0 0 1 0 0
0 0 1 0 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 0 0 1
0 1 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 0 0
0 0 0 0 1 0 0 0 1 0 1 0 0 1 0 0 0 0 0
0 0 0 0 0 0 1 0 0 1 0 1 0 0 0 1 0 0 0
0 0 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1
0 1 0 0 0 1 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 0 1 0
0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 1 0 0 0
1 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 1 0 0
0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 1 0 1 0
0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1
1 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0
```

Link: [Robertson graph - Wikipedia]

Para lidar com a confiabilidade do experimento, para cada grafo foi replicado o experimento 30 vezes. Em seguida, foi calculada a média e o desvio-padrão dessas 30 medições, os valores obtidos são apresentados abaixo:

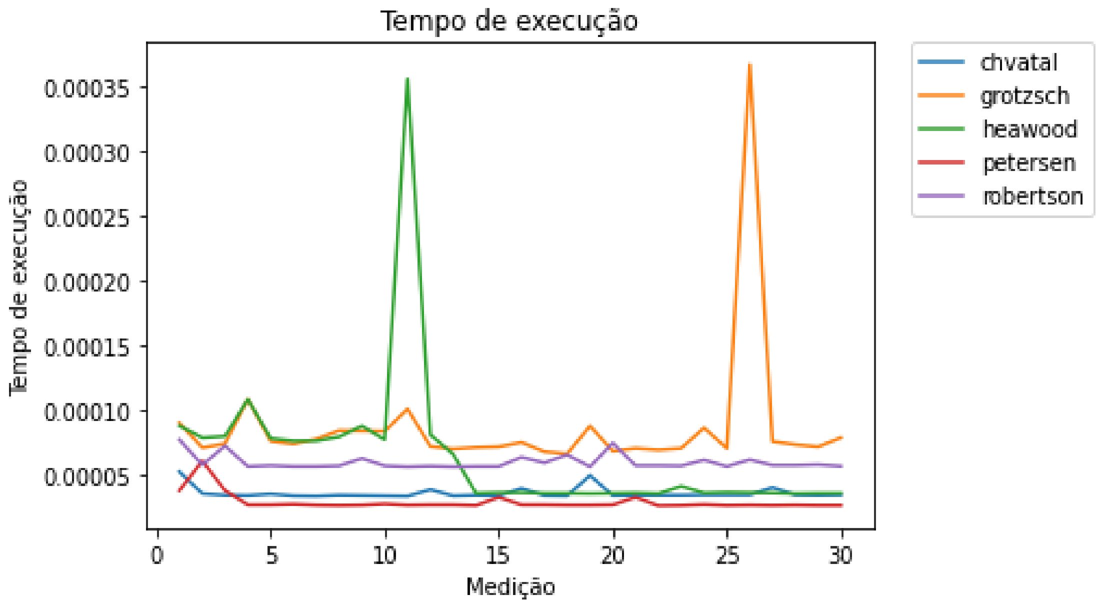


Tabela com as médias e desvios padrão dos grafos.

	Chvatal	Grotzsch	Heawood	Petersen	Robertson
Média	3.56753667195638e-05	8.671283721923828e-05	6.470680236816406e-05	2.880096435546875e-05	5.9541066487630205e-05
Desvio Padrão	4.368759791230956e-06	5.286362883304582e-05	5.859588217250054e-05	6.641972479749322e-06	5.574759118958888e-06

Diante disso, pode-se concluir que o algoritmo guloso de aproximação para o problema de coloração de vértices demonstrou um bom resultado, na maioria dos casos apresentou a solução ótima. Para este experimento, apenas o grafo de Robertson não obteve o menor número possível de cores para os vértices, mesmo assim, o programa desenvolvido obteve somente uma cor a mais da solução ótima. Além disso, com base na análise dos tempos de execução, o algoritmo se mostrou bastante rápido e uniforme, tendo alguns poucos picos no tempo.