Coloração de Grafos

Complexidade de Algoritmos
INF05515 - Turma A
Professora Mariana Kolberg

Tiago Biazus tiago.biazus@gmail.com
Rafael Vanz rafaelvanz@gmail.com

Introdução

Dado um grafo G, podemos colorir os vértices deste grafo com k cores de tal forma que as extremidades de cada aresta tenha uma cor diferente? Este e o problema de coloração de grafos.

Para k ≥ 3 temos um problema NP-Completo.

Nesta apresentação vamos mostrar que 3-Colorível é NP-Completo

Caracterização do problema

Uma k-coloração é uma função $f : V \to \{1, \ldots, k\}$ onde para cada aresta $\{u, v\}$ nós temos $f(u) \neq f(v)$. Se tal função existe para um dado grafo G, então G é k-colorível.

Para provar que 3-Colorível pertence a NP-Completo vamos primeiro provar que ele pertence à classe NP. Para fazer isso precisamos de uma coloração válida para um dado grafo, ou seja, um certificado. Também será necessário um algoritmo verificador de complexidade polinomial.

O segundo passo será provar que 3-Colorível pertence a classe NP-difícil. Para fazer isso faremos a redução de uma instância 3-SAT para 3-Colorível.

.3-Colorível pertence à classe NP

Certificado: para cada vértice uma cor de {1, 2, 3}

Verificador: checar se para cada aresta (u, v), a cor de u é diferente da cor de v

.3-Colorível pertence à classe NP

Algoritmo de verificação

Entrada:

- Um grafo G = (V, A) onde V é um conjunto de vértices e A um conjunto de pares de vértices
- S[1 ... n]: um certificado, uma sequência de cores para os vértices
- m e n, onde m representa o número de arestas e n o número de vértices

Saída:

- Uma resposta: SIM ou NÃO

.3-Colorível pertence à classe NP

Algoritmo de verificação

```
1. Para i:=1 até m faça
2.   se ( S[A[i].u.numero]   =  S[A[i].v.numero] )
3.       então retorne NÃO
4. Fim-Para
5. retorne SIM
```

Na linha 2 ocorre o teste para verificar se os vértices da aresta possuem corres diferentes no certificado. Se todos os pares de vértices das arestas tiverem cores diferentes, na linha 3 retorna SIM.

.3-Colorível pertence à classe NP

Algoritmo de verificação

```
1. Para i:=1 até m faça
2.    se ( S[A[i].u.numero]   ≠  S[A[i].v.numero] )
3.        então retorne SIM
4.        senão retorne NÃO
5.  Fim-Para
```

Cálculo da Complexidade

desemp[todas linhas] = desemp[1..5] = $\left(\sum_{i=1}^{m}(1)\right)$ = O(m)

.3-Colorível pertence à classe NP

Cálculo da Complexidade

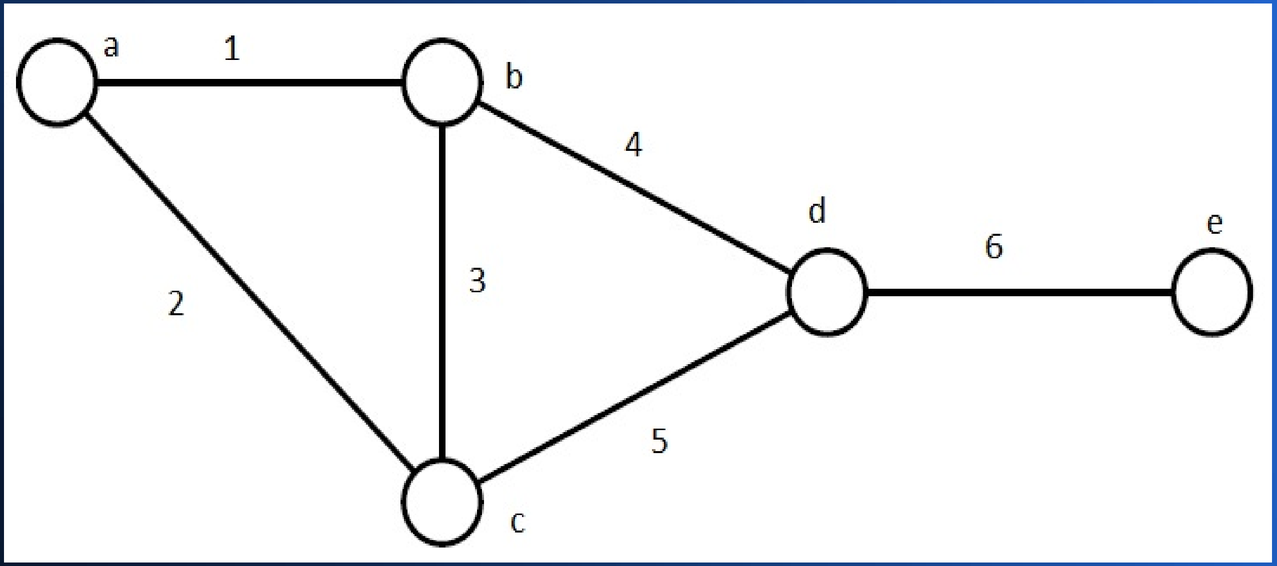
$$\text{desemp[todas linhas]} = \text{desemp[1..5]} = \left(\sum_{i=1}^{m}(1)\right) = O(m)$$

A complexidade O(m), linear, significa que o nosso algoritmo de verificação se encaixa dentro da definição de um problema NP, como queríamos demonstrar.

.3-Colorível pertence à classe NP

Verificação

Exemplo: Seja G = (V, A), onde V é representado por um array de vértices V[1..5] e A um array de arestas A[1..6]



.3-Colorível pertence à classe NP

Verificação

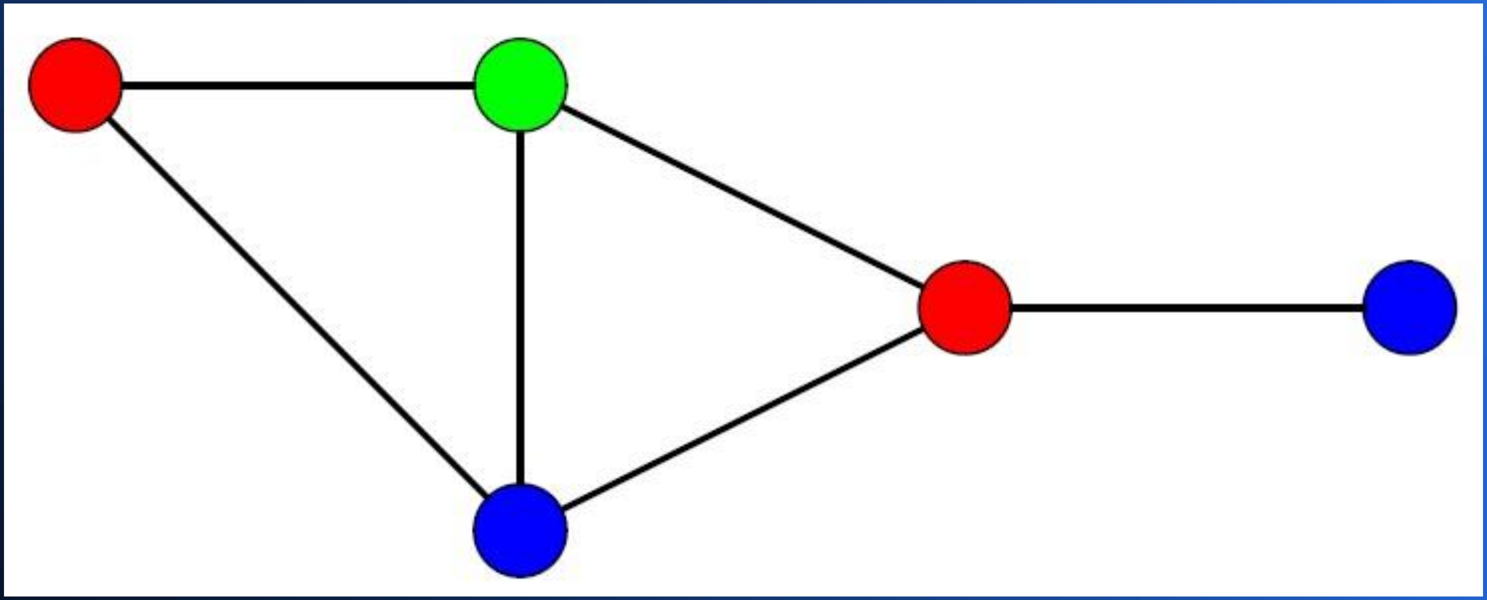
Pergunta: Dado um certificado S = {vermelho, verde, azul, vermelho, azul}, ou seja, o vértice V[1] terá cor vermelha, V[2] verde, V[3] azul, V[4] vermelha e V[5] azul, e o grafo G, ele é 3-colorível?

Resposta: Aplicando o algoritmo verificador para este grafo e este certificado, obtemos a resposta SIM.

.3-Colorível pertence à classe NP

Verificação

Grafo G 3-colorido com o certificado S = {vermelho, verde, azul, vermelho, azul}



.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

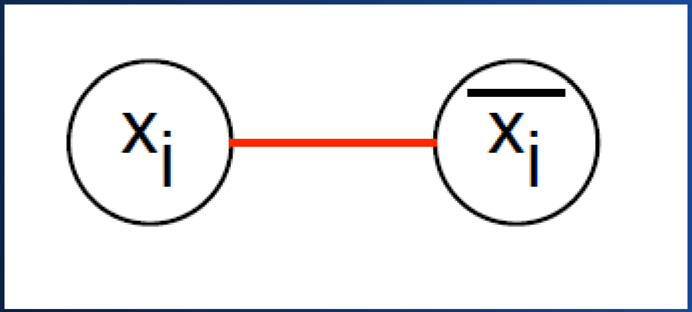
Para provar que 3-Colorível pertence a classe NP-difícil faremos a redução de uma instância 3-SAT para 3-Colorível.

Vamos construir um grafo G que será 3-Colorível se a instância 3-SAT é satisfazível.

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

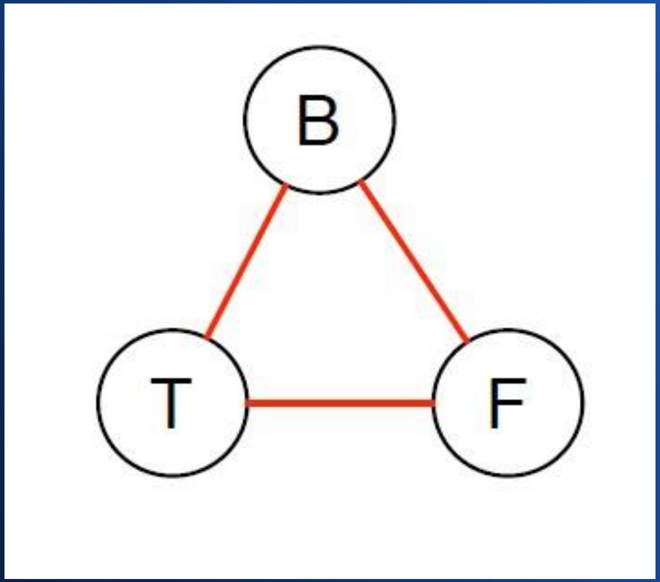
Seja x1, … , xn, (variáveis) C1, … , Ck (cláusulas) uma instância de 3-SAT.

Para cada variável xi são criados dois vértices em G, um para xi e um para ~xi. Conectar esses nós por uma aresta:



.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

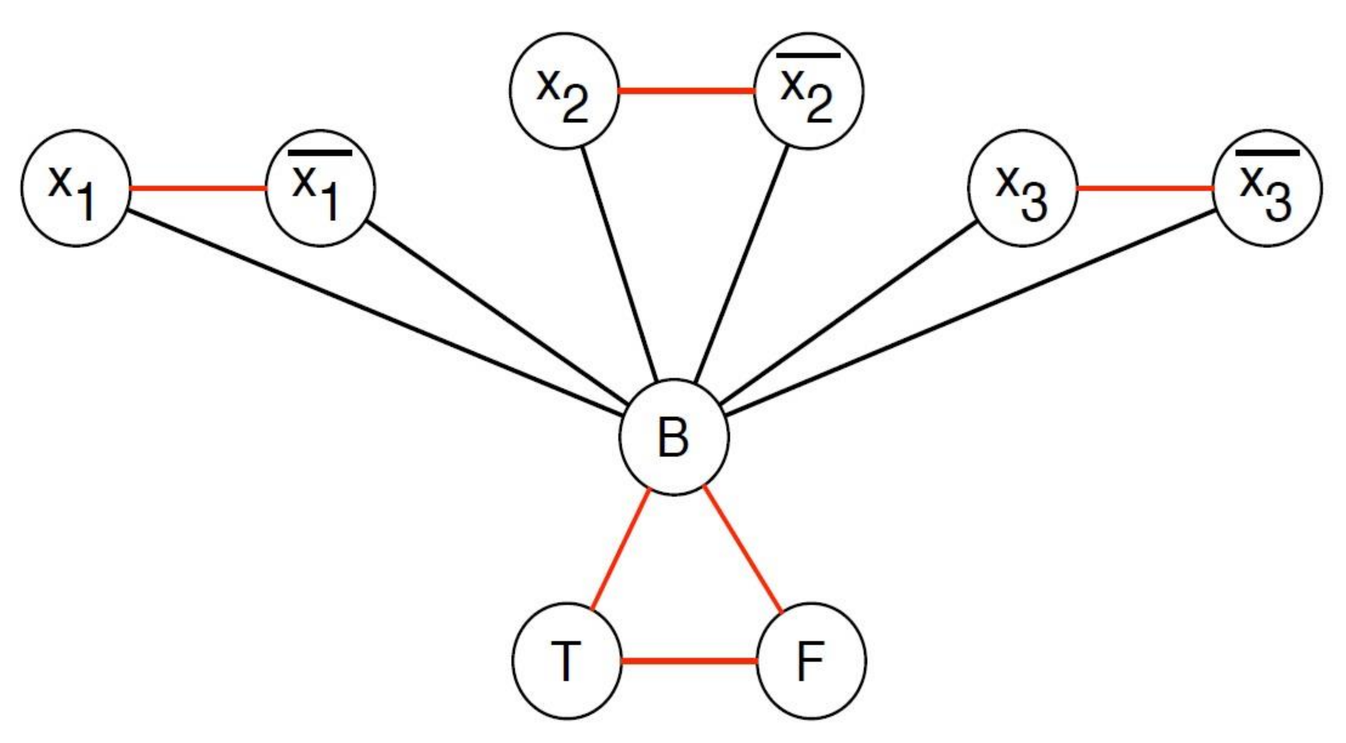
Criar três vértices especias T, F e B unidos por um triângulo:



Um grafo "gadget"

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Conectar cada variável vértice com o vértice B:



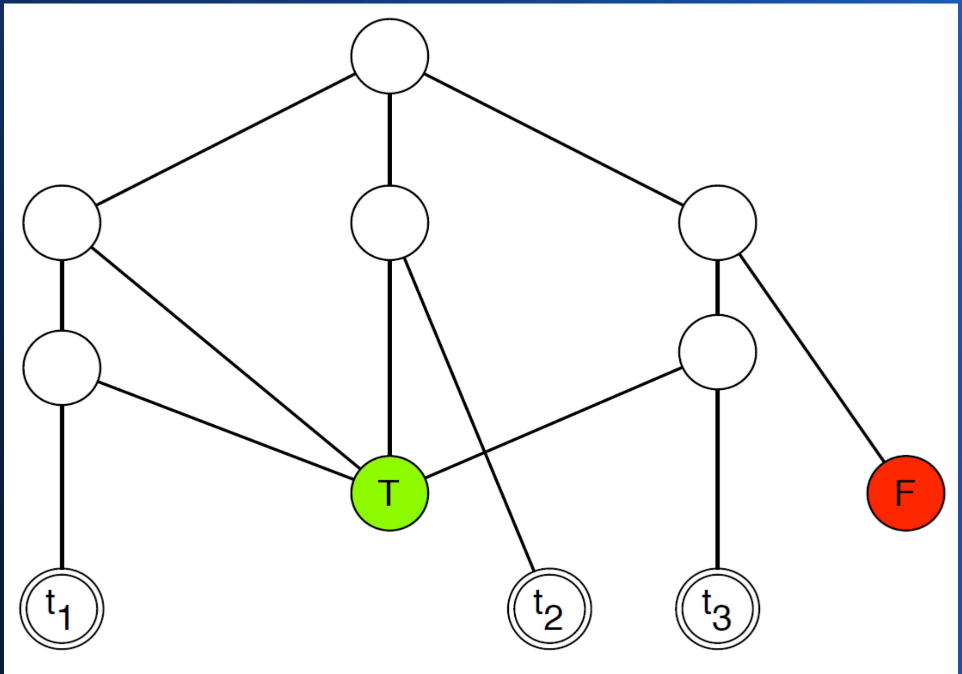
.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Propriedades:

- Cada xi e ~xi é colorido com cores diferentes
- Cada xi e ~xi deve ter cor diferente do vértice B
- B, T e F devem receber cores diferentes
- O vértice topo é colorível se um dos termos, de cada cláusula, receber a cor para “true”

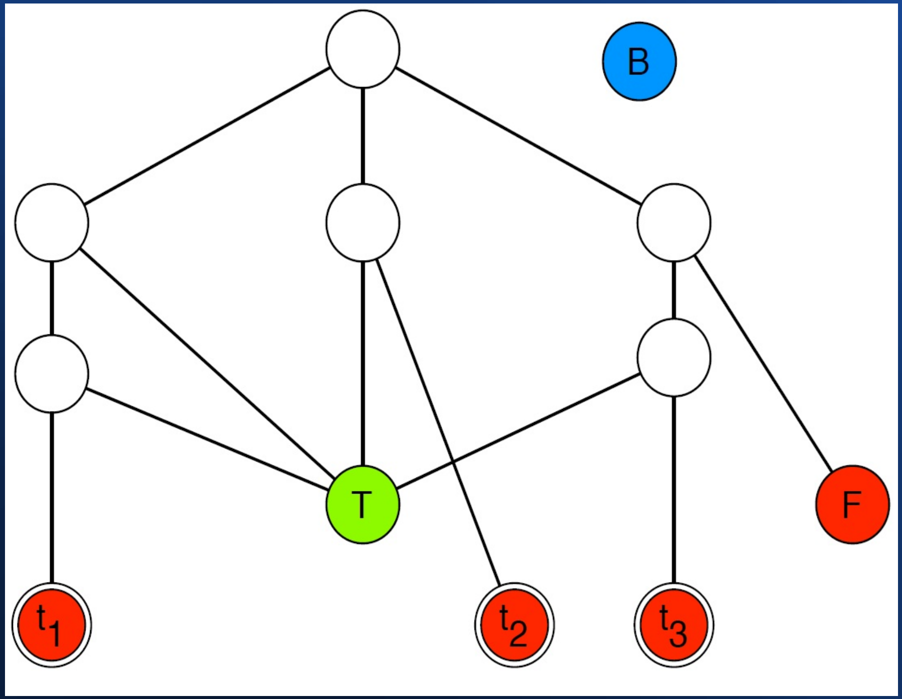
.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Seja C uma cláusula na fórmula. Pelo menos um dos seus termos deve ser “true”, porque se eles forem todos “false”, nós não conseguiremos colorir, como mostrado neste exemplo:

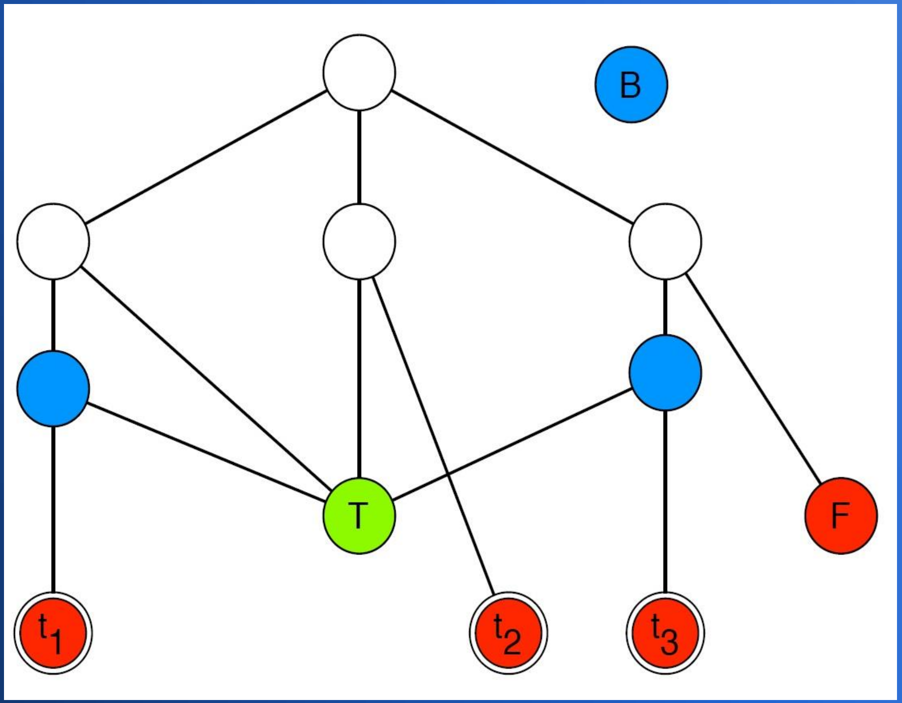


Passo 1

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

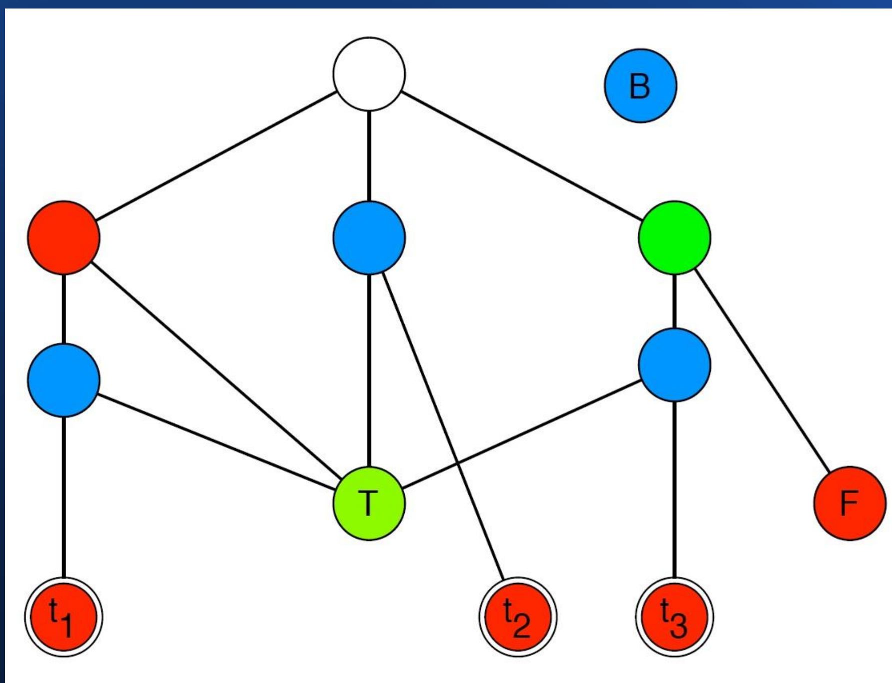


Passo 2

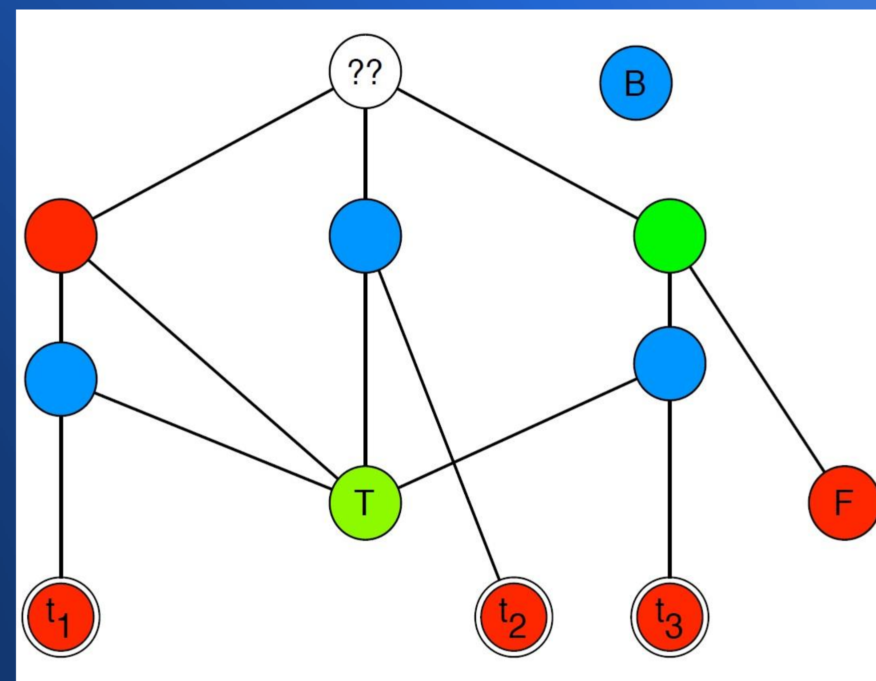


Passo 3

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)



Passo 4



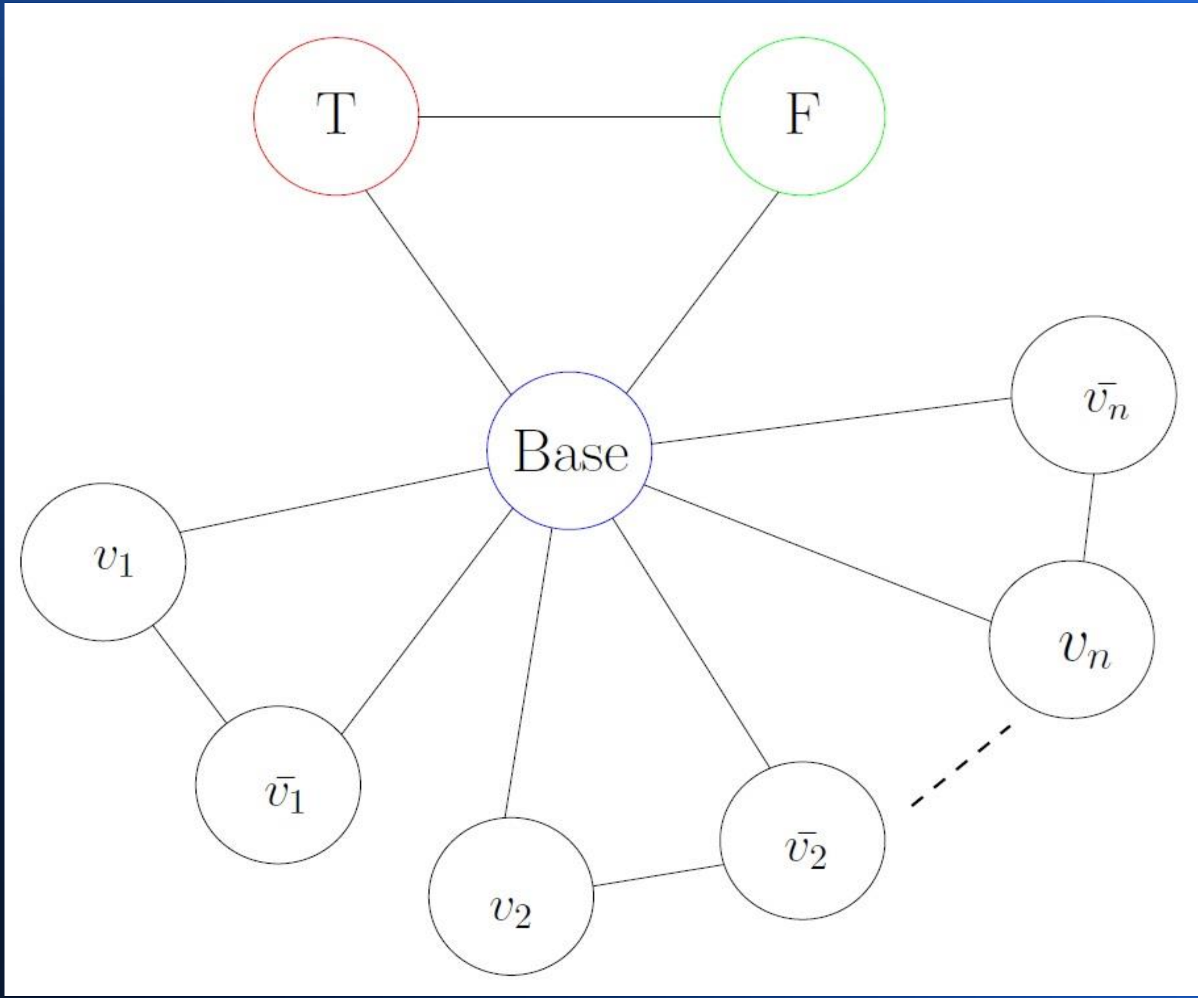
Passo 5

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Suponha que tenhamos uma fórmula satisfazível. Nós obteremos um grafo 3-Colorível de G fazendo:

- Colorindo os nós T, F e B, arbitrariamente, com três cores diferentes
- Se xi = true, colorir vi com a mesma cor que T e ~vi com a mesma cor que F
- Se xi = false, fazer o oposto
- Estender essa coloração para as cláusulas “gadgets”

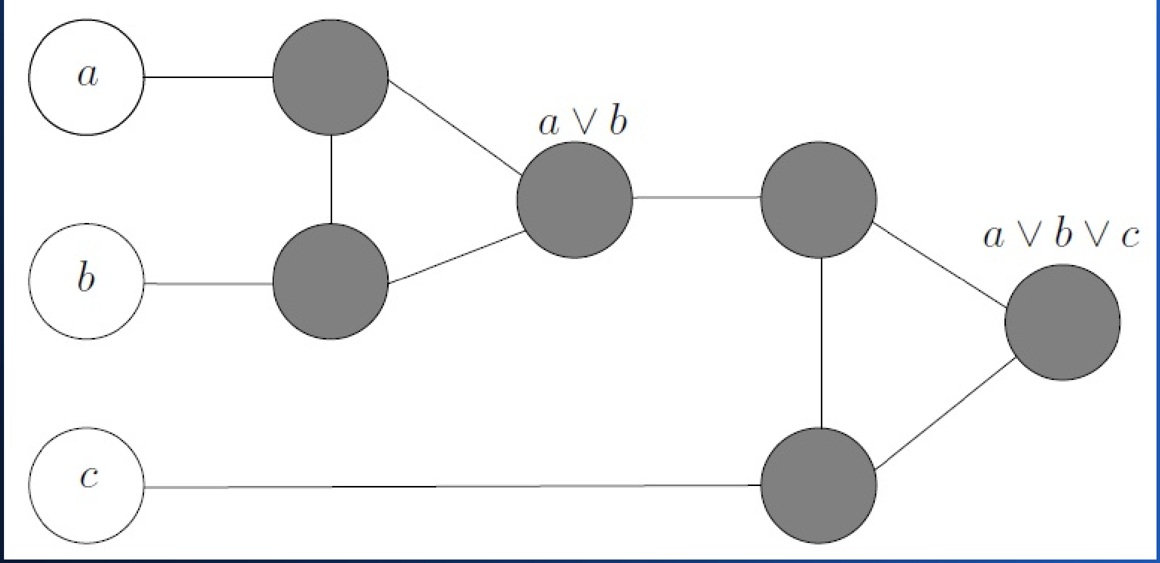
.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)



.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Para cada cláusula Cj = (a v b v c), criar um grafo gadget:

- o grafo gadget é conectado aos vértices correspondentes a a, b, c
- precisamos implementar o grafo OR-gadget



.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

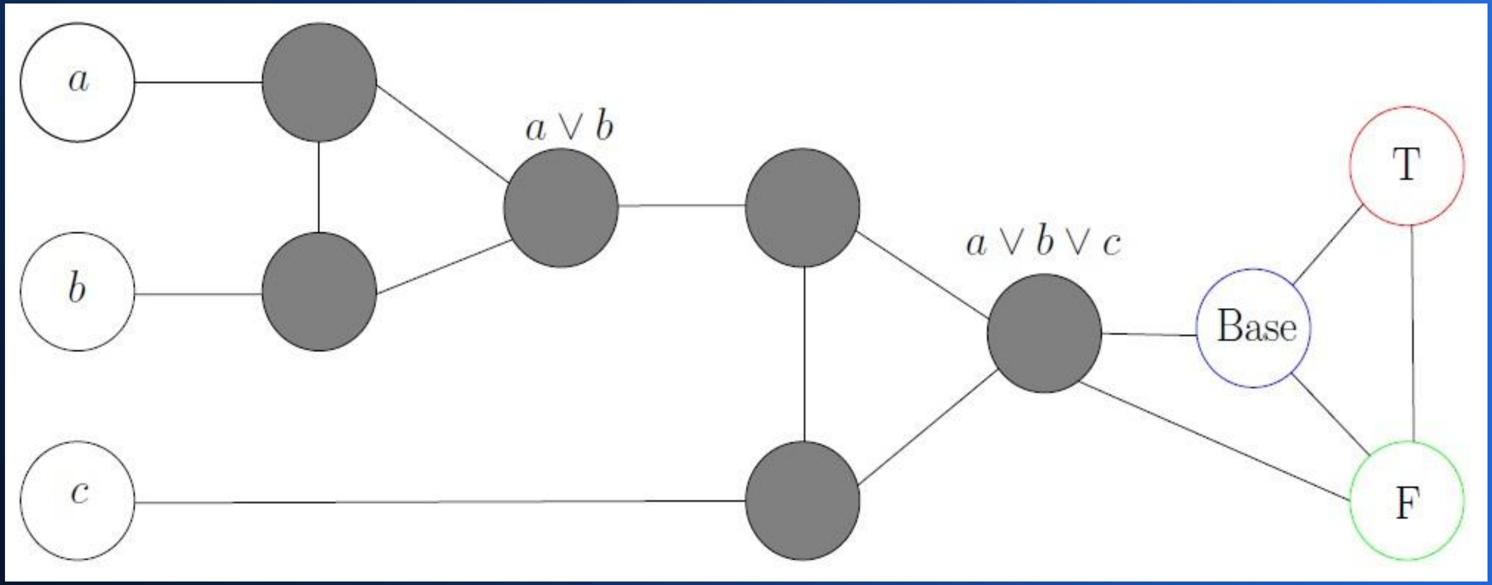
Propriedades:

- Se a, b, c são coloridos com a cor de False em um grafo 3-Colorível então o vértice saída do OR-gadget tem que ser colorido com a cor de False
- Se um dos vértices a, b ou c é colorido com a cor de True então o OR-gadget é 3-Colorível de tal forma que o vértice saída do OR-gadget é colorido com a cor de True
- Criar um triângulo com os vértices True, False e Base
- Para cada variável xi dois vértices vi e ~vi conectados em um triângulo com o vértice Base em comum

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Propriedades:

- Para cada cláusula Cj = (a v b v c), adicionar um grafo OR-gadget com vértices de entrada a, b, c e conectar o vértice saída do OR-gadget aos vértices False e Base



.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Corretez da Redução

Se a fórmula 3-SAT é satisfazível implica que o grafo G é 3-Colorível:

- Se xi recebe True, colorir o vértice vi com a cor de True e ~vi com a cor de False
- Para cada cláusula Cj = (a v b v c) ao menos um dos vértices a, b ou c é colorido com a cor de True. OR-gadget para Cj pode ser 3-Colorível de tal forma que a saída é True

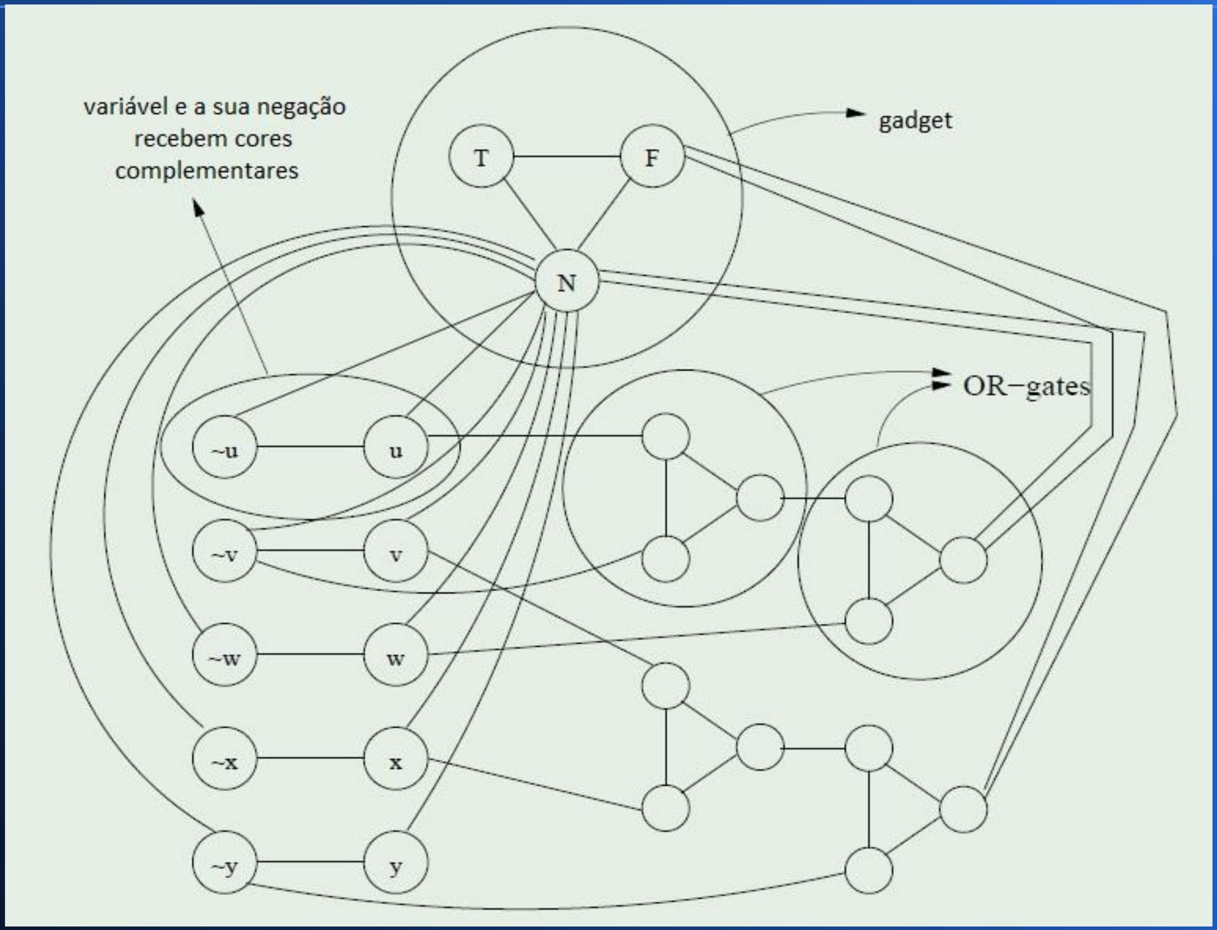
.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Corretez da Redução

Se o grafo G é 3-Colorível implica que a fórmula 3-SAT é satisfazível:

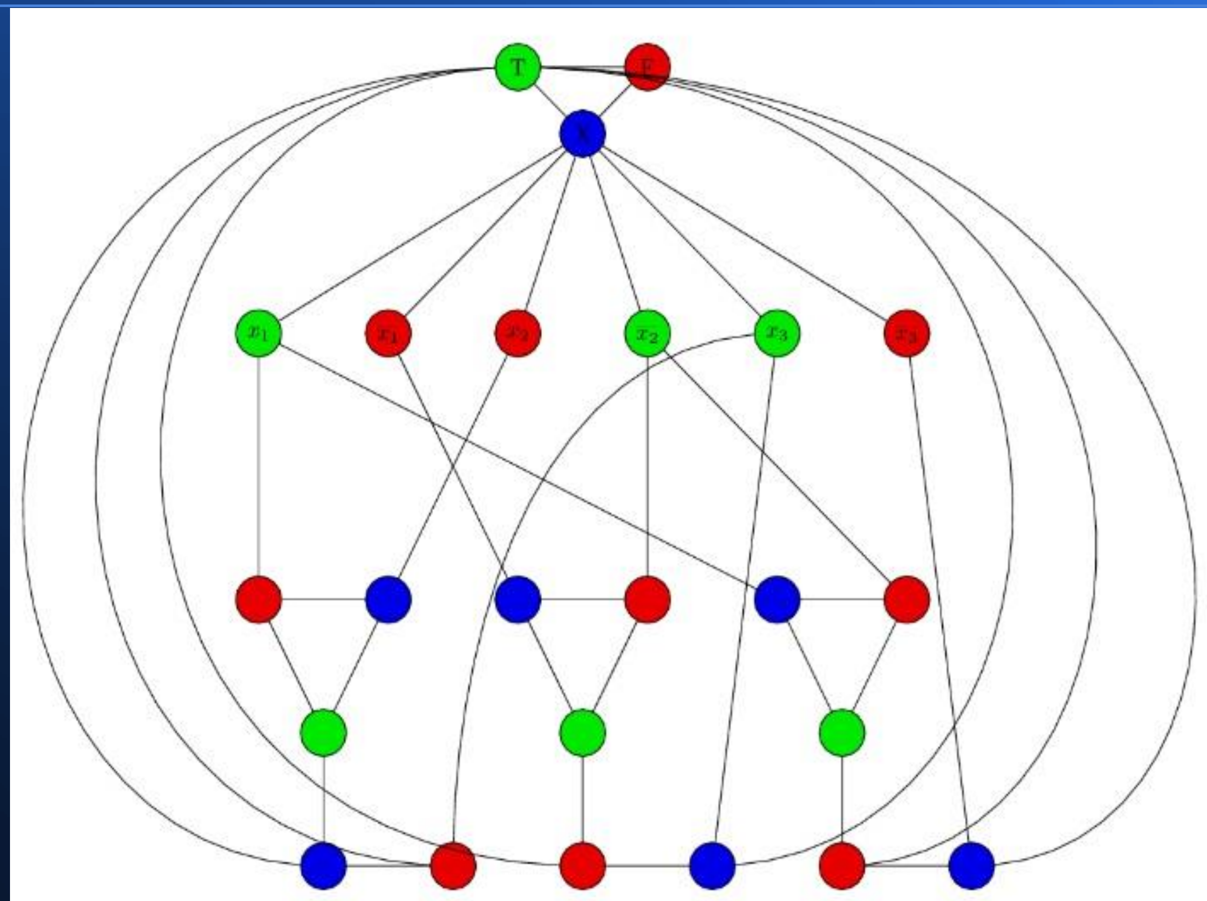
- Se vi é colorido com a cor de True então xi recebe True
- Considere qualquer cláusula Cj = (a v b v c). Não pode a, b e c serem False. Se forem False, a saída do OR-gadget para Cj tem que ser colorido com a cor de False, mas a saída é conectada ao nó Base e ao nó False, ou seja, isso não pode ocorrer!

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)



Grafo correspondente à fórmula $\Phi = (u \lor \neg v \lor w) \land (v \lor x \lor \neg y)$

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)



Grafo correspondente à fórmula $\Phi = (x_1 \vee x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3) \wedge (x_1 \vee \neg x_2 \vee \neg x_3)$

A fórmula é satisfazível. (x1 = ~x2 = x3 = True) e (~x1 = x2 = ~x3 = False)

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Algoritmo de redução

Entrada:

- Uma instância de 3-SAT
- n, onde n é o número de variáveis
- m, onde m é o número de cláusulas

Saída:

- Um grafo G

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

```
Criar gadget
Para i:=1 até n faça
    Criar variáveis (x_i e ~x_i)
    Conectar (x_i e ~x_i)
    Conectar à B (x_i e ~x_i)
Fim-Para
Para i:=1 até m faça
    Criar cláusulas (i)
Fim-Para
Retorne G
```

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Cálculo da complexidade

desemp[todas linhas] = desemp[1] + desemp[2..6] + desemp[7..9]

desemp[2..6] = $\left(\sum_{i=1}^{n}(1+1+1)\right)$ = O(3n) = O(n)

desemp[7..9] = $\left(\sum_{i=1}^{m}(1)\right)$ = O(m)

desemp[todas linhas] = O(n + m)

.3-Colorível pertence à classe NP-Difícil (3SAT $\leq_p$ 3COL)

Cálculo da complexidade

A complexidade O(n + m) caracteriza nosso algoritmo como uma solução de tempo polinomial.

.Conclusão

Mostramos que o problema 3-Colorível pertence à classe NP (possui algoritmo de verificação em tempo polinomial) e também demonstramos que ele pertence à classe NP-Difícil, através da redução do problema 3-SAT, que é NP-Completo, para 3-Colorível com um algoritmo em tempo polinomial. Esse problema atende as duas condições necessárias para ser NP-Completo (pertencer a NP e 3-SAT $\leq_p$ 3-COL), portanto o problema 3-Colorível é NP-Completo.