

TALLER PROGCOMP: TRACK GRAFOS

CICLOS

Gabriel Carmona Tabja

Universidad Técnica Federico Santa María,
Università di Pisa

September 8, 2024

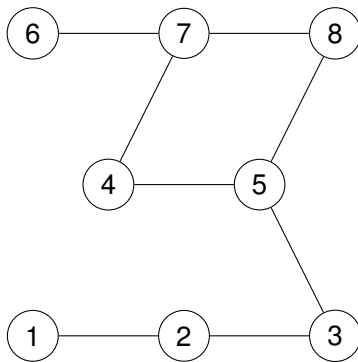
Part I

CICLOS

CICLOS EN GRAFOS NO DIRIGIDOS

Definición

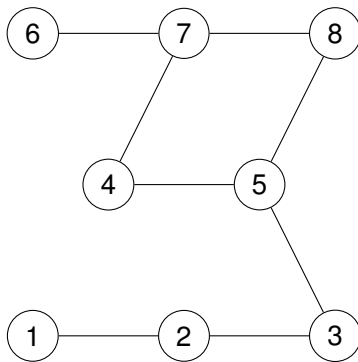
Un ciclo es un camino de al menos largo 4 que parte en un nodo y termina en el mismo nodo.



CICLOS EN GRAFOS NO DIRIGIDOS

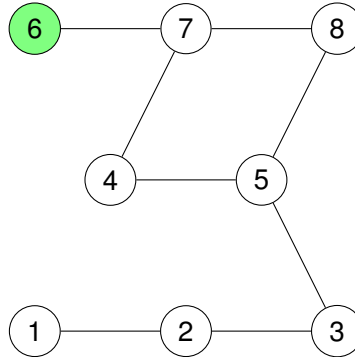
Definición

Un ciclo es un camino de al menos largo 4 que parte en un nodo y termina en el mismo nodo.

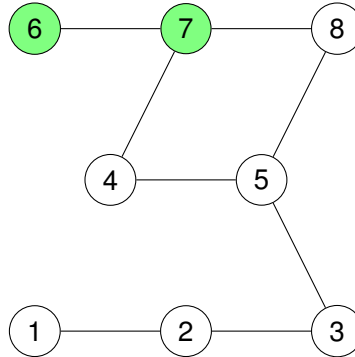


¿Cómo se puede verificar que un grafo tenga un ciclo?

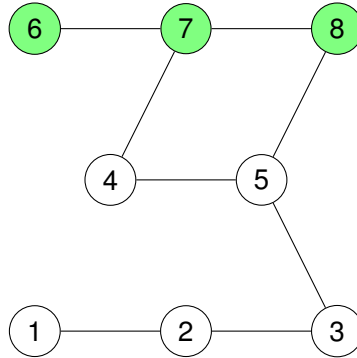
CICLOS EN GRAFOS NO DIRIGIDOS



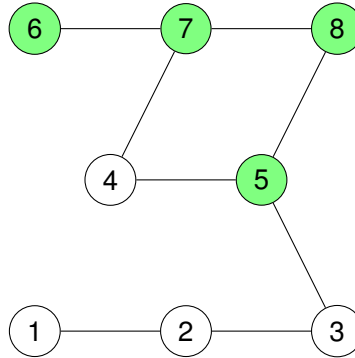
CICLOS EN GRAFOS NO DIRIGIDOS



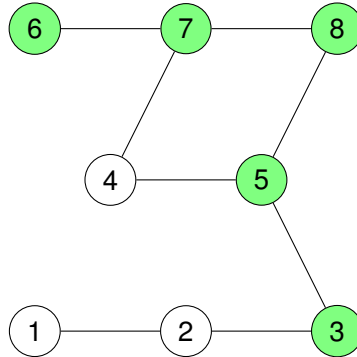
CICLOS EN GRAFOS NO DIRIGIDOS



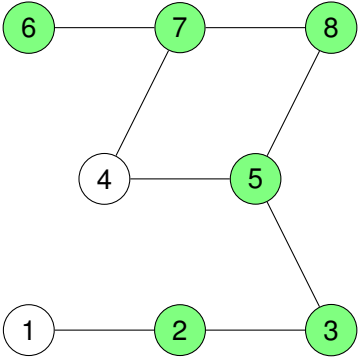
CICLOS EN GRAFOS NO DIRIGIDOS



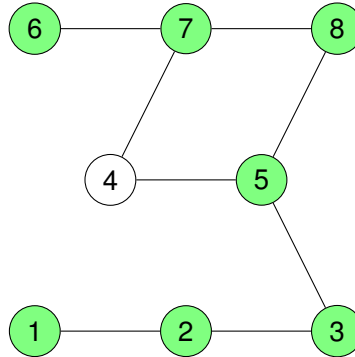
CICLOS EN GRAFOS NO DIRIGIDOS



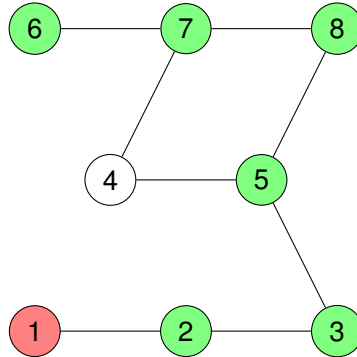
CICLOS EN GRAFOS NO DIRIGIDOS



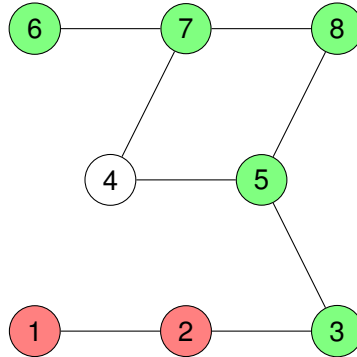
CICLOS EN GRAFOS NO DIRIGIDOS



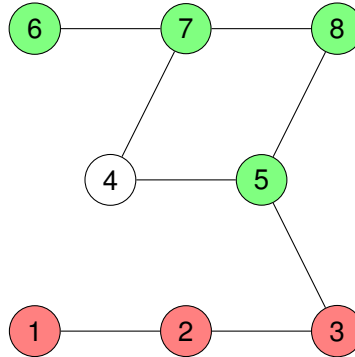
CICLOS EN GRAFOS NO DIRIGIDOS



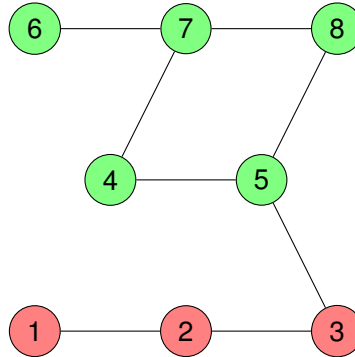
CICLOS EN GRAFOS NO DIRIGIDOS



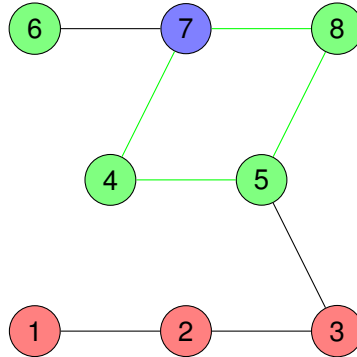
CICLOS EN GRAFOS NO DIRIGIDOS



CICLOS EN GRAFOS NO DIRIGIDOS



CICLOS EN GRAFOS NO DIRIGIDOS



CÓDIGO PARA VERIFICAR QUE EXISTA CICLOS

```
1  vector< vector< int > > lista
2  vector< int > vis;
3
4  #define NO_VISITADO 0
5  #define VISITADO_STACK 1
6  #define VISITADO 2
7
8  bool dfs(int u, int p) {
9      vis[u] = VISITADO_STACK;
10
11     bool check = false;
12     for(int i = 0; i < lista[u].size(); i++) {
13         int v = lista[u][i];
14         if(vis[v] == NO_VISITADO) {
15             if(dfs(v, u)) {
16                 check = true;
17                 return check;
18             }
19         } else if(vis[v] == VISITADO_STACK) {
20             if(v != p) {
21                 check = true;
22                 return check;
23             }
24         }
25     }
26     vis[u] = VISITADO;
27     return check;
28 }
```

OBTENER ALGÚN CICLO

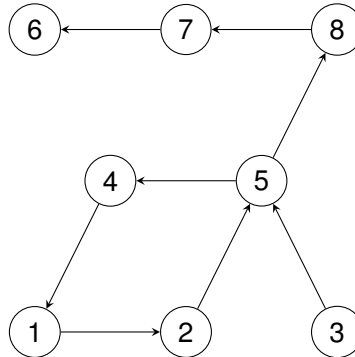
```
1  vector< vector< int > > lista
2  vector< int > vis;
3  vector< int > res;
4
5  bool dfs(int u, int p) {
6      vis[u] = VISITADO_STACK;
7
8      bool check = false;
9      for(int i = 0; i < lista[u].size(); i++) {
10         int v = lista[u][i];
11         if(vis[v] == NO_VISITADO) {
12             if(dfs(v, u)) {
13                 if(res[0] != res[res.size() - 1]) {
14                     res.push_back(u);
15                 }
16                 check = true;
17                 return check;
18             }
19         } else if(vis[v] == VISITADO_STACK) {
20             if(v != p) {
21                 check = true;
22                 res.push_back(v);
23                 res.push_back(u);
24                 return check;
25             }
26         }
27     }
28     vis[u] = VISITADO;
29     return check;
30 }
```

CICLOS EN GRAFOS DIRIGIDOS

¿Podremos encontrar si existe un ciclo en un grafo dirigido con el algoritmo anterior?

CICLOS EN GRAFOS DIRIGIDOS

¿Podremos encontrar si existe un ciclo en un grafo dirigido con el algoritmo anterior?
Si! :D. Pero tenemos que hacer una pequeña observación.



REFERENCES I