Поиск в глубину

$G = (V, E)$

```
Explore(v):
→ visited[v] = true
→ Previsit(v)
  for (v, u) ∈ E :
  → if not visited[u]:
        Explore(u)
→ Postvisit(v)
```

Списки связности:

$O(|V|+|E|)$

Матрица смежности:

$O(VE)$

```
DFS(G):
  for v ∈ V:
      visited[v] = false

→ for v ∈ V:
  → if not visited[v]:
        Explore(v)
```

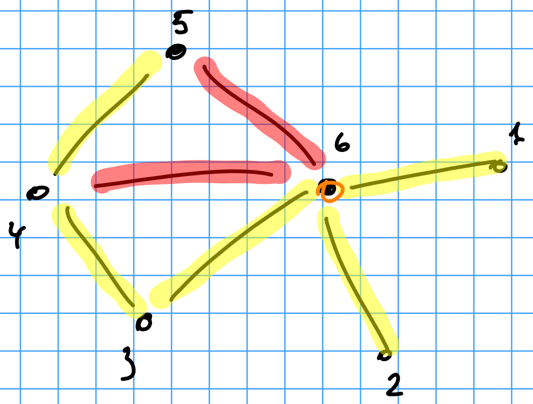
depth-first search

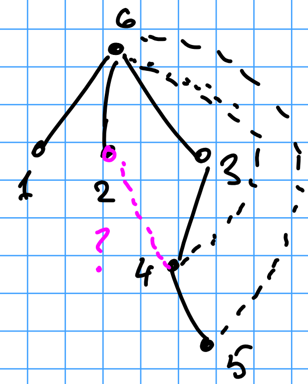
Поиск компонент связности в неор. графе

```
DFS(G):
  for v ∈ V:
      visited[v] = false
      cc[v] = 0
  component = 0
  for v ∈ V:
      if not visited[v]:
          component = component + 1
          Explore(v)
```

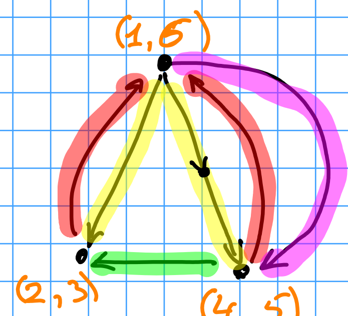
```
Previsit(v):
  cc[v] = component
```

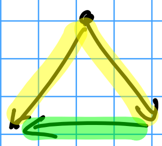
Поиск циклов в неор. графе





Поиск в глубину в ор. графе





В ор. графе есть 3 типа рёбер отн. DFS:

1. рёбра дерева
2. обратные рёбра
3. перекрёстные
4. прямое ребро

Previsit(v):

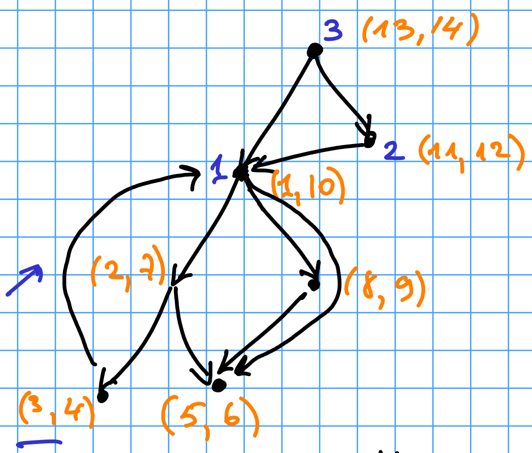
pre[v] = clock

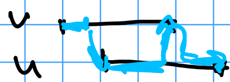
clock = clock + 1

Postvisit(v):

post[v] = clock

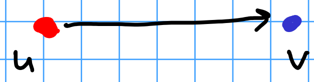
clock = clock + 1.

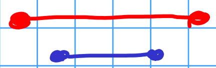


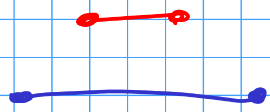


Утв: $\forall\, u, v \in V$:
$[pre[v], post[v]]$ и $[pre[u], post[u]]$
либо не пересекаются
либо один содержит другой

$\triangleright$ $pre[u] \in [pre[v], post[v]]$
$\Rightarrow post[u] \in \ldots$ $\triangleleft$



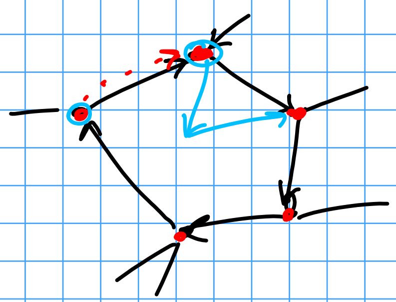
1.  древесное ребро / прямое ребро

2.  обратное ребро

3.  перекрестное ребро

Утв: Есть цикл => есть обратное ребро

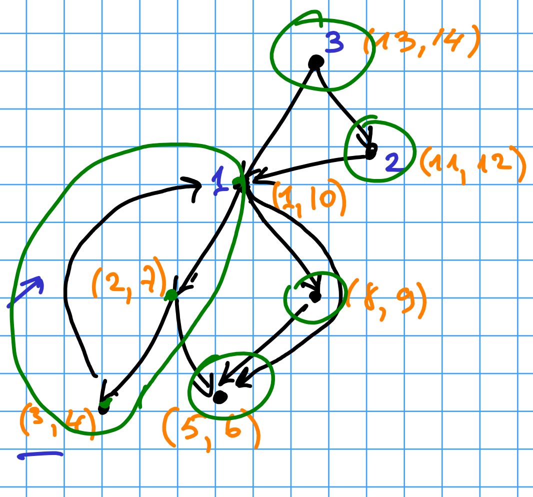
▷



◁

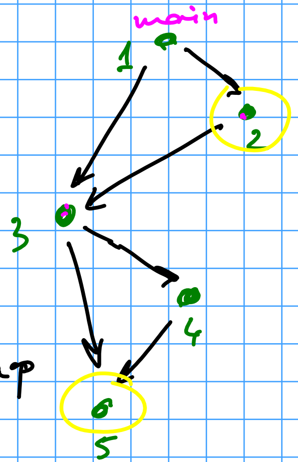
Тема компонента сильной связности

≡ u и v принадлежат одной компоненте сильной связности <=> $\begin{cases} u \rightsquigarrow v \\ v \rightsquigarrow u \end{cases}$



Утв: в метаграфе нет циклов.

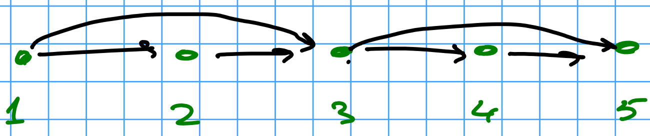
метаграф



DAG (directed acyclic graph)

Топологическая сортировка

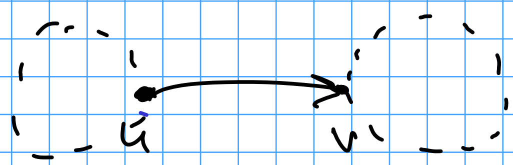
DAG можно 'линеаризовать'



Утв: $(u, v) \in E$ в DAG $\Rightarrow$

$$post[u] > post[v]$$

▷



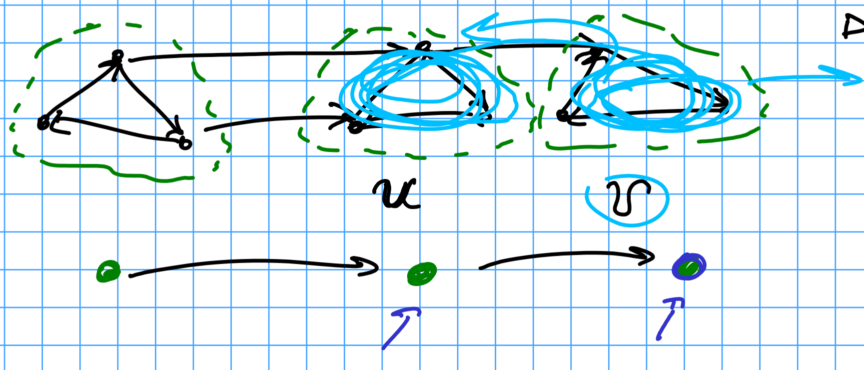
1. DFS первой посетил u $\Rightarrow$ $post[u] > post[v]$
   (т.к. $pre[u] < pre[v]$)

2. DFS сначала увидел v $\Rightarrow$
   $post[v] < pre[u] < post[u]$ ◁

$\Rightarrow$ топологическая сортировка $\Leftrightarrow$ сортировка по post по убыванию.

Утв: Если в метаграфе $(\mathcal{U}, \mathcal{V}) \in E \Rightarrow$

$$\max_{u \in \mathcal{U}} post[u] > \max_{v \in \mathcal{V}} post[v]$$



▷ 1. сначала DFS в $\mathcal{U}$

2. сначала DFS в $\mathcal{V}$

$post[v] < pre[u]$ $\forall v \in \mathcal{V}$ $\forall u \in \mathcal{U}$

Утв: вершина с max post лежит в конц. истоке.

SCC (G):

DFS (G^T)

→ for $v \in V$ (по убыванию post[v])

Explore(v) // на G

(см *)

$O(V+E)$