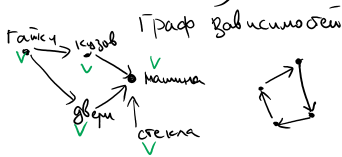


Задача Топологической сортировки. (Ориент. Графа)

Топ. сорт. - $P_1, \dots, P_n: e=(P_i, P_j) \in E$
 $i < j$



Лм. Топ. сорт. графа G существует \Leftrightarrow в G нет циклов.
 (DAG)

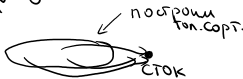
Ф-во \Leftrightarrow Топ. сорт. существует.

$P_{i_1}, P_{i_2}, P_{i_3}, \dots, P_{i_n}$ - цикл $\Rightarrow i_1 < i_2 < i_3 < \dots < i_n < i_1$ (!)

\Leftrightarrow \exists в G сток, т.е. вершина $v: \text{deg}_{out}(v) = 0$.

Ф-во $v \rightarrow v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_k$ Граф корректен \rightarrow найдем го стока.

сток - $\text{deg}_in v = 0$
 source



Алгоритм:

```
q = Queue(); ans = []
for v in range(n):
    if deg_in[v] == 0:
```



$O(n)$

$n = \# \text{вершин}$

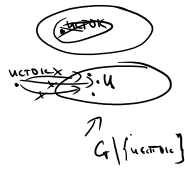
while not q.empty():

```
v = q.pop()
ans.append(v)
```

$\leftarrow v$ верш. обр. $= 1$ раз

```
for u in edges[v]:
    deg_in[u] -= 1
    if deg_in[u] == 0:
        q.push(u)
```

\leftarrow где v работает за $\text{deg}_{out}(v)$



$$T = O\left(V + \sum_{v=1}^V \text{deg}_{out}(v)\right) = O(V + E)$$

print(ans)

Формула алг. топологической сортировки.

```
ans = []
time = 0
def dfs(v):
```

```
    used[v] = 1; time_in[v] = time; time += 1
```

```
    for u in edges[v]:
```

```
        if used[u] == 0:
```

```
            dfs(u)
```

```
        elif used[u] == -1:
```

```
            print("Cycle!") # не топ. сорт.
```

```
    used[v] = 2
```

```
    ans.append(v)
```

```
    time_out[v] = time; time += 1
```

```
for v in range(V):
```

```
    if used[v] == 0:
```

```
        dfs(v)
```

```
print(reversed(ans))
```

\leftarrow по убыванию времени выхода

$$i < j \quad \text{ans}[i] \text{ ans}[j] \quad \text{timeOut}[\text{ans}[i]] < \text{timeOut}[\text{ans}[j]]$$

Формула корректности

```
 $v \rightarrow u$   $\Rightarrow$   $\text{timeOut}[v] < \text{timeOut}[u]$ 
```

Когда dfs был в вершине v и решил рекурсию $v \rightarrow u$:

- 1) и была корректна и чётко, $\text{timeOut}[u] < \text{timeOut}[v]$
- 2) и не была корректна не будем мы v , пока не закончится dfs(u) $\text{timeOut}[u] < \text{timeOut}[v]$