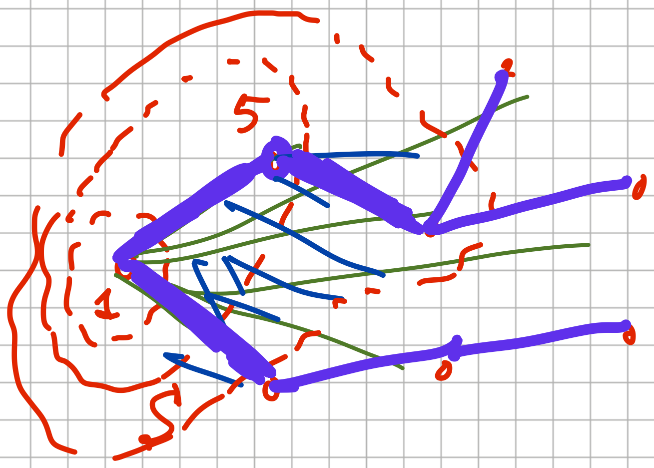
OR 1 29 - NOV - 2021

PRIM - DIJKSTRA alg.

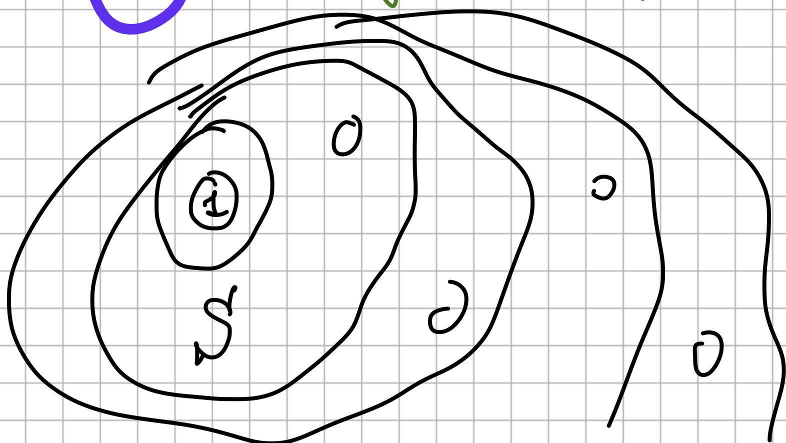


— T^*

$S = \{1, 5, \ldots\}$

$\delta(S)$

$e = \arg\min \{ c_f : f \in \delta(S) \}$



NESTED SETS S

$n := |V|$

PSEUDO - CODE

$T^* := \emptyset$; $S = \{1\}$;

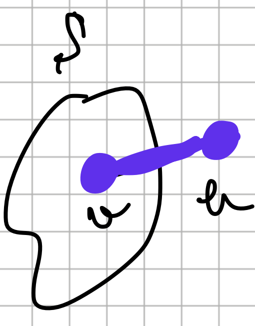
while $|T^*| \neq n-1$ do

$\boxed{[v, h] := \arg\min \{ c_{ij} : i \in S, j \notin S \}}$

$T^* := T^* \cup \{[v, h]\}$;

$S := S \cup \{h\}$

od



TIME - COMPLEXITY $O(|E| \cdot |V|)$

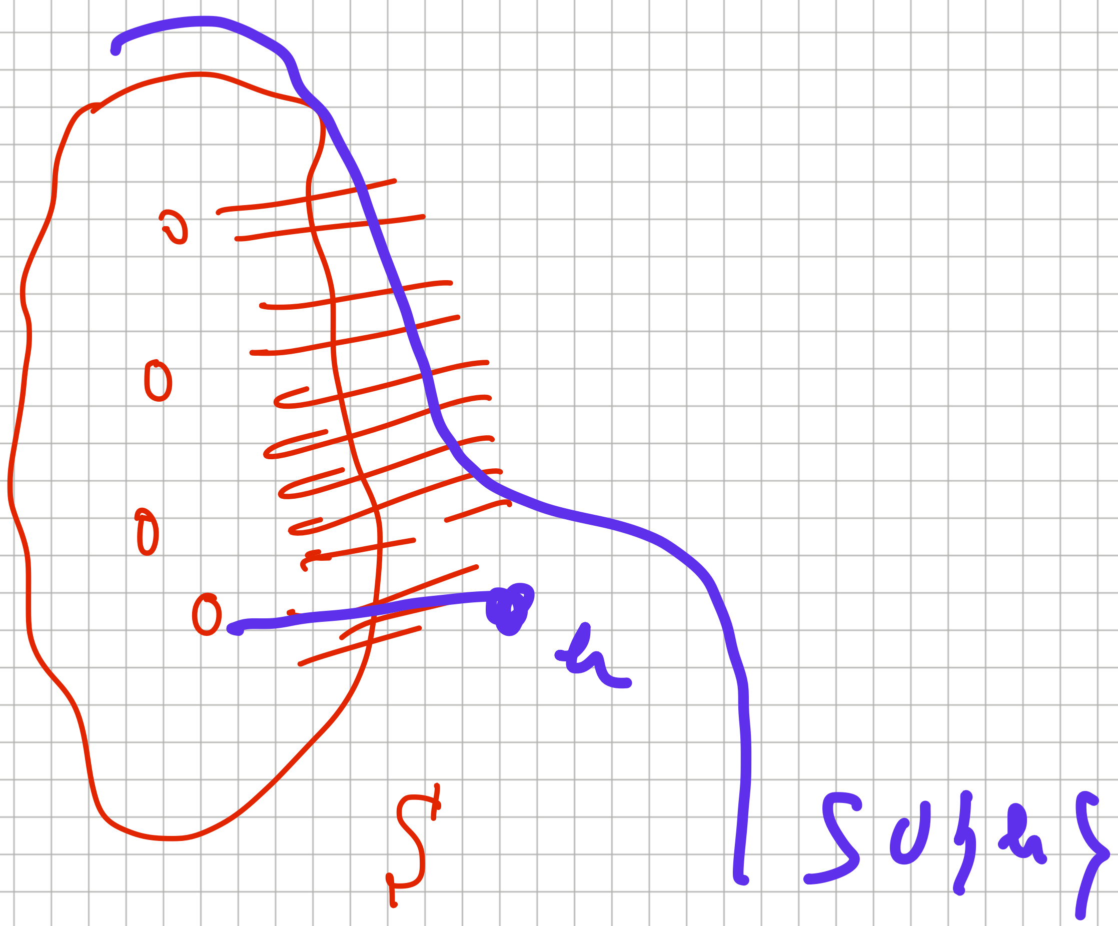
Complete graph $|E| \simeq |V|^2/2$

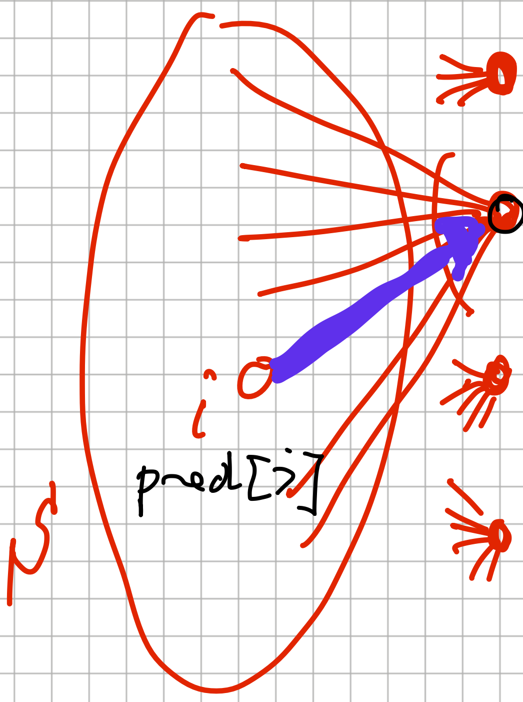
$\left(|E| = n \cdot (n-1)/2 \right)$

complexity $\simeq O(n^3)$, $n = |V|$

$\Rightarrow O(n^2)$?

YES ! PRIM / DIJKSTRA





$L[j] := \min\{ c_{ij} : i \in S \}$

pred[j]

$j \notin S$

" $\underbrace{\min\{ L[j] : j \notin S \}}_{O(n) \text{ time}}$ "

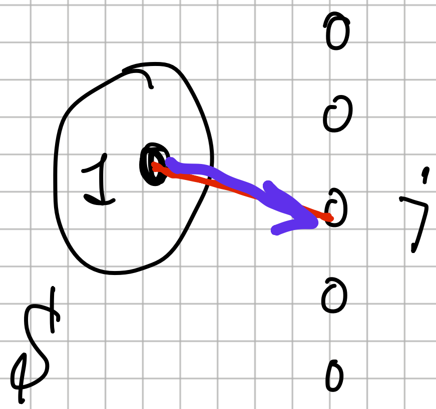
$O(n^2) \to O(n)$

INITIALIZATION:

$O(n)$

$C = \boxed{c_{ij}}$

SYMMETRIC MATRIX

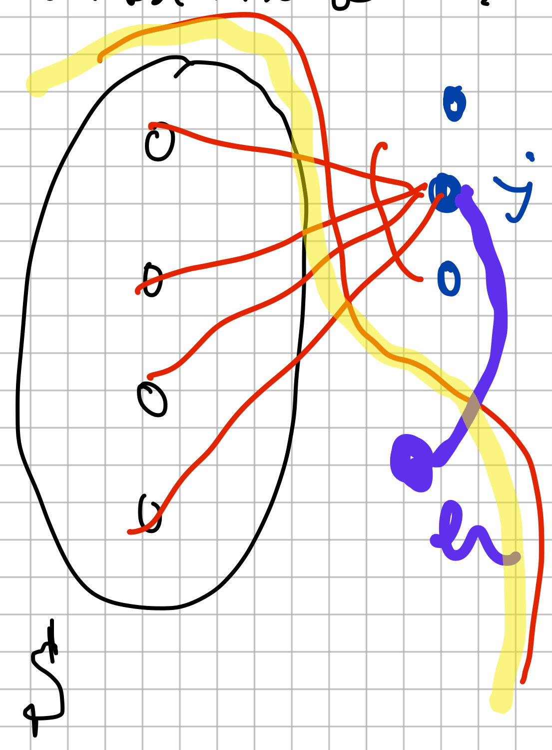


- $L[j] := c[1, j]$;
- $pred[j] := 1$

$\forall j = 2, \dots, n$

$L[1] := 0$; $pred[1] := 1$;

UPDATING:



$L[j] := \min\{ L[j], c_{hj} \}$

$S' = S \cup \{h\}$

$O(n)$ time in total

$T^* := \emptyset$; $S = \{1\}$; ← $O(n)$

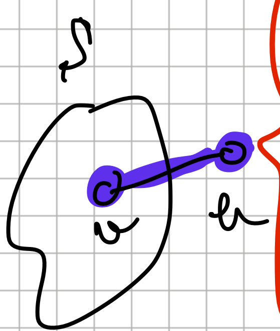
while $|T^*| \neq n-1$ do

  $[v,h] := \arg\min \{ c_{ij} : i \in S, j \notin S \}$ $O(n)$

  $T^* := T^* \cup \{[v,h]\}$;

  $S := S \cup \{h\}$ $O(n)$

od



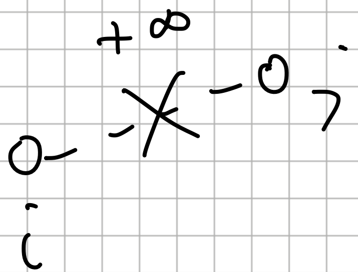
$n-1$ times

OVERALL TIME COMPLEXITY:

$O(n^2)$ time

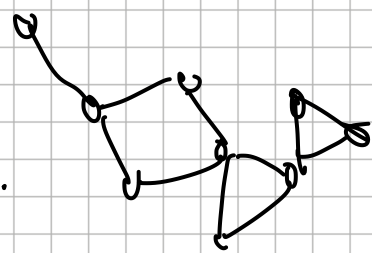
WHAT ABOUT SPARSE GRAPHS ?

- $|E| \approx n^2/2$



$$C = \begin{array}{cc} \begin{bmatrix} & \vdots & \\ \cdots & \boxed{+\infty} & \cdots \\ & \vdots & \end{bmatrix} & i \\ j & \end{array}$$

- $|E| \leq 3|V|$ planar gr.



$\approx k|V|$ for a small (fixed) k

$\Rightarrow O(n^2)$ can be improved !

"$\boxed{O(|E| \log n)}$"

complete

$O(n^2 \log n)$

worse than P-D $O(n^2)$

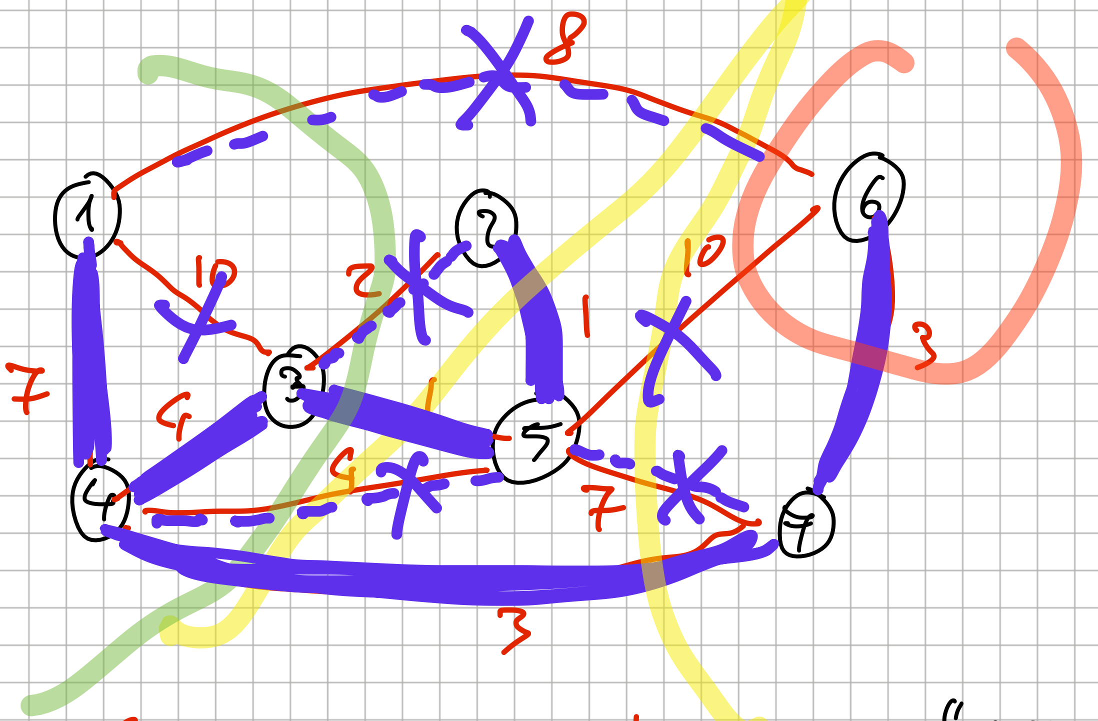
sparse graphs

$|E| = kn$

$O(n \log n)$

much better than P-D $O(n^2)$

KRUSKAL alg.



GREEDY POLICY

PSEUDO-CODE

1. SORT the edges by increasing costs: "non decreasing"

$$c_{e_1} \le c_{e_2} \le \dots \le c_{e_m} \qquad O(m \log m)$$

where $m := |E|$

2. $T^* := \emptyset$;

3. **for** $k = 1$ **to** m **do**

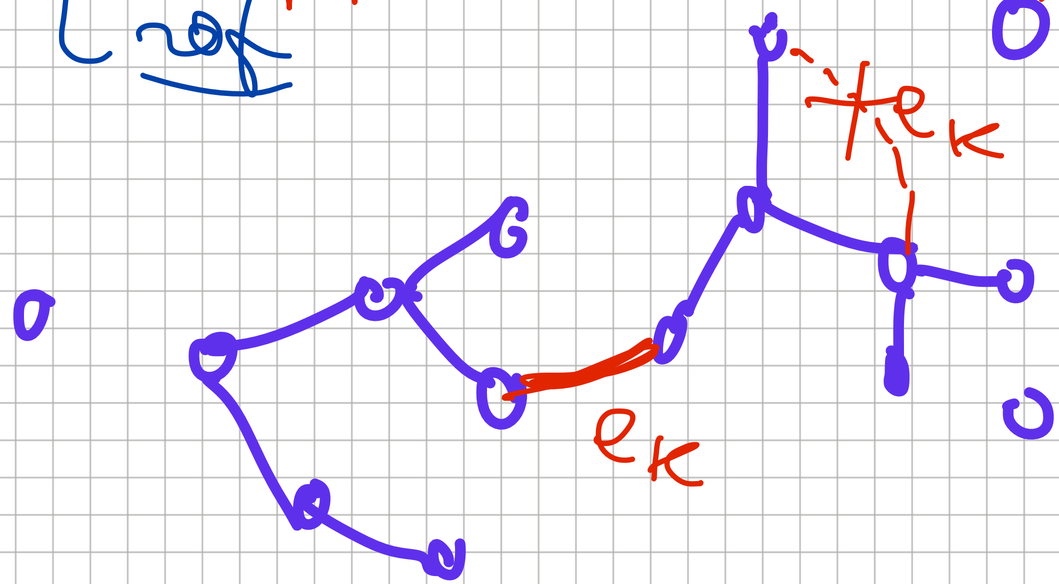
   if $T^* \cup \{e_k\}$ does not create a cycle

   then $T^* := T^* \cup \{e_k\}$

fi

if $|T^*| = n-1$ then BREAK ;

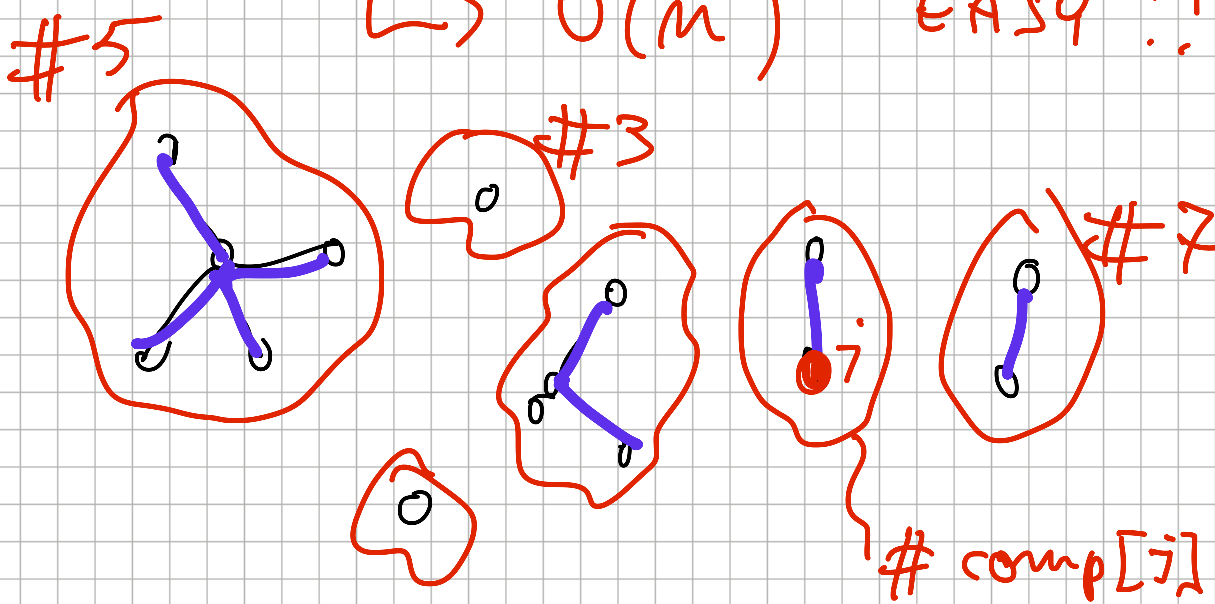
od



n times

↳ $O(\log n)$?? YES!

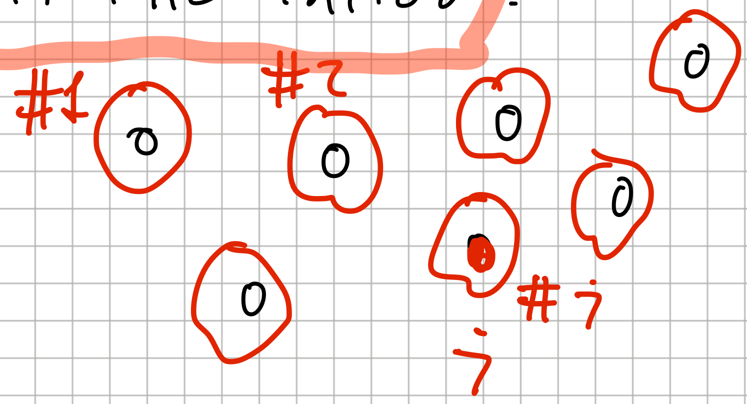
↳ $O(n)$ EASY !!



comp[j] = "name" of the component containing node j

INITIALIZATION:

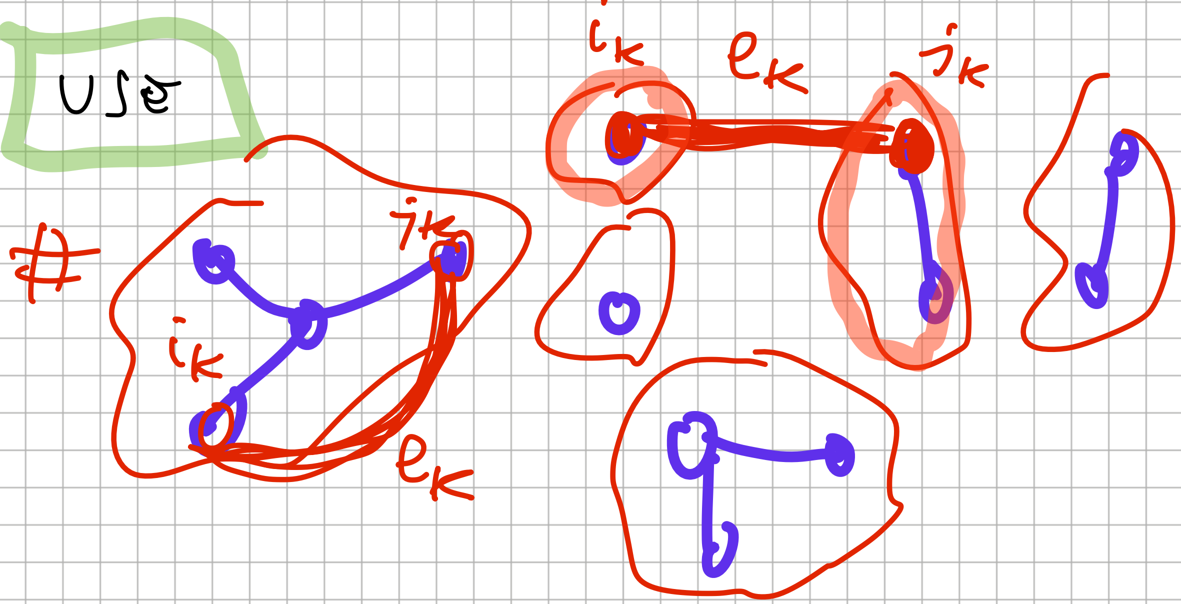
$T^* = \emptyset$



$O(n)$ time

$comp[i] := i, \quad \forall i = 1, \dots, n$

USE



$e_k = [i_k, j_k]$

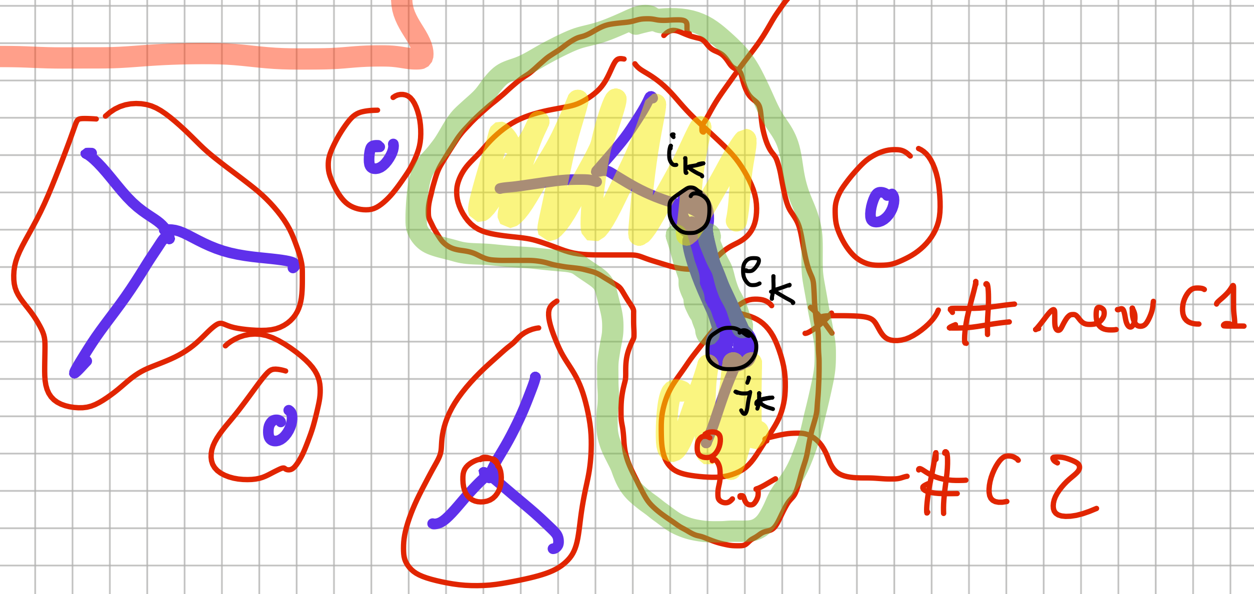
$T^* \cup \{e_k\}$ has NOT cycle

$\Longleftrightarrow$

$comp[i_k] \neq comp[j_k]$

$O(1)$

UPDATING



$$T^* := T^* \cup \{\underbrace{[i_k, j_k]}_{e_k}\}$$

"C1 ≠ C2"

$O(n)$

```
C1 := comp[i_k];
C2 := comp[j_k];
for v = 1 to n do
    if comp[v] = C2 then
        comp[v] := C1
    fi
rof
```

$n - 1$ unions $\rightarrow O(n^2)$

" too much ! "