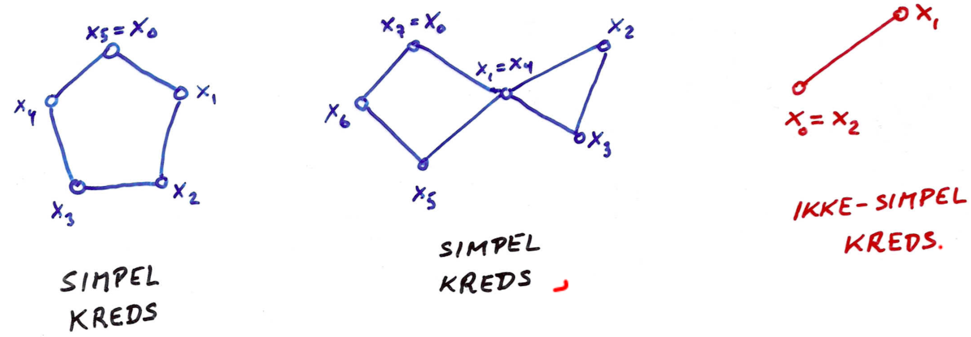
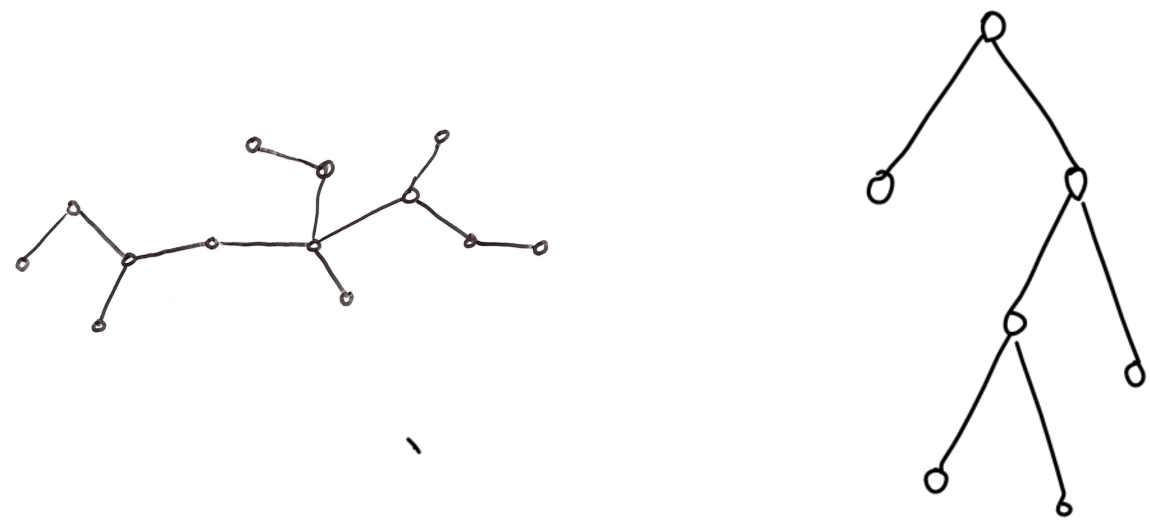
11.1



Definition. Et træ er en sammenhængende graf uden simple kredse.



Sætning 1.

En ikke-orienteret graf er et træ

$\Updownarrow$ hvis og kun hvis

der er en entydig simpel vej mellem ethvert par af punkter.

Bevis

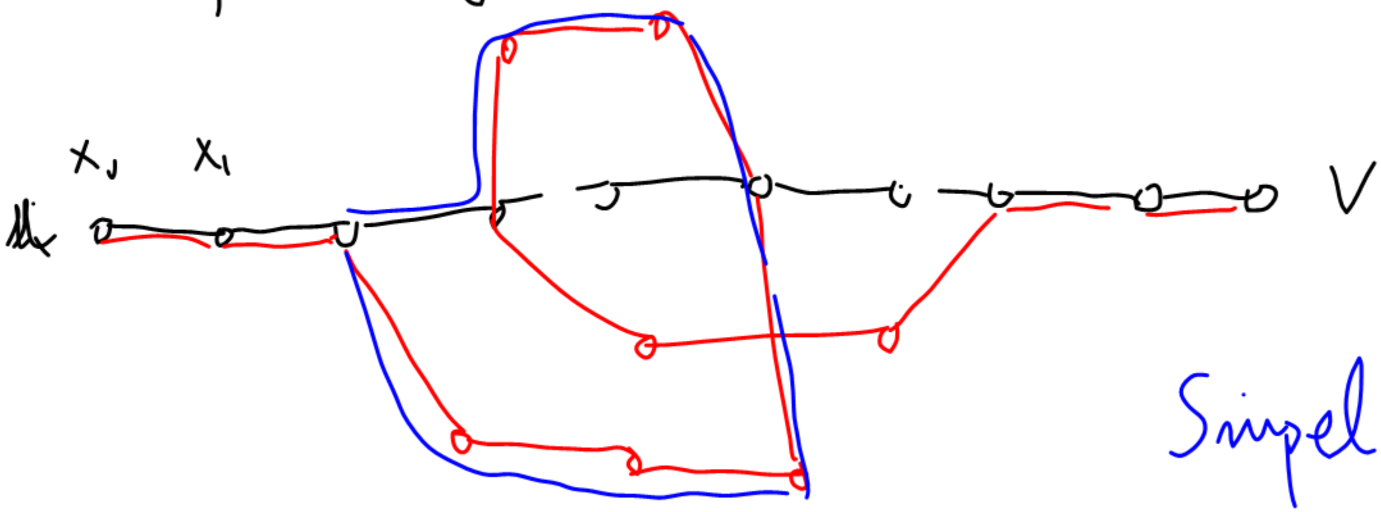
Kun hvis ($\Downarrow$)

Lad u, v være punkter i et træ T.

T sammenhængende. Der er en simpel $u-v$ vej,

Antag $u = x_0, x_1, \ldots, x_k = v$

og $u = y_0, y_1, \ldots, y_l = v$ er to forskellige simple veje.



Simpel kreds.

Modstrid. Der er altså kun én simpel $u-v$ vej.

<u>Hvis</u> ...

Sætning 2. Et træ med n punkter har præcis $n-1$ kanter.

Bevis ved induktion efter n.

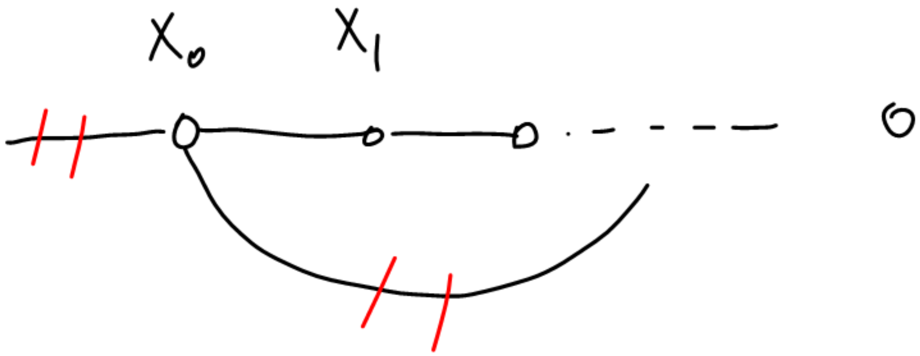
Basis $(n=1)$ OK

Induktion

Lad T være et træ med n punkter.

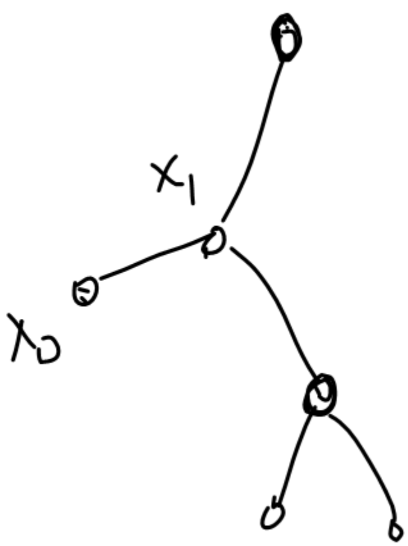
Antag Sætningen er sand for træer med $n-1$ punkter.

Lad $x_0, x_1, \ldots, x_l$ være en længste simpel vej i T.



$\deg(x_0) = 1$

T



$T' = T$ med x_0 og kanten $\{x_0, x_1\}$ fjernet

T' tre med $n-1$ punkter og $n-2$ kanter (ifølge antagelse).

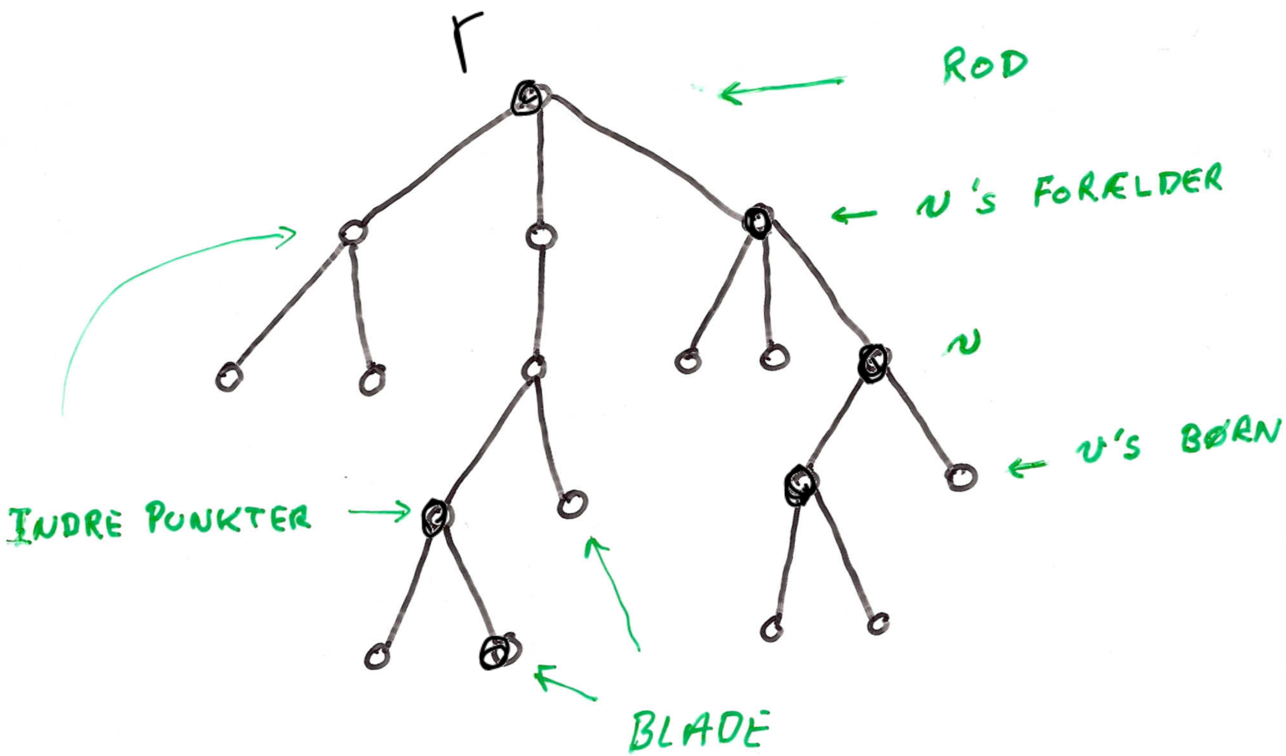
T har $n-2+1 = n-1$ kanter. $\square$

Træer med rod r.

v: et punkt i træet.
Hvis $v \neq r$ så er der en entydig nabo til v hvis afstand fra r er 1 mindre end v's afstand fra r. Dette punkt kaldes v's forælder.

De øvrige naboer til v (Hvis $v = r$: alle naboer til v.) har afstand fra r: 1 større end v's afstand fra r. Disse punkter kaldes v's børn.

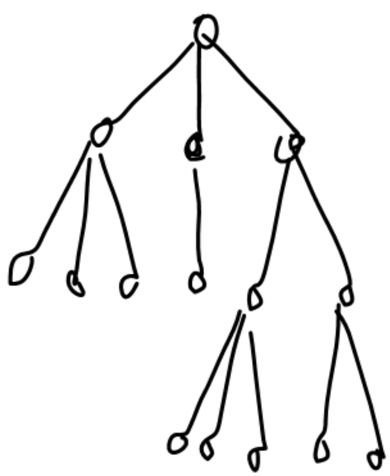
Punkter uden børn kaldes blade. Andre punkter er indre.


r
ROD
v's FORÆLDER
v
v's BØRN
INDRE PUNKTER
BLADE

m: positivt helt tal.
Et m-ært træ er et træ med rod hvor hvert punkt har højst m børn.
Hvis hvert indre punkt har præcis m børn så er det et fuldt m-ært træ.

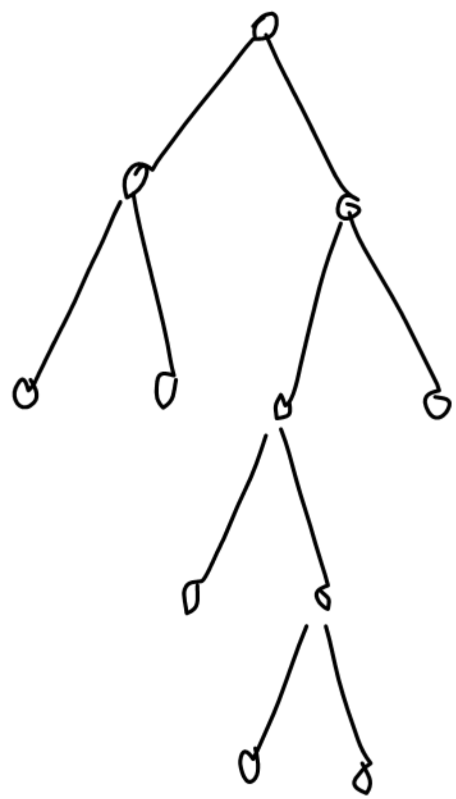
Et 2-ært træ kaldes binært.

I et ordnet binært træ kan et punkt have et venstre barn og et højre barn.



3-ært træ

Fuldt binært træ



Sætning 3. Et fuldt m-ært træ med i indre punkter har i alt $n = mi + 1$ punkter.

Hvert indre punkt har m børn

i alt $m \cdot i$ børn

Hvert punkt $\neq r$ er barn af indre punkt.

$$n - 1 = m \cdot i$$

Sætning 4. Betragt et fuldt m-ært træ med i indre punkter, ℓ blade og i alt n punkter.
For et fast m har vi da lineært ligningssystem med to ligninger og tre ubekendte (som er n, i og ℓ):

$$n = mi + 1$$

$$n = i + \ell$$

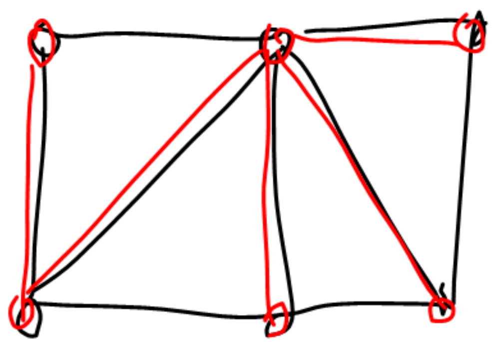
$$\begin{aligned} n - m \cdot i \quad &= 1 \\ n - i \quad - \ell &= 0 \end{aligned}$$

En vilkårlig af de tre variable kan vælges som fri variabel og de andre variable kan da udtrykket ved denne.

11.4

Definition.
Lad G være en ikke-orienteret graf.
Hvis T er et træ, som er delgraf af G og indeholder alle G's punkter så siger vi at T er et udspændende træ i G.

G



T

Sætning.
En ikke-orienteret graf G har et udspændende træ
hvis og kun hvis
G er sammenhængende.

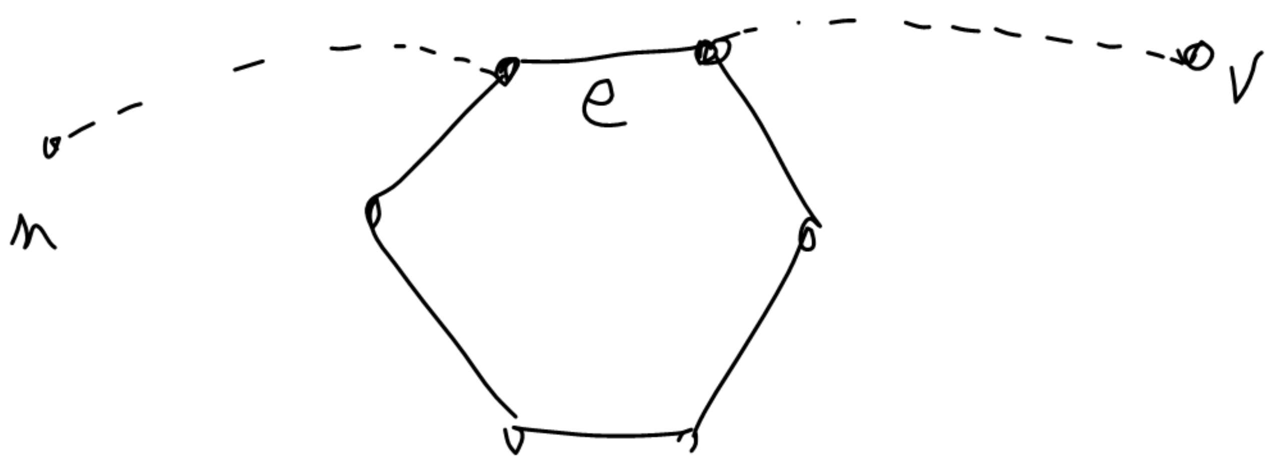
Bevis

Kun hvis $\Downarrow$ ...

Hvis:
G: en (endelig) sammenhængende graf.
Hvis G er et træ så er G et udspændende træ i G.

Ellers har G en simpel kreds.

e: kant på denne kreds.



$G' = G - e$. G' er sammenhængende

Hvis G' er et træ så er G' et udspændende træ i G.

Ellers: gentag ....

Formelt: brug induktion efter antal kanter.

11.5.

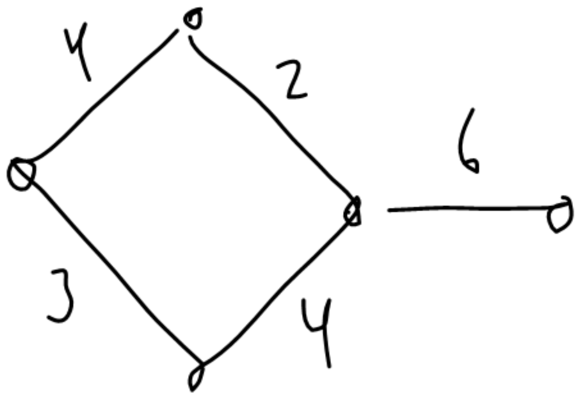
Lad G være en simpel, sammenhængende, vægtet graf med vægtfunktion $w : E \mapsto \mathbb{R}$.

Et minimum vægt udspændende træ i G er et udspændende træ T som opfylder at vægten

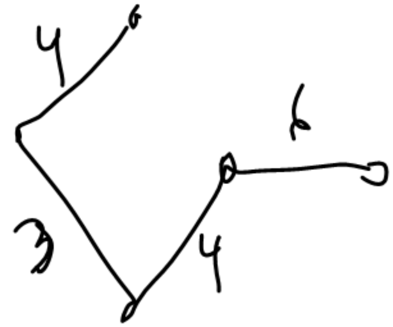
$$w(T) = \sum_{e \in T} w(e)$$

er mindst mulig.

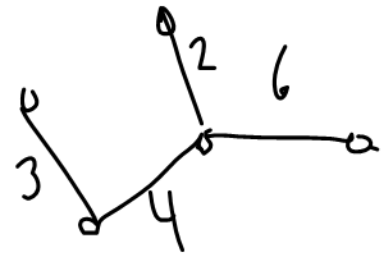
G



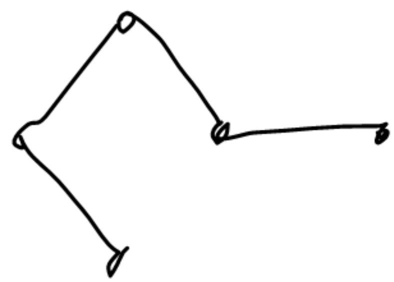
T



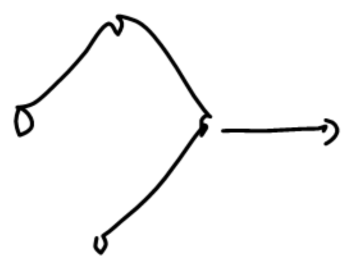
$w = 17$



$W = 15$



$w = 15$



$W = 16$

Procedure Prim (G: vægtet graf med n punkter)

T := træ bestående af ét punkt

for $i := 1$ **to** $n - 1$

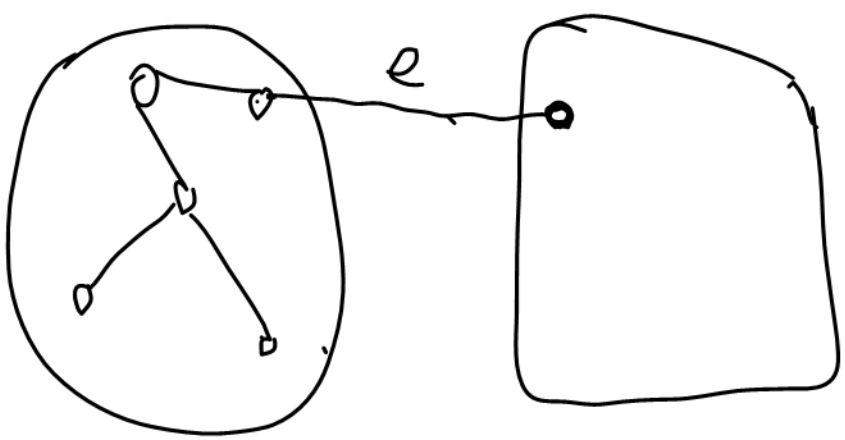
  e := kant med minimal vægt mellem

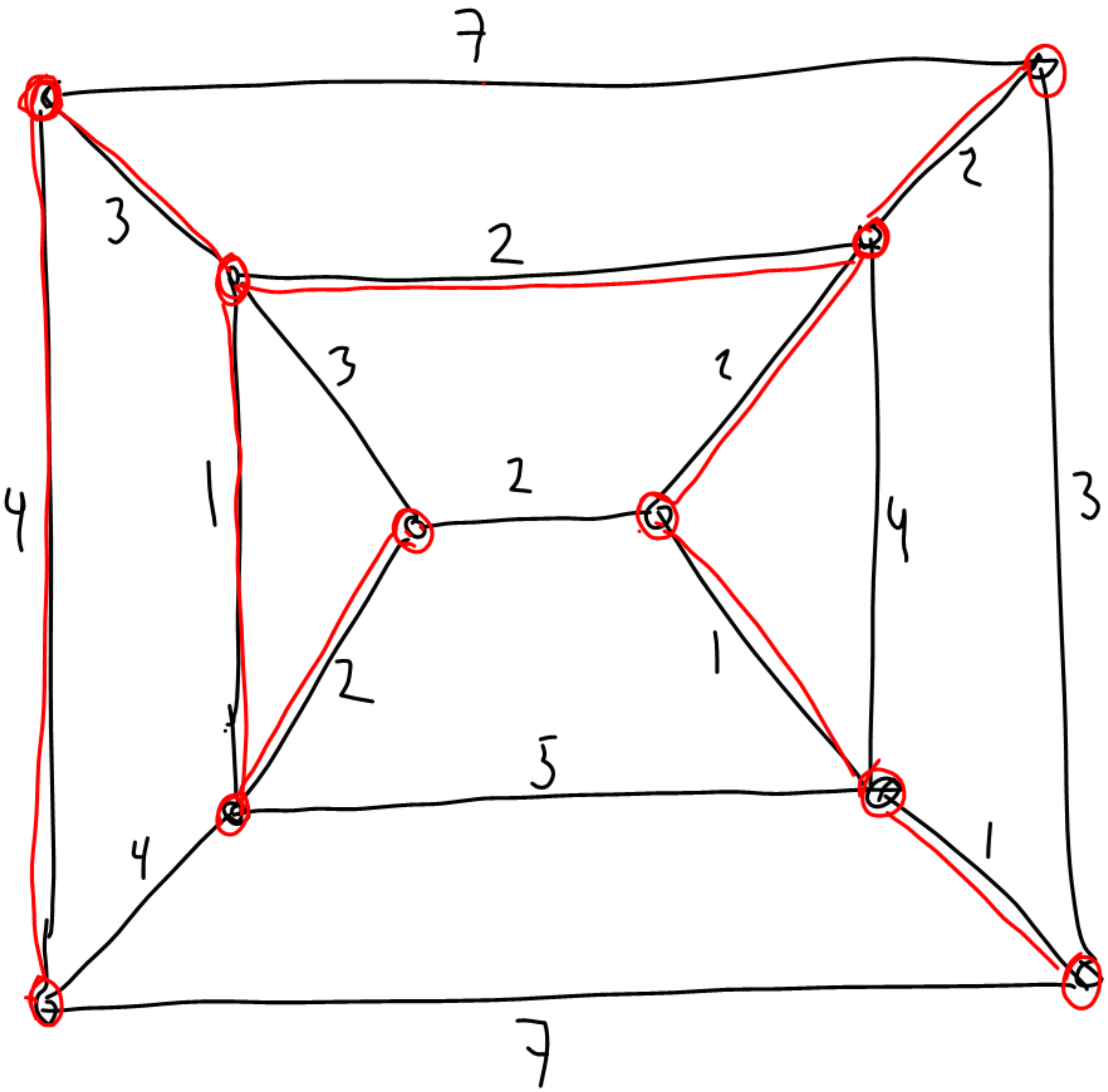
    et punkt i T og et punkt ikke i T

  Tilføj e og endepunkt til T

{ T er et minimum vægt udspændende træ}

T





Prim

Sætning. Prims algoritme finder et minimum vægt udspændende træ i en vægtet sammenhængende graf.

Bevis.
Det er "klart" at Prims algoritme finder et udspændende træ.
Vi kalder dette træ S.
Vi viser at S har minimum vægt.

Kanterne i S betegnes $e_1, e_2, \ldots, e_{n-1}$.
Kanterne tilføjes i algoritmen i denne rækkefølge.

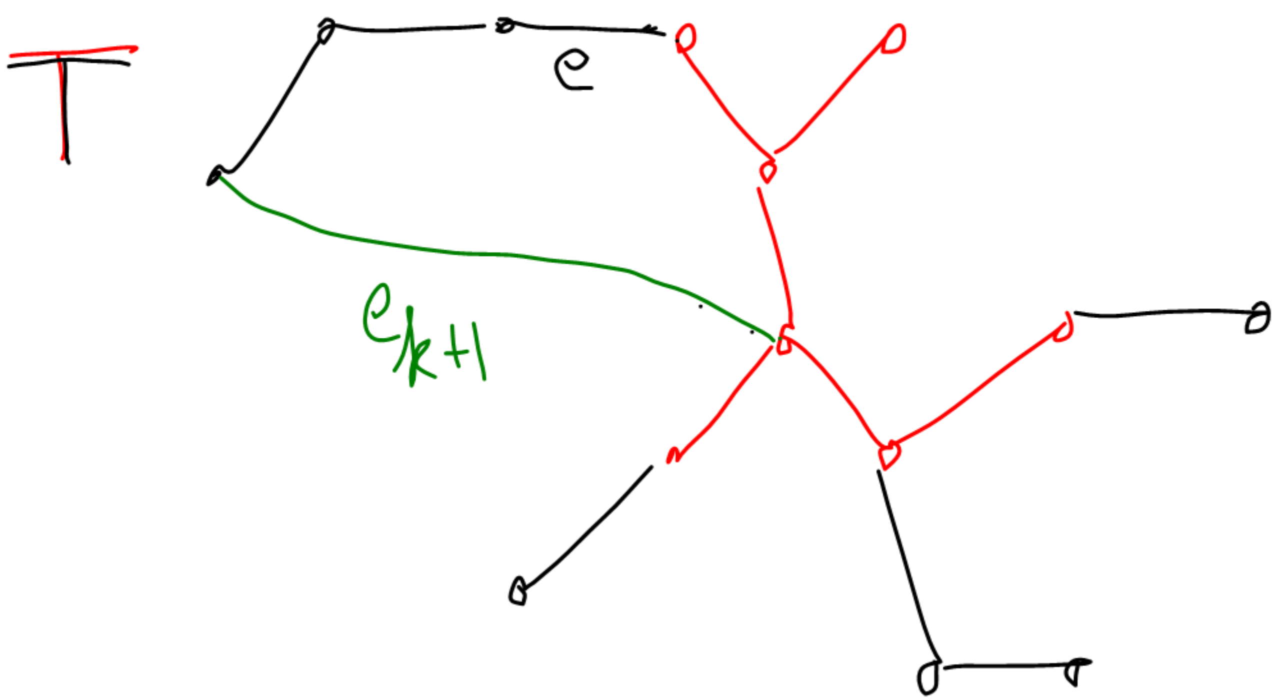
Lad T være et minimum vægt udspændende træ.

k: det største tal så $e_1, e_2, \ldots, e_k$ er i T
(evt. $k = 0$)

Velg T så k er størst mulig.

Skal vise at $S = T$ altså $k = n-1$.

Antag $k < n-1$



$e_1, \dots, e_k$

e_{k+1}

Prim velger e_{k+1} ikke e : $w(e_{k+1}) \leq w(e)$

$T' := T - e + e_{k+1}$.

T' er et udspændende træ

$$w(T') = w(T) - w(e) + w(e_{k+1}) \leq w(T)$$

T har minimum vægt $\Rightarrow w(T') = w(T)$

T' er minimum vægt udspændende træ.
som indeholder $e_1, e_2, \ldots, e_{k+1}$

Modstrid med valget af T og k.

Altså $k = n-1$ og $S = T$ er et min. vægt udsp. træ.

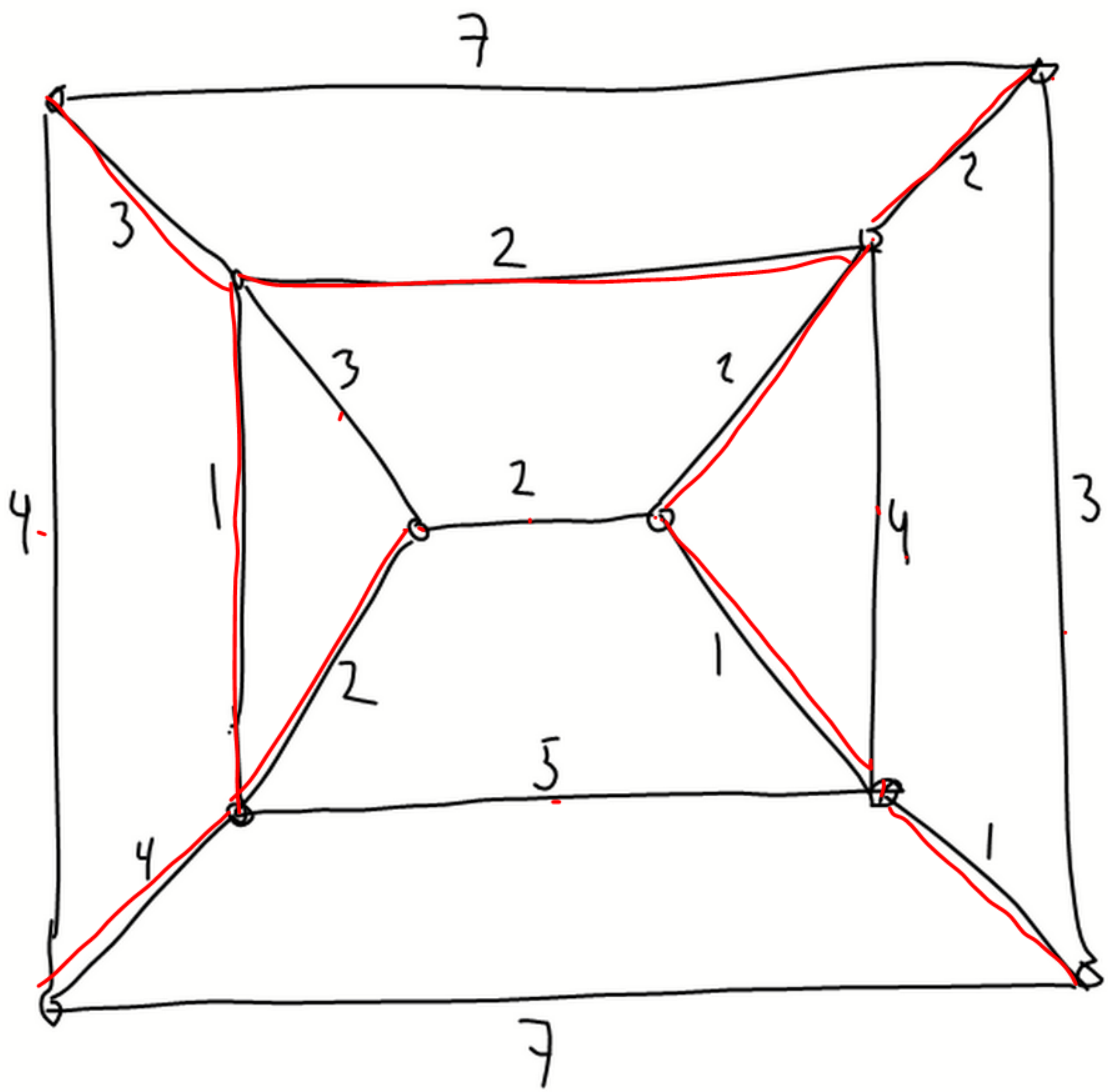
(Prims algoritme med en datastruktur, der giver en kompleksitet på $O(n^2)$.)

Procedure Prim ($G = (V, E)$: vægtet graf med n punkter)
T := træ bestående af ét punkt v_1
for alle punkter $u \neq v_1$
    if $\{v_1, u\} \in E$ **then** $L(u) := w(v_1, u)$ **else** $L(u) := \infty$
{For $u \notin T$: $L(u)$ = længden af korteste kant fra u til T}
for $i := 1$ **to** $n - 1$
    vælg $u \notin T$ så $L(u)$ er minimal
    $T := T \cup \{u\} \cup$ {korteste $u - T$ kant}
    for alle v hvor $\{u, v\} \in E, v \notin T$
        if $w(u, v) < L(v)$ **then** $L(v) := w(u, v)$

Procedure Kruskal(G: vægtet graf)
sorter G's kanter $e_1, \ldots, e_m$ efter vægt så $w(e_1) \leq w(e_2) \leq \ldots \leq w(e_m)$.
$T :=$ skov uden kanter
for $i := 1$ **to** m
  if $T \cup \{e_i\}$ ikke har simpel kreds **then**
    $T := T \cup \{e_i\}$

{ T er et minimum vægt udspændende træ.}

Kruskal



Kompeksitet af Kruskals algoritme: $O(m \log m)$

Kompleksitet af Prims algoritme: $O(n^2)$

$m \log n$

$m < n^2$

$\log m < 2 \log n$