Procedure Dijkstra($G = (V, E)$: vægtet sh. graf,
a, z: punkter)

{ Det antages at $w(e) > 0$ for alle $e \in E$}

For alle $v \in V$: $L(v) := \infty$

$L(a) := 0$, $S := \emptyset$

while $z \notin S$

begin

  $u :=$ punkt ikke i S, så $L(u)$ er mindst mulig

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

  For alle v hvor $\{u, v\} \in E$ og $v \notin S$

    if $L(u) + w(u, v) < L(v)$ **then**

    begin

      $L(v) := L(u) + w(u, v)$

      $F(v) := u$

    end

end {$z, F(z), F(F(z)), \ldots, a$ er en korteste vej fra z til a med længde $L(z)$}

Invariant:

1. Hvis $v \in S$ så er $L(v)$ længden af en korteste vej fra a til v i G. Denne vej er indeholdt i S.

2. Hvis $v \notin S$ så er $L(v)$ længden af en korteste vej fra a til v, hvor alle vejens punkter er i $S \cup \{v\}$.

3. Hvis $v \neq a$ og $L(v) < \infty$ så er der en vej fra a til v af længde $L(v)$, hvor vejens sidste kant er $\{F(v), v\}$.

Punkterne tilføjes til S i rækkefølge bestemt ved voksende afstand fra a.

Kompleksitet af Dijkstras algoritme:

Lad n være antallet af punkter i input-grafen.

Ved hvert gennemløb af while-løkken tilføjes et punkt til S.
While-løkken gennemløbes altså højst n gange.

Ved hvert gennemløb:
Find i en liste med (højst) n punkter et punkt med mindst L-værdi: $O(n)$

Undersøg hver af u's højst n naboer: $O(n)$

Komplesitet af den samlede algoritme: $O(n^2)$

Den handelsrejsendes problem

Travelling Salesman Problem (TSP).

Lad G være en vægtet graf med en Hamilton-kreds. F.eks. G er komplet graf (altså: en simpel graf med en kant mellem alle par af punkter).

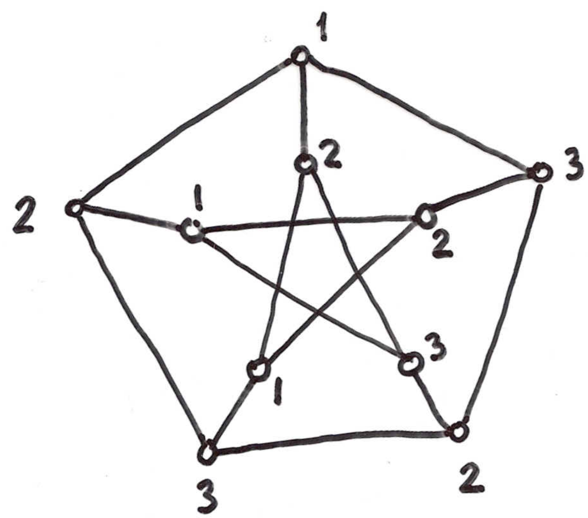
Find en *kortest* Hamilton-kreds i G.

En farvning med k farver af en graf $G = (V, E)$ er funtion $c : V \to K$, som opfylder at

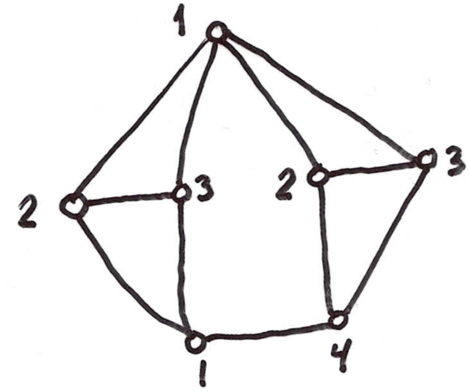
$$c(v) \neq c(u) \quad \text{for alle } \{v, u\} \in E,$$

hvor K er en mængde af k elementer (farver), f.eks. $K = \{1, 2, \ldots, k\}$.

Det kromatiske tal $\chi(G)$ af en graf G er det mindste tal k så G har en farvning med k farver.



1
2
3
2
1
2
3
1
3
2



1
2
3
2
3
1
4

Et beslutningsproblem er et problem hvor svaret på inputtet er enten JA eller NEJ.

Et beslutningsproblem tilhører klassen P hvis det kan løses af en algoritme i polynomiel tid. Altså en algoritme med kompleksitet $O(n^k)$, hvor n er størrelsen af inputtet og k er en konstant.

Et beslutningsproblem tilhører klassen NP (non-deterministisk polynomiel) hvis der findes en algoritme der med et ekstra input i polynomiel tid kan eftervise at svaret er JA.

$$P \subseteq NP$$

Eksempler på beslutningsproblemer:

Euler-kreds

Input: graf G

Spørgsmål: har G en Euler-kreds.

Euler-kreds er i P.

Kortest vej

Input: vægtet graf G, to punkter a og z, en konstant C.

Spørgsmål: har G en vej fra a til z af længde højst C.

Kortest vej er i P.

Hamilton-kreds

Input: graf G

Spørgsmål: har G en Hamilton-kreds.

Hamilton-kreds er i NP, ekstra input: en Hamilton-kreds.

TSP

Input: vægtet komplet graf K, en konstant C

Spørgsmål: har K en Hamilton-kreds af længde højst C.

Hamilton-kreds er i NP, ekstra input: en Hamilton-kreds.

k-farvning

Input: graf G

Spørgsmål: har G en farvning med k farver.

k-farving er i NP, ekstra input: en k-farvning.

2-farvning er i P.

Planar k-farvning

Input: planar graf G

Spørgsmål: har G en farvning med k farver.

Planar 2-farvning er i P.

Planar 4-farvning er i P.

Givet: beslutningsproblemer A og B hvor mængden af mulige inputs er hhv. S_A og S_B.
En funktion $f : S_A \rightarrow S_B$ som opfylder

- A's svar på input x = B's svar på input $f(x)$

- der findes en algoritme der beregner $f(x)$ i polynomiel tid

kaldes en polynomiel tids reduktion af A til B.

Hvis der findes en algoritme der løser B i polynomiel tid så findes der en algoritme der løser A i polynomiel tid.

Et problem er NP-komplet hvis

- det er i NP, og

- ethvert problem i NP har en polynomiel tids reduktion til dette problem.

Hvis et NP-komplet problem kan løses i polynomiel tid så kan ethvert problem i NP løses i polynomiel tid og så er $P = NP$.

Eksempel:

Vi ved at Hamilton-kreds er NP-komplet.
For at vise at TSP er NP-komplet skal vi

- vise at TSP er i NP,
- vise at der findes polynomiel tids reduktion af Hamilton-kreds til TSP.

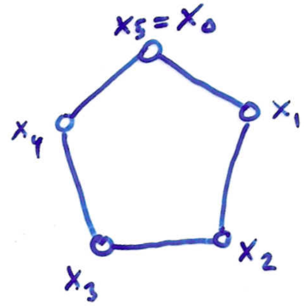
Reduktion af Hamilton-kreds til TSP:

$G = (V, E)$: input til Hamilton. Antal punkter i G er n.

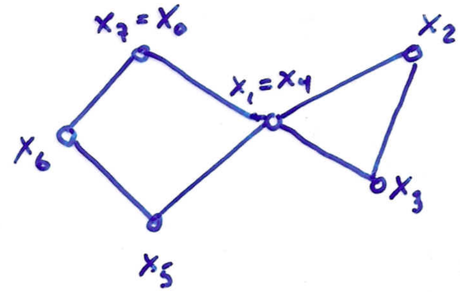
Input til TSP:
K: komplet vægtet graf med punktmængde V.
  Hvis $\{u, v\} \in E$ så $w(u, v) := 1$.
  Hvis $\{u, v\} \notin E$ så $w(u, v) := 2$.
$C = n$.

G har en Hamilton-kreds hvis og kun hvis K har en Hamilton-kreds med længde højst C.

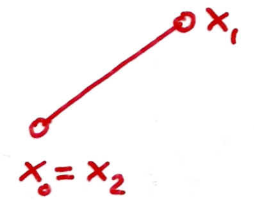
K, C kan beregnes i polynomiel tid.



SIMPEL KREDS

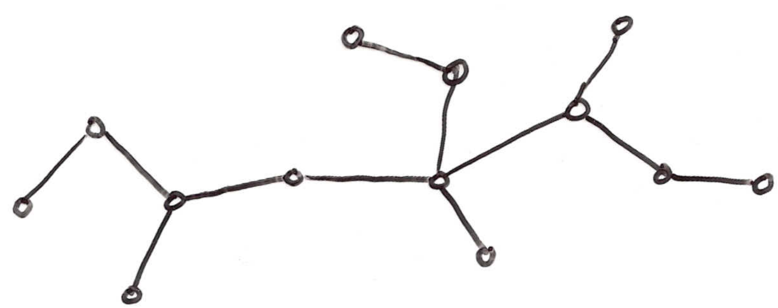


SIMPEL KREDS



IKKE-SIMPEL KREDS

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



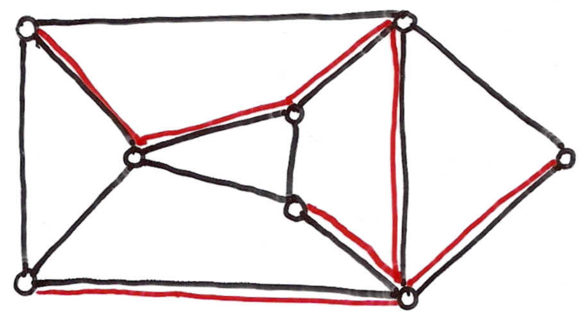
Sætning.
En ikke-orienteret graf er et træ
hvis og kun hvis
der er en entydig simpel vej mellem ethvert par af punkter.

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

Definition 10.4.1. Lad G være en ikke-orienteret graf. Hvis T opfylder

- T er et træ,
- T er en delgraf af G og
- T indeholder alle G's punkter,

så siges T at være et udspændende træ i G.



G

T : ET UDSPÆNDENDE TRÆ I G

Sætning.
G har et udspændende træ
$\Updownarrow$
G er sammenhængende.

Korollar. En sammenhængende graf G med n punkter og $n-1$ kanter er et træ.

Bevis. G har et udspændende træ T, da G er sammenhængende.
T har n punkter og dermed $n-1$ kanter.
Altså $G = T$. $\square$

Sætning. En graf uden simple kredse med n punkter og $n-1$ kanter er et træ.