MA-180

Kapittel 1

Regel for sum

- Hvis
  - en oppgave kan utføres på m måter
  - en annen oppgave kan utføres på n måter
  - de to oppgavene kan ikke utføres samtidig
- Så
  - kan det å utføre en av oppgavene utføres på n+m måter

Eksempel

- I en studentgrupper er det 11 jenter og 23 gutter. Du skal velge en student. Hvor mange valg har du?
- 11+23

- I programmering grunnkurs hadde vi 3 alternative bøker i Python, 5 i Java og 7 i C.
- Hvor mange alternative har vi nå vi skal velge en bok?
- 3+5+7

Regel for produkt

- Hvis
  - en oppgave kan deles inn i to faser der
  - den første fasen kan utføres på n måter
  - den andre fasen kan utføres på m måter
- Så
  - kan oppgaven utførste på n*m måter

Eksempel 1

- I en studentgrupper er det 11 jenter og 23 gutter. Du skal velge to tillitsvalgte, men du skal velge en fra hvert kjønn. Hvor mange mulige valg har du?
- Du kan gjøre oppgaven slik:
  - velge en gutt og deretter velge en jente
  - Første fase 23 valg, andre fase 11 valg, totalt 11*23 valg
- Kan du velge på en annen måte?
- Du kan gjøre oppgaven slik:
  - velge en jente og deretter velge en gutt
  - Første fase 11 valg, andre fase 23 valg, totalt 23*11 valg

Eksempel 2

- Du skal finne alle mulige utfall av 5 håndballkamper der et utfall er hjemmeseier, uavgjort eller borteseier
- Vi kaller antall mulige utfall av n håndballkamper for A(n). Oppgaven er å finne A(5)
- Du kan velge ved først å velge et resultat av den en tilfeldig kamp og deretter velge et resultat av de 4 resterende kampene
- Den første fasen har 3 muligheter -A(1)- og den siste har A(4), vet betyr at
  A(5)=4*A(4)
- Tilsvarende blir A(4)=3*A(3),…
- A(5)=3*A(4)=3*3*A(3)=3*3*3*A(2)=3*3*3*3*A(1) = 3^5

Eksempel 3

- Hvor mange ulike tilhengerskilt kan du lage når et bilskilt består av to bokstaver (utenom æøå) og 4 siffer der det første ikke kan være 0
- Du kan velge slik:
  - Velg bokstav 1
  - Velg bokstav 2
  - Velg siffer 1
  - Velg siffer 2
  - Velg siffer 3
  - Velg siffer 4
- Antall mulige tilhengerskilt blir 26*26*9*10*10*10

Eksempel 4

- Du skal trekke to kort ut fra en stabel med 52 ulike kort. Hvor mange mulige utvalg har du?
- En måte å gjøre oppgaven
  - Trekk ett kort
  - Trekk ett kort til
- Den første fasen kan gjøres på 52 måter
- Når du trekker ett kort til, så er det 51 kort igjen.
- Svar: 52*51

Eksempel 5

- Du skal trekke ut 52 kort fra en stable med 52 ulike kort. Hvor mange måter kan du gjøre det
- Metode:Velg første,velg andre, velg tredje,…, velg siste
- Svar 52*51*50*…*3*2*1=52! (leses 52 fakultet)

Eksempel 6

- Fra en gruppe på 10 studenter, så skal det velges 4 som skal sitte på første rad i forelesningssalen på plassene 23,24,25 og 26. Finn antall mulige måter å velge de 4 studentene.
- Første forsøk
  Velg studentene i rekkefølge, dvs velg sete 23,så sete 24, sete 25 og sist sete 26
- Antall mulige valg er da 10*9*8*7
- Som også kan skrives slik

$$10 * 9 * 8 * 7 = \frac{10 * 9 * 8 * 7 * 6 * 5 * 4 * 3 * 2 * 1}{6 * 5 * 4 * 3 * 2 * 1} = 10!/6!$$

- **Generelt**
- **Skal du velge ut r fra n og rekkefølgen teller, så kan dette gjøres på $\frac{\boldsymbol{n}!}{(\boldsymbol{n}-\boldsymbol{r})!}$ måter**

Permutasjoner - rekkefølgen

- **Definisjon**
  Hvis en samling av n ulike objekter skal plasseres i rekkefølge, så kaller vi en rekkefølge en permutasjon av objektene.
- Har vi ulike objekter så finnes det n! permutasjoner av disse objektene
- Eksempel A,B,C
  ABC, ACB, BAC,BCA,CAB,CBA

Bruk av begrep

- Skal du velge ut r fra n og rekkefølgen teller, så kan dette gjøres på $\frac{n!}{(n-r)!}$ måter
- Eller
- **Har du n ulike objekter, så kan du lage $\boldsymbol{\frac{n!}{(n-m)!}}$ permutasjoer ved å velge r elementer, $\boldsymbol{r \leq n}$**

Eksempel

- Hvor permutasjoner kan du lage av de fire bokstavene BALL
- OBS! Det er ikke 4!=24
- Grunnen er at L forekommer to ganger slik at
  du har ikke 4 ulike objekter.
- Siden L forekommer to ganger, så vil vi, jfr tabell til høyre, telle samme permutasjonen to ganger (2!)
- Korrekt svar $\frac{4!}{2!}$

Table 1.1

A	B	L	L	A	B	L_1	L_2	A	B	L_2	L_1
A	L	B	L	A	L_1	B	L_2	A	L_2	B	L_1
A	L	L	B	A	L_1	L_2	B	A	L_2	L_1	B
B	A	L	L	B	A	L_1	L_2	B	A	L_2	L_1
B	L	A	L	B	L_1	A	L_2	B	L_2	A	L_1
B	L	L	A	B	L_1	L_2	A	B	L_2	L_1	A
L	A	B	L	L_1	A	B	L_2	L_2	A	B	L_1
L	A	L	B	L_1	A	L_2	B	L_2	A	L_1	B
L	B	A	L	L_1	B	A	L_2	L_2	B	A	L_1
L	B	L	A	L_1	B	L_2	A	L_2	B	L_1	A
L	L	A	B	L_1	L_2	A	B	L_2	L_1	A	B
L	L	B	A	L_1	L_2	B	A	L_2	L_1	B	A

(a) (b)

Eksempel

- Hvor mange permutasjoner kan vi lage av de ni bokstavene i DATABASES
- Løsning lag en frekevenstabell

A	B	D	S	T
3	1	1	2	1

- Svar:
- $\frac{9!}{3!2!}$

Regel

- Hvis du har n objekter med n_1 entydige objekter av type 1, n_2 n1 entydige objekter av type 1,.., n_r entydige objekter av type r,
  så finnes det

$$A = \frac{n!}{n_1! \cdot n_r! \dots n_r!}$$

permutasjoner av de n objektene

Eksempel

- Du skal bevege deg i planet fra (2,1) til (7,4) og du har kun lov å bevege ved å velge å gå til høyre en lengdenehet(R) eller opp en lengdeenhet (U)?
- Hvor mange forskjellige veier kan du velge?

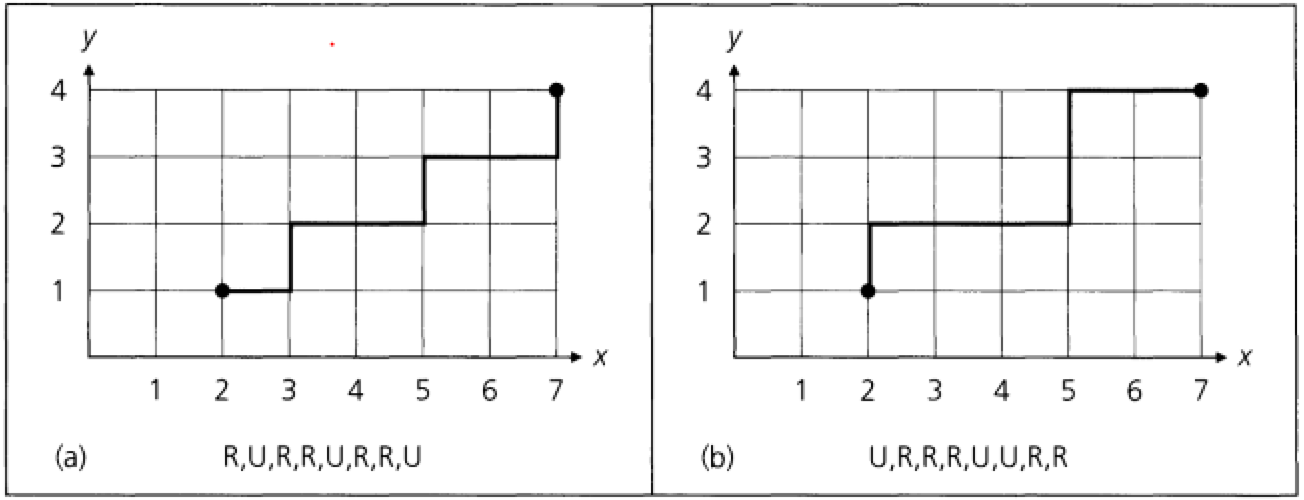


Figure 1.1

- Du må velge 3 U og 5 R i en eller annen rekkefølge, dvs du må ha 8 steg
- Svar: Du har 8 objekter som kan deles ti to ulike grupper, en med 3 elemetner og en med 8 elementer

$$\frac{8!}{3! \cdot 5!} = 56$$

Algoritme

- En algoritme er en metode for å løse et problem som beskriver
- 1. hvilke handlinger som må utføres
- 2. rekkefølgen handlingene skal utføres
- Ofte kan et problem løses på flere måter

Forfatter:	Abu Ja’far Mohammed ibn Mûsâ al-Khowârizm:
Tittel:	Kitab al jabr w’al-muqabala(rules of restoration and reduction
År:	825
Land:	Persia

Tittelen er opphav til algoritme og algebra

Polynomberegning

- $p_n(x) = a_0 + a_1x^1 + \cdots + a_nx^n$
- Hvor mange addisjoner er det i utregning av en funksjonsverdi:
- Hvor mange multiplikasjoner*:

addisjoner : n
multiplikasjoner*: 0+1+…+n=(n+1)*n/2≈n^2

Polynomberegning alternativ

- Alternativ
- Eksempel $f(x) = 5x^3 - 4x^2 + 7x + 3$

z=5
z=z*x-4
z=z*x+7
z=z*x+3

$\Rightarrow$

Z=((5*x-4)*x+7)*x+3
$=5x^3 - 4x^2 + 7x + 3$

Da har vi 3 addisjoner og 3 multiplikasjoner, generelt: et polynom av grad n kan beregnes med n addisjoner og n multiplikasjoner

n=100, addisjoner likt, multiplikasjoner redusert fra 5050 til 100

Kode

```
/**
 * evaluate polynom function value
 * @param a defines a[0]+a[1]x+a[2]x^2 etc
 * @param x
 * @return polynom value
 */
public  int evaluate1(int[] a,int x){
    int res=0;
    for (int i=0;i<a.length;i++){
            res+=a[i]*Math.pow(x, i);
    }
    return res;
}
/**
 * evaluate polynom function value
 * @param a defines a[0]+a[1]x+a[2]x^2 etc
 * @param x
 * @return polynom value
 */
public int evaluate2(int[] a,int x){
    int size=a.length;
    if (a==null) return 0;
    if (size==0)return 0;
    int res=a[size-1];
    for (int i=size-2;i>=0;i--){
            res=res*x+a[i];
    }
    return res;
}
```

Fibonacci-tallene

- Definisjon av en Fibonacci-tallfølge
  - $F_0=0$
  - $F_1=1$
  - $F_n=F_{n-1}+F_{n-2}$, $n\geq 2$
- Brute force programmering
- Programmer en rekursiv funksjonen

  F(n)= { hvis n==0, returner 0;
  hvis n==1, returner 1;
  hvis n ≥2, returner F(n-1)+F(n-2)}
- Se på arbeidsmengden

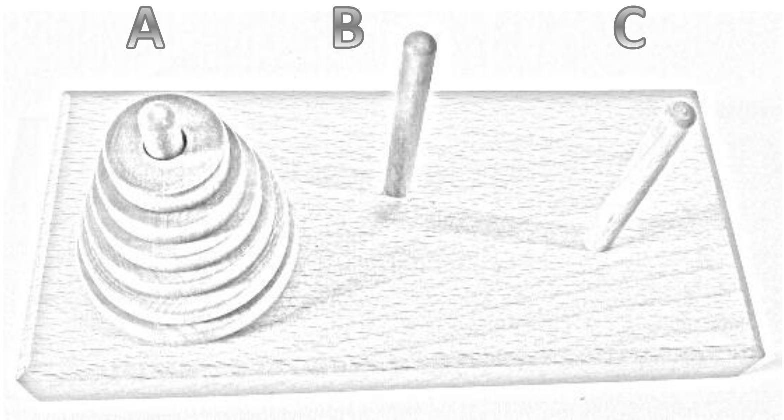
Optimalisering

- $F_2=F_1+F_0=0+1=1$
- $F_3=F_2+F_1=1+1=2$
- $F_3=F_2+F_1=2+1=3$ videre svar
- Bruk en tabell

Mitt ønske er at
dette ønsket skal bli oppfylt

Hvordan løse "Hanois tårn "

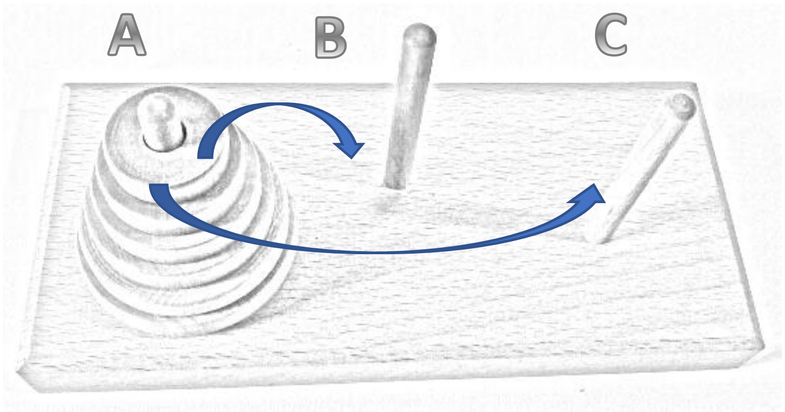
Hanois tårn



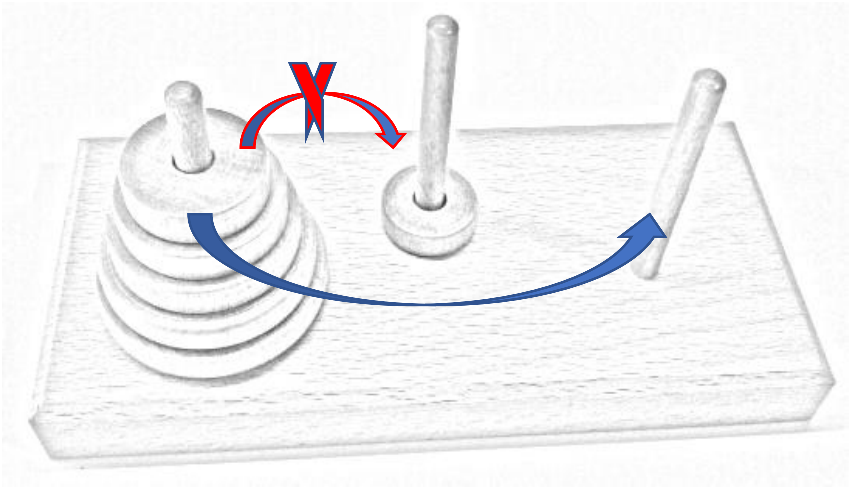
- Hanois tårn er et matematisk spill som består av 3 stenger festet til en plate og 7 ringer.
- Målet med spillet er å utføre lovlige trekk slik at alle ringene fra stang A flyttes til stang C.

Regler

- Reglene for lovlige trekk
  - Et trekk består i å flytte en og bare en ring fra en stang til en annen
  - I et trekk er det **ikke** lov å plassere en større ring på en mindre

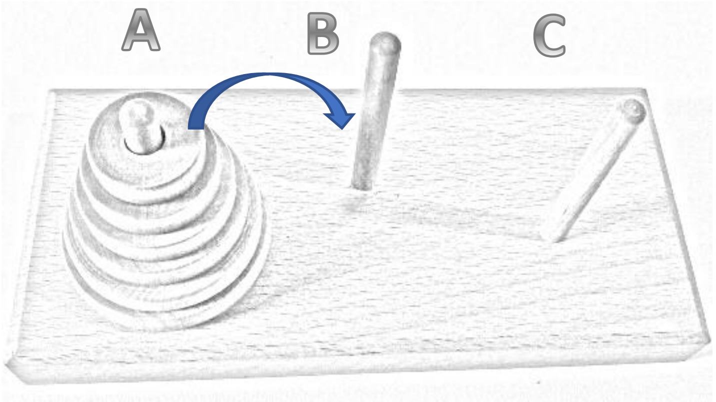


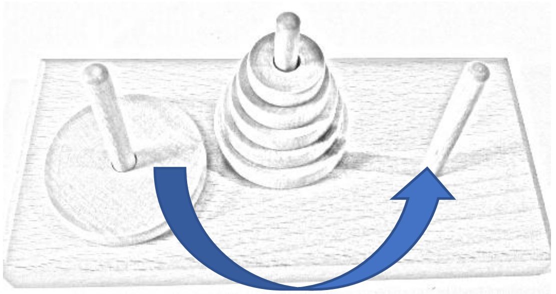
- Lovlige flytt i figuren over:
  Den øverste ringen på A kan flyttes til pinne B eller pinne C

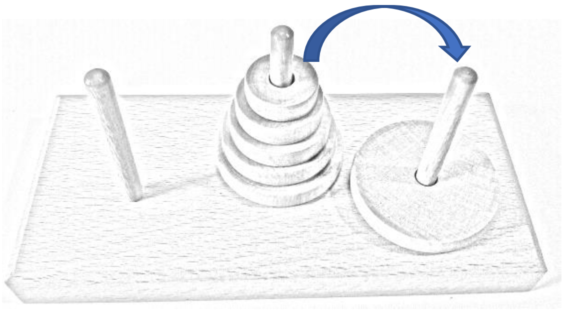


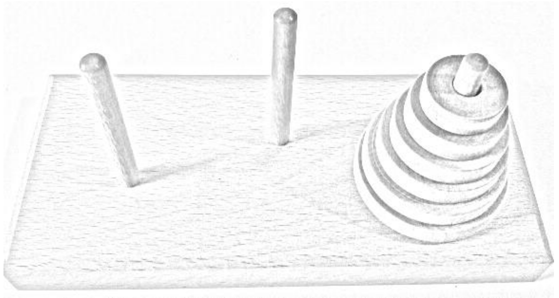
Hvordan tenke

- Det finnes en litt uvant måte å tenke på som løses problemet
- Hadde jeg bare kunnet!





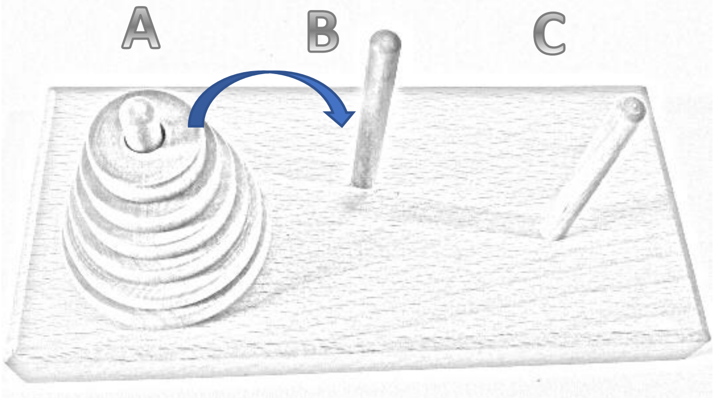




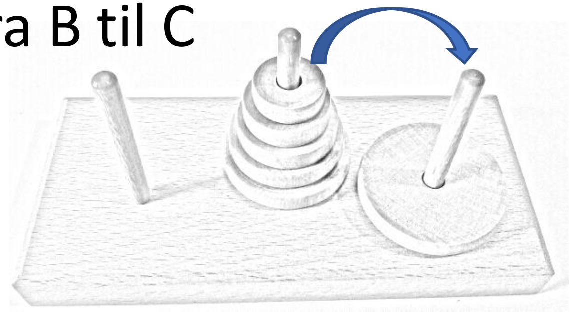
- Problemet har nå en løsning, fordi løsningen inneholder det samme problemet, men med en mindre ring.

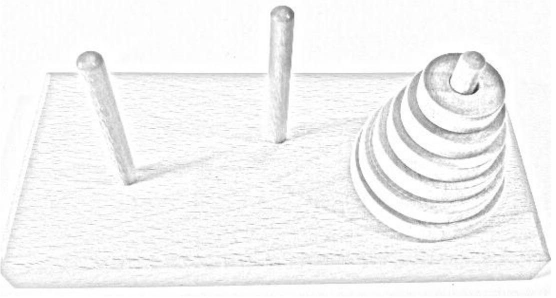
Hvordan tenke

- Vi skal flytte 7 ringer fra A til C.



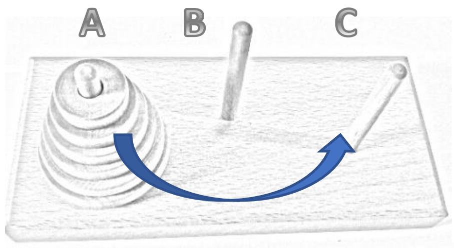
- Løsning
  - Flytt 6 ringer fra A til B
  - Flytt 1 ring fra A til C
  - Flytt 6 ringer fra B til C



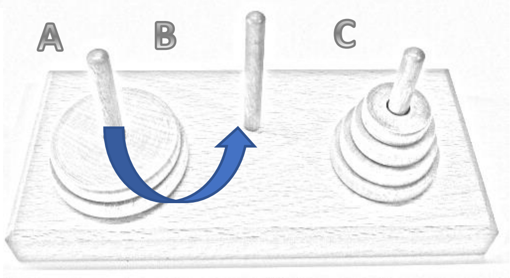


- Vi fortsetter med å gjøre problemet mindre helt til vi har vi bare trenger å flytte en ring. Da er vi ferdig.

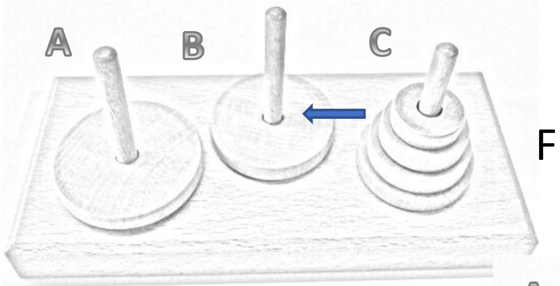
Hvordan flytte 6 ringer fra A til B



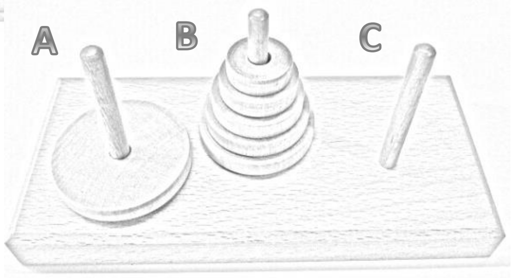
Flytt 5 ringer fra A til C



Flytt 1 ringer fra A til B



Flytt 5 ringer fra C til B



Done

Finn de to punktene som er ligger tettest

- Gitt en mengde med n punkter i planet.
- Finn to punkter som har den minste avstanden. (det kan være flere)

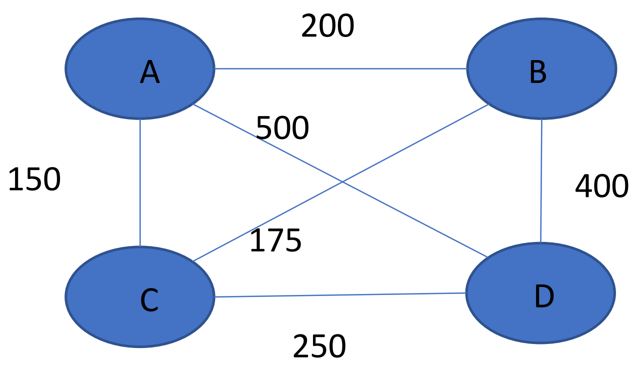
La punktene være nummerert p1 til pn. Beregn alle di,j=|pj-pi|. Hvis vi tenker dette som en matrise, så er di,j=dj,i og di,i skal vi ikke beregne(fordi de alltid er null).Finn den minste av disse

```
double dMin=Double.MAX_VALUE;
int iMin=0, jMin=0;
for (int i=1 ; i<=n ; i++){
  for (int j=i+1 ; j<=n ; j++){
      double d =|pj-pi|;
      if (d<dMin){
         dMin=d;     iMin=i;jMin=j;
      }
  }
}
//iMin, og jMin er to punkter som har kortest avstand
```

Antall avstandsberegninger blir i størrelseorden n^2

Traveling salesman problem

- Gitt n byer med en kjent avstand mellom alle byene
- Finn den den korteste ruta som går gjennom alle byene en gang før den returnerer til utgangsbyen



Hvordan du finner alle mulige reiseruter

- For hver by, velg en av de andre byene, velg en av de andre byene som ikke er valgt, helt til du ikke har flere alternativer

A-B-C-D-A
A-B-D-C-A
A-C-B-D-A
A-C-D-B-A
A-D-B-C-A
A-D-C-B-A

Psudokode

Bruk en stakk med nodesekvenser
Push en node A
Så lenge det er noder på stakken
Pop dvs først A
Push naboene til den siste i sekvensen som ikke er sekvensen som en forlengelse
dvs AB,AC,AD
Hvis det ikke er naboer,
en løsning er sekvensen med A som siste