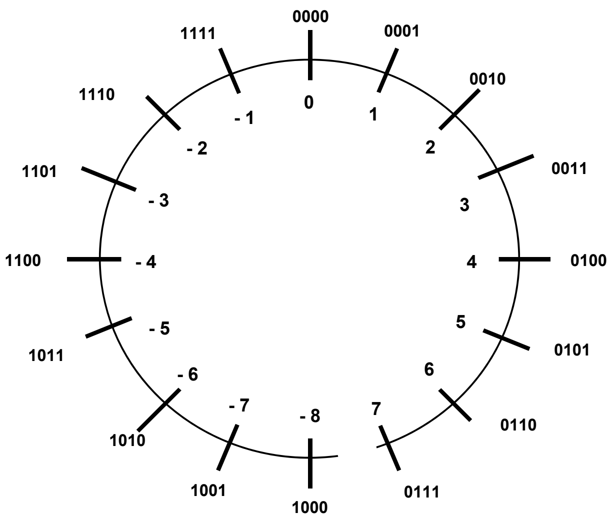
Tall



(Kapittel 7.1, 7.4-7.8, 8 + Appendiks B)

Læringsmål – tall

- Kunne prefikser for store tall i
  - titallsystemet
  - det binære tallsystemet
- Forstå ulike prinsipper for representasjon av
  - negative heltall
  - reelle tall
- Forstå ulike prinsipper for representasjon av geometriske former

Store tall

- For å håndtere store tall i titallsystemet bruker vi en-bokstavs SI-symboler som betegner potenser av 1000
  - k (kilo) = 10^3, M (mega) = 10^6, G (giga) = 10^9, T (tera) = 10^{12},
  - P (peta) = 10^{15}, E (exa) = 10^{18}, Z (zeta) = 10^{21}, Y (yotta) = 10^{24}
  - (Merk at vi her bruker k for 1 000, fordi K i SI-systemet er en temperatur.)
- Anta at vi har et digitalt bilde med 1 024 x 1 024 piksler (bildeelementer)
  - La hvert piksel representeres med 1 byte
  - Bildets størrelse blir oftest angitt til 1 MB
  - Men bildet er jo 1 024 x 1 024 x 1 byte = 1 048 576 byte ≈ 1.05 MB
  - Denne feilen øker jo større tall vi snakker om!

Prefikser for potenser av 1 024

- SI-prefiksene k, M, G osv er desimale enheter og har ingen mening som potenser av 1 024.
- IEC publiserte i 1999 en standard for potenser av 1 024.
- Navnene er satt sammen av de to første bokstavene i SI-prefiksene pluss bi for ”binær”.
- Les mer om dette i Appendiks B i læreboka!

Navn	Symbol	Potens av 2 og verdi i titallsystemet
kibi	Ki	2^{10} = 1 024
mebi	Mi	2^{20} = 1 048 576
gibi	Gi	2^{30} = 1 073 741 824
tebi	Ti	2^{40} = 1 099 511 627 776
pebi	Pi	2^{50} = 1 125 899 906 842 624
exbi	Ei	2^{60} = 1 152 921 504 606 846 976
zebi	Zi	2^{70} = 1 180 591 620 717 411 303 424
yobi	Yi	2^{80} = 1 208 925 819 614 629 174 706 176

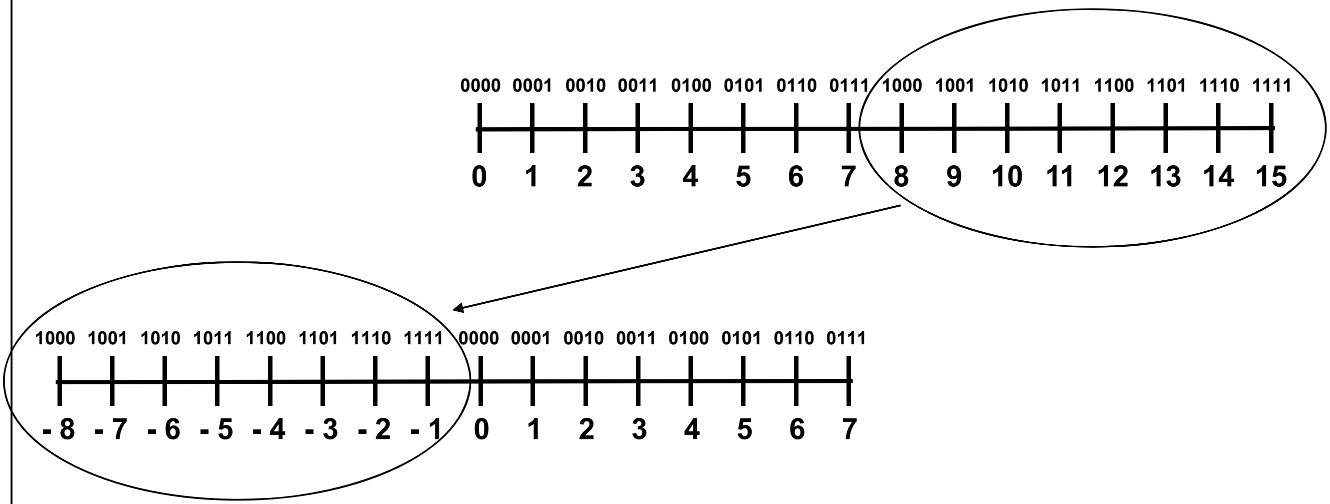
Noen konvensjoner det er nyttig å kjenne …

- **Størrelsen på RAM, ROM eller flash-minner gis som regel i binære enheter.**
- **Kapasiteten til harddisker og lagre som betraktes som en stor disk oppgis i desimale enheter.**
  - **Sektorstørrelsene på en disk gis nesten alltid i toerpotenser, siden de mapper direkte til RAM.**
  - **Det finnes en forvirrende hybrid, der en ”megabyte” betyr 1000 ”kilobytes” a 1024 byte.**
  - **En ”1.44 MB” diskett er verken 1.44 × 220 byte eller 1.44 × 106 byte,
men 1.44 × 1 000 × 1 024 bytes (som er ca 1.406 MiB, eller 1.475 MB).**
- **Dette kan også gjelde disk-lignende flashminner (toerpotens multipler av desimale megabyte!)**
- **Kapasiteten til en CD er alltid gitt i binære enheter.**
  - **En ”700 MB” CD har en nominell kapasitet på 700 MiB.**
- **Kapasiteten til en DVD er gitt i desimale enheter.**
  - **En 4,7 GB DVD har en nominell kapasitet på 4,38 GiB.**
- **Overføringskapasitet uttrykkes som bps (biter per sekund) eller Bps (byte per sekund)**
  - **angis alltid i titallsystemet, med SI-prefikser
eksempel: kBps (103 byte per sekund), MBps (106 byte per sekund), osv**

Ulike klasser tall

- **De naturlige tallene:**
$\mathbb{N}$ = { 1, 2, 3, … }
- **De hele tallene:**
$\mathbb{Z}$ = { …, -2, -1, 0, 1, 2, … }
- **De rasjonale tallene:**
$\mathbb{Q}$ = alle tall som kan skrives som en brøk
- **De reelle tallene:**
$\mathbb{R}$ = alle tallene på tallinjen

Negative tall: toer-komplement



- For å negere et tall:
  1. Snu alle bit'ene i tallet (0 ↔ 1).
  2. Legg til 1.

Regning med toer-komplement

- Hvis vi legger sammen to tall og får en ekstra mente til venstre, kan den bare kastes.
  Eksempel:
  -1 + 4:
- Men: Hvis de to mentene lengst til venstre er ulike, har vi overflyt og ugyldig svar.
  Eksempler:
  3 + 5: -8 - 1:

Negative tall: Bias

- Et alternativ er å legge til en konstant *bias* til alle tallene.
- Med 8 bitposisjoner og bias 128 kan tallene fra -128 til 127 representeres ved hjelp av tallene fra 0 til 255.
- Eksempler:
  53: -21:

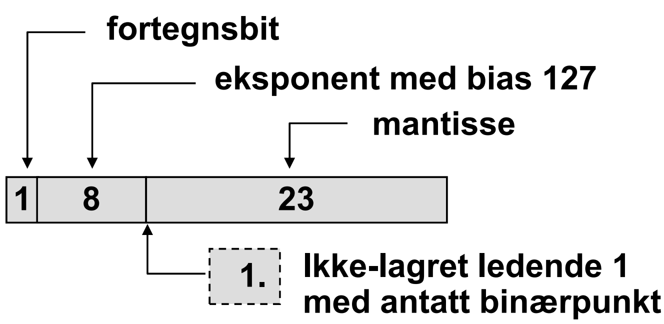
Regning med bias

- Ved addisjon kommer bias med to ganger, så vi må trekke den fra igjen (en gang).
- Eksempel:
  53 + (-21):

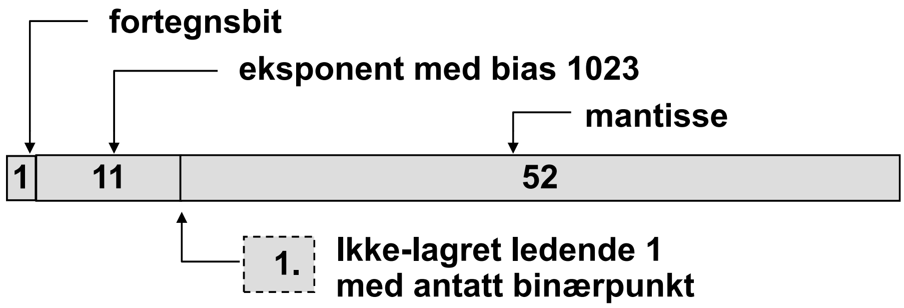
Flyttall

- I titallsystemet kan et tall skrives på formen
  mantisse * $10^{eksponent}$
- Eksempler:
  $0.5 * 10^2 = 0.5 * 100 = 50$
  $0.5 * 10^0 = 0.5 * 1 = 0.5$
  $0.5 * 10^{-1} = 0.5 * 0.1 = 0.05$
  $-5 * 10^{-1} = -0.5 * 0.1 = -0.05$
- Tilsvarende kan vi skrive binære flyttall på formen
  mantisse * $2^{eksponent}$
- For flyttall må vi altså representere både eksponent og mantisse. Begge må kunne være positive, negative og null.

Binære flyttall: IEEE 754 single precision



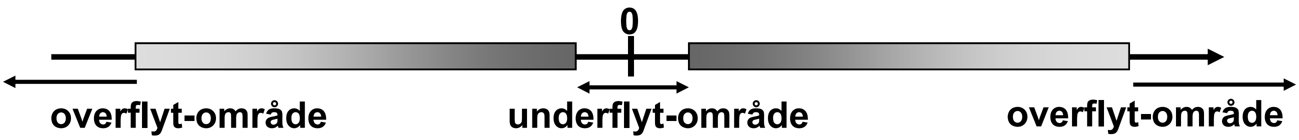
Binære flyttall: IEEE 754 double precision



IEEE 754: Spesielle verdier

- **Null**: Både eksponent og mantisse er 0
- **Uendelig**: Eksponent med bare 1ere, mantisse med bare 0ere
- **Not A Number**: Eksponent med bare 1ere, mantisse ≠ 0
  - Mantisse som starter med 1 : Resultat av en udefinert operasjon (eksempel: 0/0)
  - Mantisse som starter med 0: Resultat av en ulovlig operasjon (eksempel: N/0)

Flyttallsområder i IEEE 754 (og i Java)

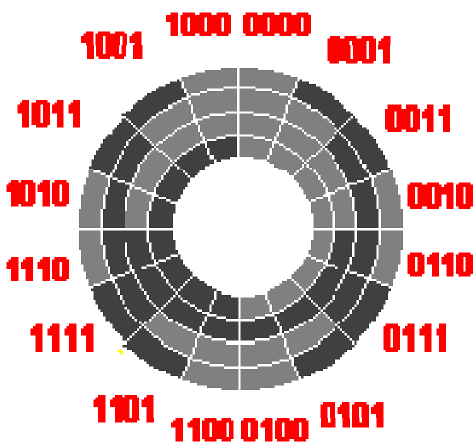


datatype	antall biter	minste positive tall	største positive tall
float	32	$2^{-126} \approx 10^{-44,85}$	$(2 - 2^{-23}) * 2^{127} \approx 10^{38,53}$
double	64	$2^{-1022} \approx 10^{-323,3}$	$(2 - 2^{-52}) * 2^{1023} \approx 10^{308.3}$

Gray-kode

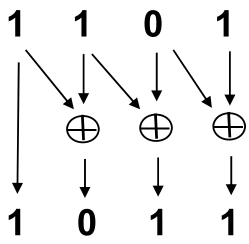
Gray-kode	Binært tallsystem	Titallsystem
0000	0000	0
0001	0001	1
0011	0010	2
0010	0011	3
0110	0100	4
0111	0101	5
0101	0110	6
0100	0111	7
1100	1000	8
1101	1001	9
1111	1010	10
1110	1011	11
1010	1100	12
1011	1101	13
1001	1110	14
1000	1111	15

Et ikke-posisjonssystem der representasjonen av et tall og det neste tallet i tallrekken atskiller seg i bare én bit

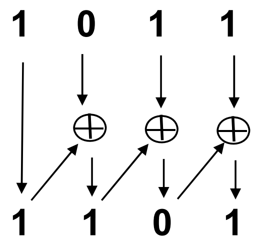


Konvertering binært tallsystem ↔ Gray-kode

Fra det binære tallsystemet til Gray-kode



Fra Gray-kode til det binære tallsystemet

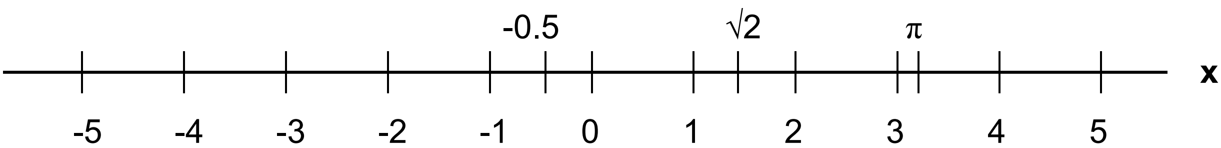


⊕ : "exclusive or"-operasjonen
to like biter gir 0, to ulike biter gir 1

Huskeregel:
I begge konverteringer XOR'es med forrige bit i det binære tallsystemet!

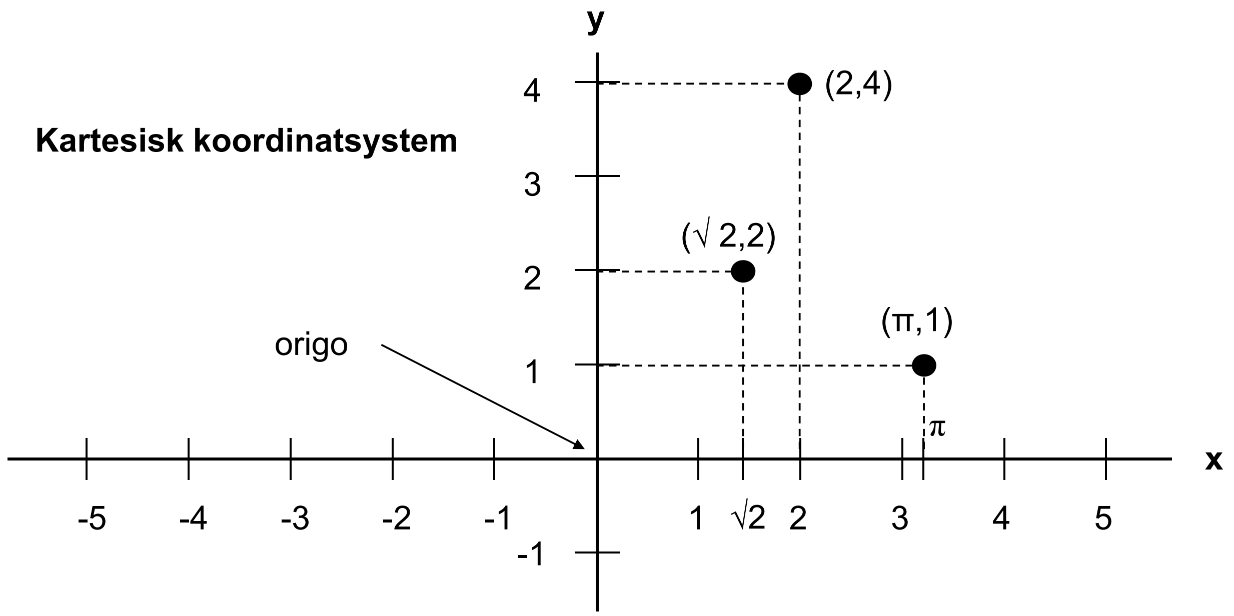
Punkter i endimensjonalt rom

- **Et tall kan oppfattes som et punkt i et endimensjonalt rom.**
- **Tallet er da en koordinatverdi.**
- **Eksempler: Punktene -0.5, $\sqrt{2}$, π**



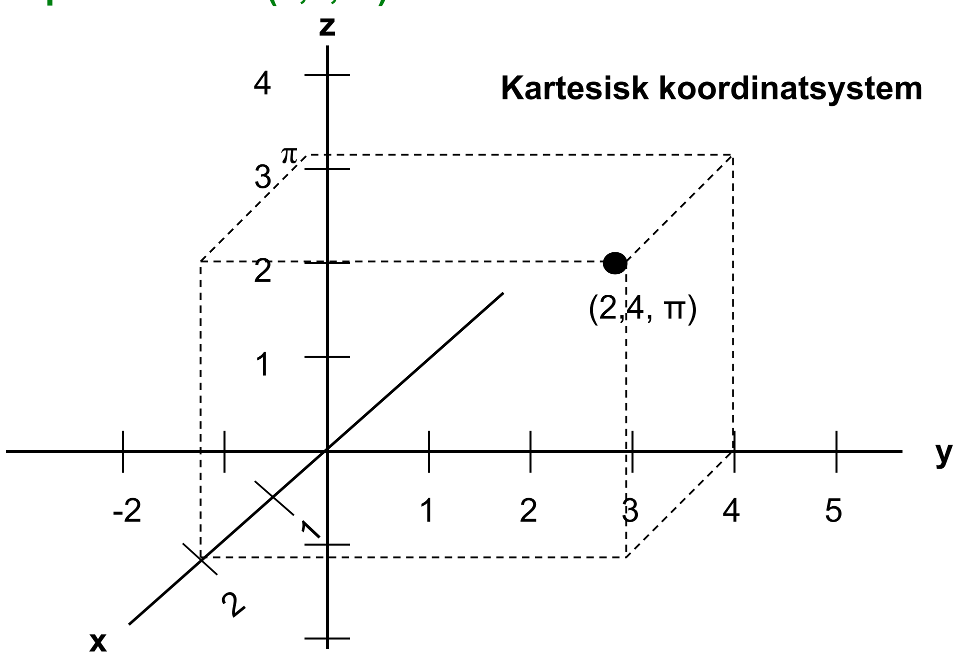
Punkter i todimensjonalt rom

- I et todimensjonalt rom trenger vi et koordinatpar.
- Eksempel: Punktet (√ 2,2)



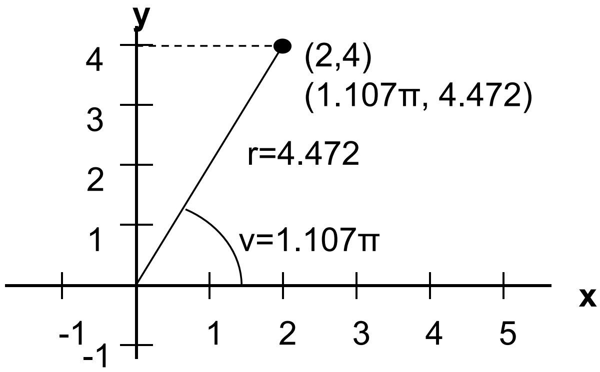
Punkter i tredimensjonalt rom

- I et todimensjonalt rom trenger vi et koordinattrippel.
- Eksempel: Punktet (2,4, π)



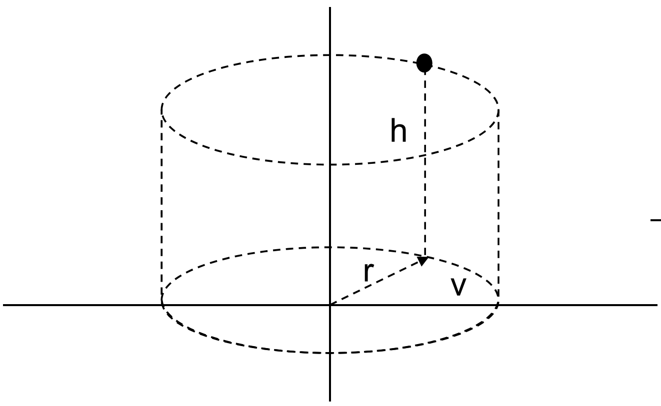
Polarkoordinater

- Alternativ til kartesiske koordinater ved to dimensjoner:

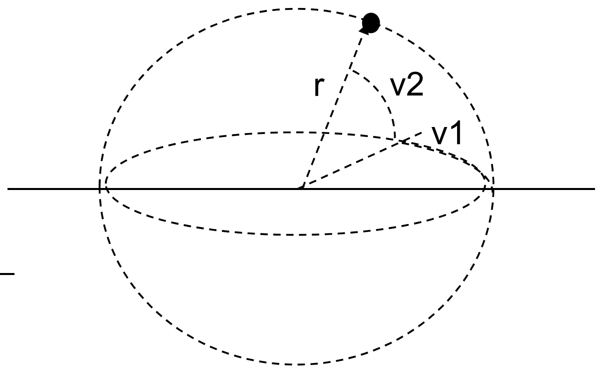


Sylinderkoordinater og sfæriske koordinater

- Alternativ til kartesiske koordinater ved tre dimensjoner:



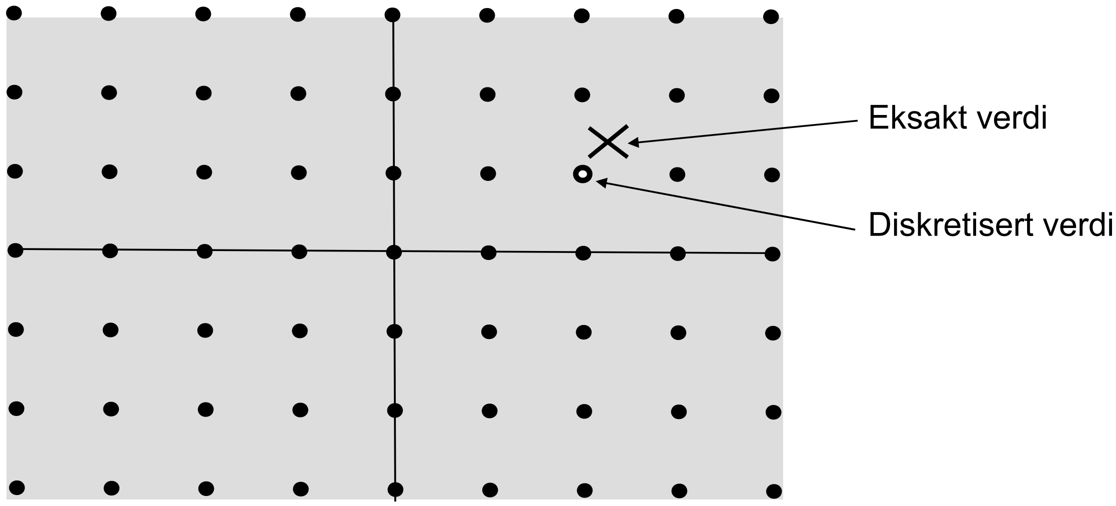
Sylinderkoordinater



Sfæriske koordinater

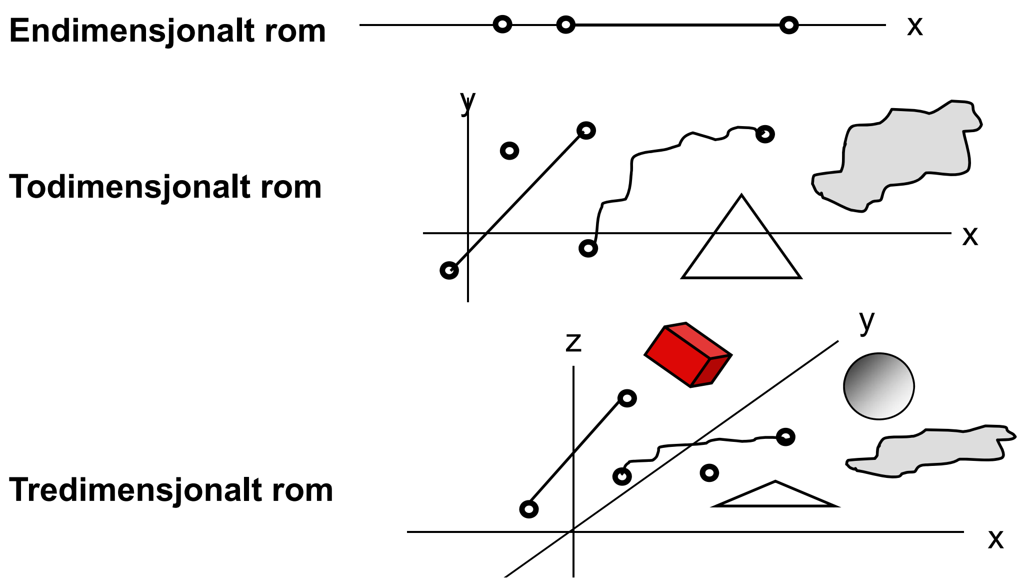
Diskretisering i rommet

- Punkter i flerdimensjonale rom må ”snappes” til nærmeste representerbare punkt – på samme måte som tall (punkter i det endimensjonale rom) ”snappes” til nærmeste representerbare tall



Geometrier i rommet

- En geometri kan oppfattes som en punktsky med uendelig mange punkter. Dimensjonaliteten kan ikke være større enn rommets dimensjonalitet.



Representasjoner av geometri

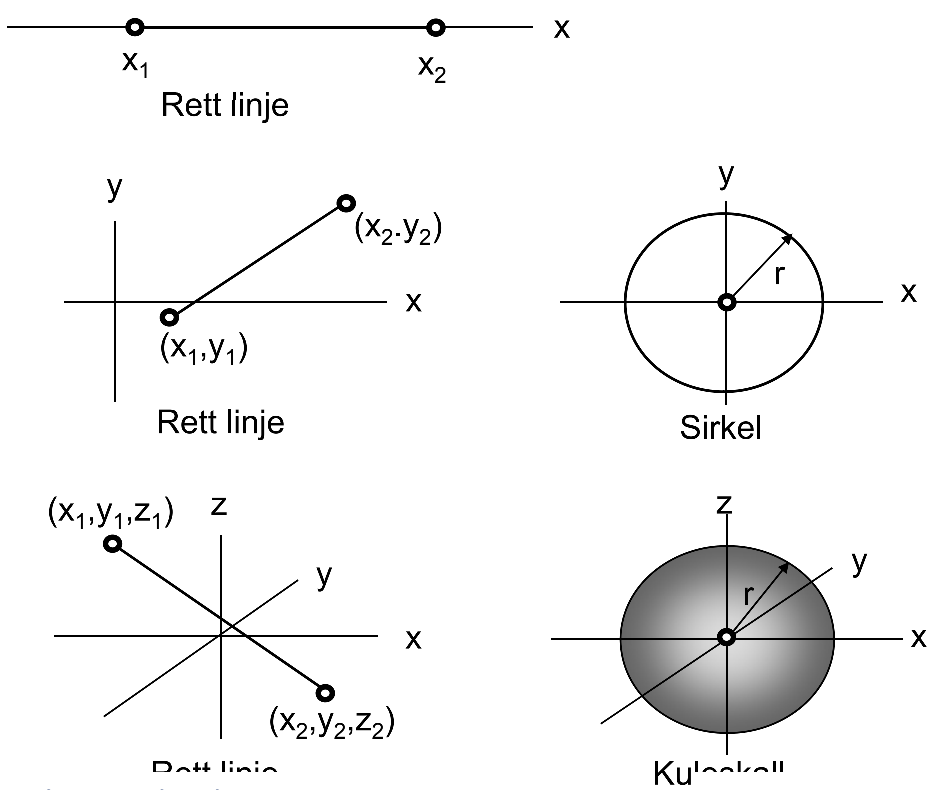
- **"Vektorrepresentasjon":**
  - **Representere noen viktige punkter, og avlede de øvrige punktene matematisk ved behov.**
  - **Egnet for "regulære" geometrier.**
- **"Rasterrepresentasjon"**
  - **Bygge opp representasjonen av et endelig antall "punkter med utstrekning".**
  - **Gir vanligvis bare en tilnærmet korrekt geometri.**

"Regulære" geometrier

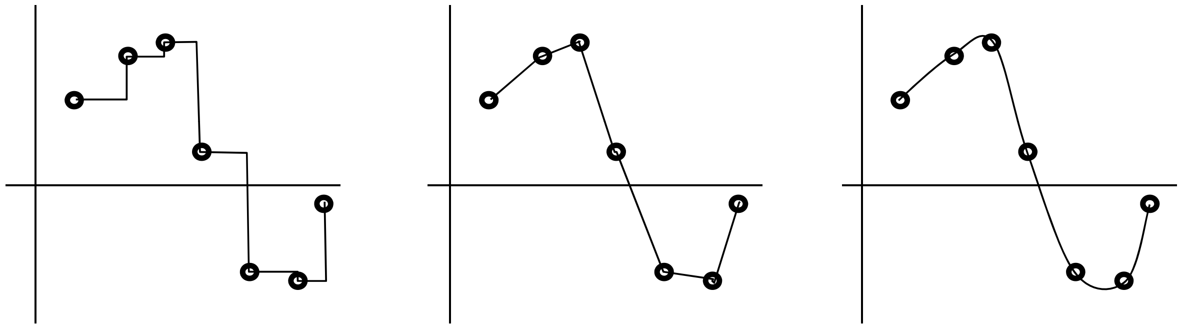
Endimensjonalt rom

Todimensjonalt rom

Tredimensjonalt rom

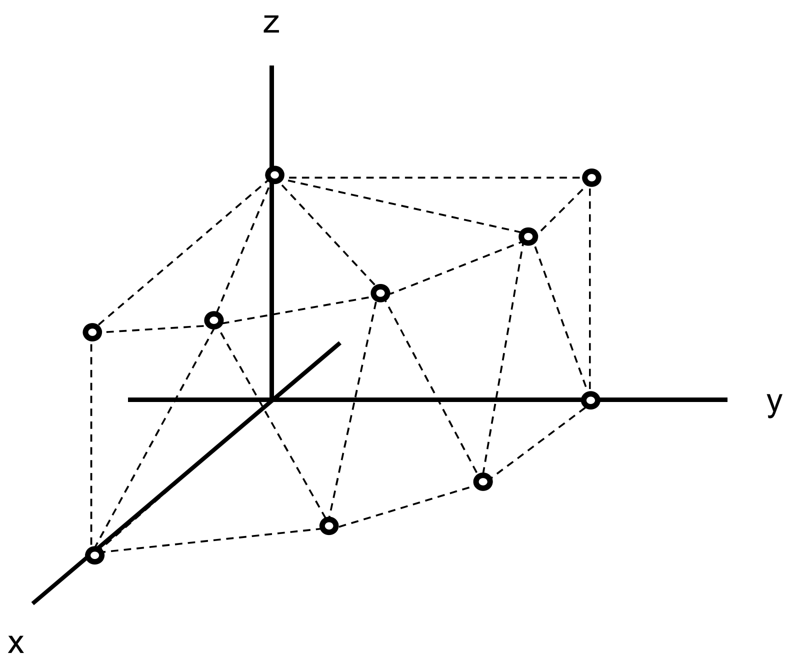


Interpolasjonsteknikker

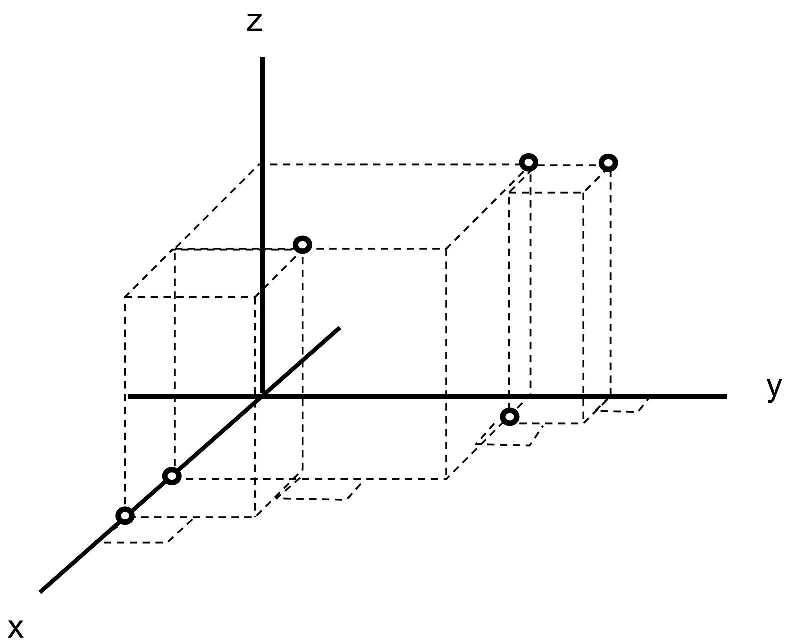


Interpolasjon med konstant   Lineær interpolasjon   Interpolasjon med glatting

Triangulated irregular network (TIN)



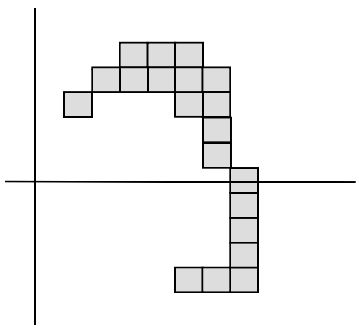
Quad-tre



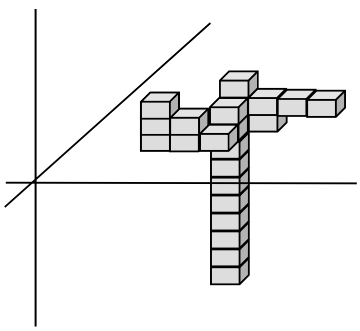
Rasterrepresentasjon

- I rasterrepresentasjon bygges geometrien opp av ”punkter med utstrekning”.

Endimensjonalt raster



Todimensjonalt raster



Tredimensjonalt raster

Tall – oppsummering

- Tall kan representeres tekstlig, som binære tall, i Gray-kode eller som flyttall.
- For negative binære heltall brukes vanligvis toer-komplementer. Andre alternativer er bruk av fortegnsbit eller bruk av bias.
- Tall kan betraktes som punkter i et rom – tallene fungerer da som koordinater.
- Punkter som ikke kan representeres eksakt, må "snappes" til nærmeste representerbare punkt – såkalt diskretisering.
- For geometrier med utstrekning kan vi bruke enten vektorrepresentasjon eller rasterrepresentasjon.

Planen videre

- Vi er nå ferdige med
  - hvordan tegn og tekst representeres og presenteres.
  - hvordan nettsider og stilark konstrueres.
  - hvordan tall og geometrier representeres.
- Neste uke starter Fritz med
  - introduksjon til lyd.