UKE 13

Plan for i dag (siste gruppetime!)

- Pepperkaker
- Eksamensinfo
- Inspera
- Gjennomgang av noen eksamensoppgaver + litt ekstra pensum
  - Så langt vi rekker

Info om eksamen

- Dato og sted: 11. desember, Silurveien 2
- Klokkeslett: 09:00-13:00
- Hjelpemidler:

  Alle trykte og skrevne hjelpemidler er tillatt, samt en kalkulator (kun med batteri, og uten kommunikasjonsmuligheter)

- Viktig: Husk gyldig legitimasjon med bilde
- Info om silurveien: https://www.uio.no/studier/eksamen/silurveien/

HyperText Markup Language (HTML) er et markeringsspråk for formatering av nettsider der man blant mye annet kan vise tekst i forskjellige farger. I HTML representeres farger med en rød (R), grønn (G) og blå (B) verdi - såkalt RGB. Disse er vanligvis angitt i hexadesimal notasjon, der den første byten (fra venstre) er rød, den andre er grønn, og den tredje er blå.

Hvilke verdier for rød, grønn og blå har fargekoden 0xA07CD1?



Velg ett eller flere alternativer

- [ ] rød=148, grønn=124, blå=209
- [ ] rød=148, grønn=128, blå=191
- [ ] rød=160, grønn=134, blå=179
- [ ] rød=160, grønn=124, blå=209

Heksadesimal notasjon

- Base 16
  - Representeres med 16 forskjellige siffer

Desimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Konvertering fra binært til desimalt

$10010111_2 = ?_{10}$

1	0	0	1	0	1	1	1
$1 * 2^7$	$0 * 2^6$	$0 * 2^5$	$1 * 2^4$	$0 * 2^3$	$1 * 2^2$	$1 * 2^1$	$1 * 2^0$
128	0	0	16	0	4	2	1

128 + 16 + 4 + 2 + 1 = 151

$10010111_2 = 151_{10}$

Du har følgende to bytes:

Byte A:

0	1	0	0	1	1	0	0

Byte B:

0	1	0	1	0	0	0	1

Regn ut en ny byte C som er summen av disse (C = A + B). Hvilke verdier kan C ha?

Velg ett eller flere alternativer

- [ ] -99
- [ ] 157
- [ ] -67
- [ ] 126

Hvordan representere negative tall?

- 2-er komplement!
  - Fortegnsbitet tolkes som det negative av verdien den ellers ville hatt
    - De resterende bytene legger vi til denne verdien

1	**0**	**1**	**1**	**0**	**1**	**1**	**0**

Hva er verdien til tallet med 2-er komplement?

- $-2^7 + 2^5 + 2^4 + 2^2 + 2^1 = -128 + 32 + 16 + 4 + 2 =$ **-74**

Hva er verdien til tallet uten 2-er komplement?

- $2^7 + 2^5 + 2^4 + 2^2 + 2^1 = 128 + 32 + 16 + 4 + 2 =$ **182**

1.3 Tallsystemer

Konverter til desimal (10-tall) systemet.

a) 1101_2 Velg alternativ (Ingen av disse, 23, 51, 13, 135)

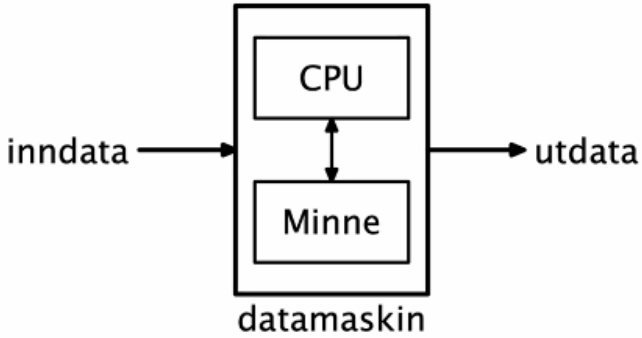
b) 113_4 Velg alternativ (23, 13, Ingen av disse, 135, 51)

c) 0x33 Velg alternativ (51, 23, Ingen av disse, 13, 135)

d) 207_8 Velg alternativ (51, 23, 135, Ingen av disse, 13)

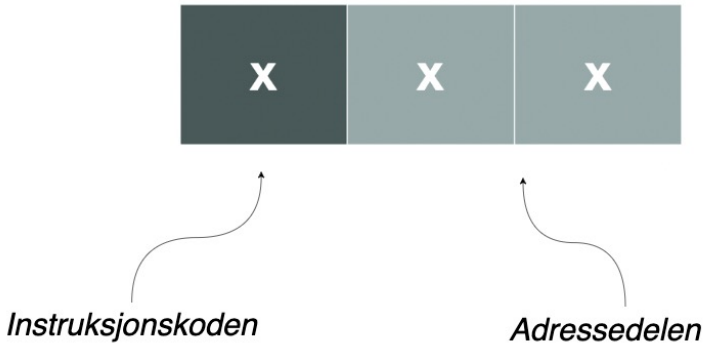
Von Neumann-arkitektur

- CPU
  - Utfører oppgaver
- Programmet
  - Det som bestemmer hva CPU'en skal gjøre
  - Ligger som kode i minnet på maskinen



Fra LMC:

- Instruksjonskoden: Viser hvilken instruksjon kontroll-logikken
  - F.eks. **skriv ut** et tall til akkumulatoren
- Adressedelen: viser hvilken adresse instruksjonen skal bruke
  - F.eks. Lagre tallet som nå ligger i akkumulator **på adresse 23**



LMC - instruksjonskoder

Kode	Navn	Beskrivelse
0xx	HLT	Stopper eksekveringen
1xx	ADD	Adderer verdien i angitt minnelokasjon med akkumulatoren
2xx	SUB	Subtraherer verdien i angitt minnelokasjon med akkumulatoren
3xx	STA	Lagrer akkumulatoren i angitt minnelokasjon
4xx	-	Ikke i bruk
5xx	LDA	Henter verdi fra minnet til akkumulatoren
6xx	BRA	Hopper til angitt adresse
7xx	BRZ	Hopper hvis akkumulatoren er 0
8xx	BRP	Hopper hvis akkumulatoren er ≥ 0
901	INP	Leser verdi fra input, og legger svaret i akkumulatoren
902	OUT	Skriver ut verdien i akkumulatoren
922	OTC	Skriver ut ASCII-tegn (ikke i boka)

ASCII

Første vellykkede forsøk på standardisering var ASCII (American Standard Code for Information Interchange) i 1963; den har 128 tegn:

ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

```
        INP
        STA a
        INP
        BRZ print
        LDA a
        SUB b
        STA a
print   LDA a
        OTC
        HLT
a       DAT
b       DAT 32
```

Hva skrives ut når denne koden kjøres og bruker gir følgende inndata: **114** og **999**?

Velg ett eller flere alternativer

- [ ] r
- [ ] T
- [ ] 999
- [ ] q
- [ ] R

```
        INP
        STA a
        INP
        STA b
loop    LDA a
        INSTRUKSJON 1
        INSTRUKSJON 2
        INSTRUKSJON 3
        INSTRUKSJON 4
        LDA a
        SUB en
        STA a
        BRA loop
slutt   LDA res
        OUT
        HLT
res     DAT 0
a       DAT
b       DAT
en      DAT 1
```

Du lager et lite program for å gange sammen to tall **a** og **b**, slik at

res = a * b

Du har skrevet programmet over, men det mangler noen instruksjoner. Hvilke?

Instruksjon 1:

Velg alternativ (BRA slutt, BRZ slutt, BRP slutt)

Instruksjon 2:

Velg alternativ (STA a, LDA a, LDA res, LDA b)

Instruksjon 3:

Velg alternativ (ADD res, ADD a, ADD en, ADD b)

Instruksjon 4:

Velg alternativ (STA a, Ingen instruksjon, STA b, STA res)

```
start   INP                 00 INP
        BRZ slutt           01 BRZ 13
        STA in              02 STA 27
        LDA laster          03 LDA 28
        ADD in              04 ADD 27
        STA loop            05 STA 06
loop    HLT                 06 HLT
        BRZ start           07 BRZ 00
        OTC                 08 OTC
        LDA loop            09 LDA 06
        ADD en              10 ADD 26
        STA loop            11 STA 06
        BRA loop            12 BRA 06
slutt   HLT                 13 HLT
lmc     DAT 76 // L         14 DAT 76
        DAT 77 // M         15 DAT 77
        DAT 67 // C         16 DAT 67
        DAT 0 // NULL       17 DAT 00
jul     DAT 74 // J         18 DAT 74
        DAT 117 // u        19 DAT 117
        DAT 108 // l        20 DAT 108
        DAT 0 // NULL       21 DAT 00
ifi     DAT 73 // I         22 DAT 73
        DAT 70 // F         23 DAT 70
        DAT 73 // I         24 DAT 73
        DAT 0 // NULL       25 DAT 00
en      DAT 1               26 DAT 01
in      DAT                 27 DAT 00
laster  LDA 0               28 LDA 00
```

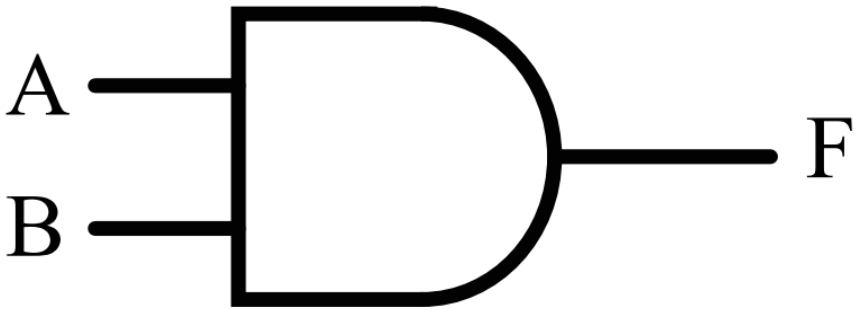
Koden over skriver ut én og én karakter fra ønsket adresse inntil en "null" leses fra minnet.

Etter å ha tastet inn "14", taster brukeren inn "18" og så "0". Hvilken tekst skrives ut?

Velg alternativ ("IFI" og ikke noe mer., Programmet kræsjer., "IFI", "Jul", og ikke noe mer., "LMC", "Jul", og ikke noe mer.)

Hvilken maskinkode finner vi på etikett **loop** etter at den sjette instruksjonen har blitt eksekvert?

Velg alternativ (906, 500, 518, 914, 514, 300)



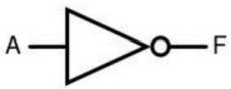
Hvilke(n) port er avbildet over:
Velg ett eller flere alternativer

- [ ] NAND-port
- [ ] XNOR-port
- [ ] NOR-port
- [ ] AND-port
- [ ] XOR-port
- [ ] OR-port
- [ ] NOT-port

INV - port

$F = A'$

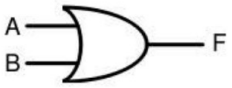
A	F
0	1
1	0



OR - port

$F = A+B$

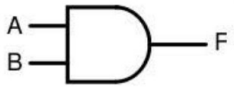
A	B	F
0	0	0
0	1	1
1	0	1
1	1	1



AND - port

$F = AB$

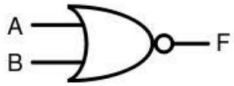
A	B	F
0	0	0
0	1	0
1	0	0
1	1	1



NOR - port

$F = (A + B)'$

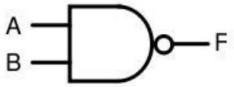
A	B	F
0	0	1
0	1	0
1	0	0
1	1	0



NAND - port

$F = (AB)'$

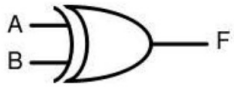
A	B	F
0	0	1
0	1	1
1	0	1
1	1	0



XOR - port

$F = A \oplus B$

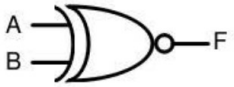
A	B	F
0	0	0
0	1	1
1	0	1
1	1	0



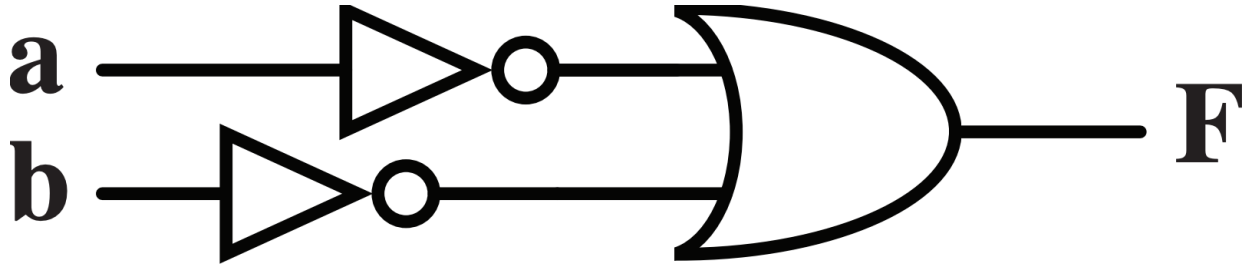
XNOR - port

$F = (A \oplus B)'$

A	B	F
0	0	1
0	1	0
1	0	0
1	1	1



Alle logiske port-typer



Funksjonsuttrykket for kretsen over er:

Velg ett eller flere alternativer:

- $F = \overline{a} + \overline{b}$
- $F = \overline{a} \cdot \overline{b}$
- $F = \overline{(ab)}$
- $F = \overline{a + b}$

Boolean algebra simplification rules

1.	Law of Identity	$A = A$ $\overline{A} = \overline{A}$
2.	Commutative Law	$A \cdot B = B \cdot A$ $A + B = B + A$
3.	Associative Law	$A \cdot (B \cdot C) = A \cdot B \cdot C$ $A + (B + C) = A + B + C$
4.	Idempotent Law	$A \cdot A = A$ $A + A = A$
5.	Double Negative Law	$\overline{\overline{A}} = A$
6.	Complementary Law	$A \cdot \overline{A} = 0$ $A + \overline{A} = 1$
7.	Law of Intersection	$A \cdot 1 = A$ $A \cdot 0 = 0$
8.	Law of Union	$A + 1 = 1$ $A + 0 = A$
9.	DeMorgan's Theorem	$\overline{AB} = \overline{A} + \overline{B}$ $\overline{A + B} = \overline{A}\,\overline{B}$
10.	Distributive Law	$A \cdot (B + C) = (A \cdot B) + (A \cdot C)$ $A + (BC) = (A + B) \cdot (A + C)$
11.	Law of Absorption	$A \cdot (A + B) = A$ $A + (AB) = A$
12.	Law of Common Identities	$A \cdot (\overline{A} + B) = AB$ $A + (\overline{A}B) = A + B$

Anta at det er igjen 3000 instruksjoner og at en instruksjon tar 1 klokkesykel, bortsett fra eventuelle cache-miss. Videre kan du anta at det vil være 50% cache-miss hvor det vil ta totalt 4 klokkesykel for hver instruksjon i cache-miss.

Hvor lang til vil det totalt ta?
Velg ett eller flere alternativer

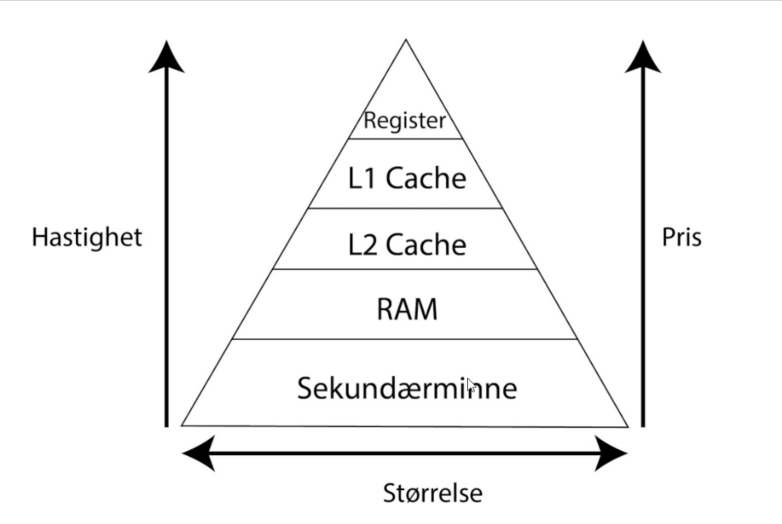
- [ ] 10000
- [ ] 8250
- [ ] 9000
- [ ] 5500
- [ ] 3000
- [ ] 13250
- [ ] 5000
- [ ] 7500
- [ ] 6600
- [ ] 15000

Klokkefrekvens

- Frekvens: Antall klokkesykler per sekund
  - Hver gang klokken ‘slår’, henter CPUen en instruksjon og utfører den
  - Oppgis ofte i gigahertz (GHz)
- Jo større frekvens, jo fortere kan CPUen prosessere instruksjoner

$$frekvens = \frac{1}{klokkeperioden}$$

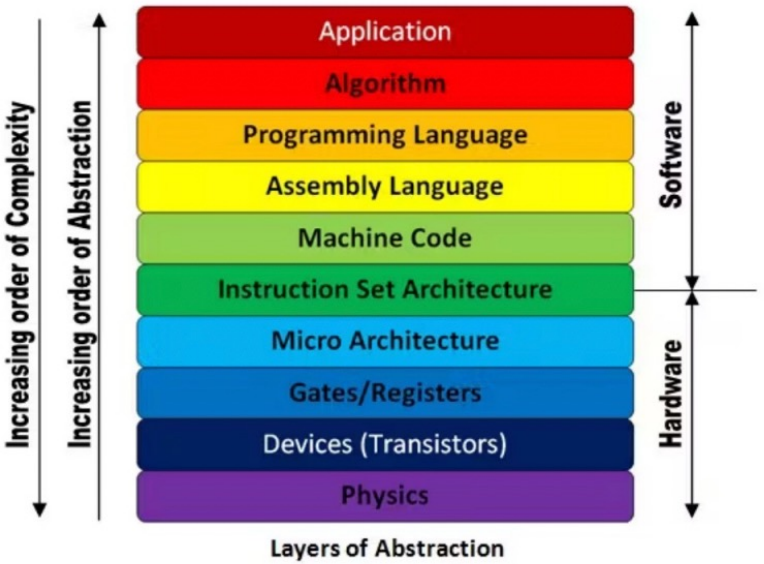
Minnehierarki



- All data er lagret, og vi har forskjellige minneenheter som kan brukes til å lagre dem
- De som er raskest å aksessere er de som koster mest, men også de som har minst lagringskapasitet
- Hva vil det si at et minneelement har høy hastighet?
    - At det tar kort tid å kommunisere med CPUen

Abstraksjonsnivåer for datamaskiner

Abstraksjon: forenkling



Abstraksjonsnivåer for datamaskin

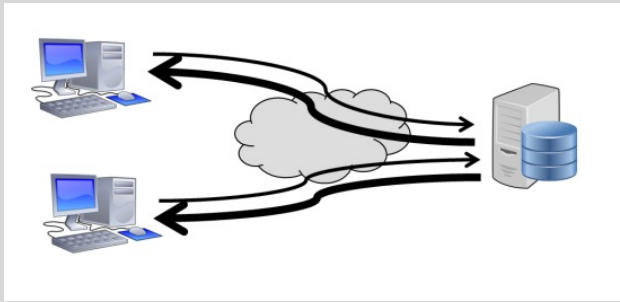
Hva kjennetegner aksessmodellen klient-tjener?
Velg ett eller flere alternativer

- [ ] En tjener lytter etter henvendelser og leverer tjenester på forespørsel.
- [ ] En klient initierer utvekslingen ved å koble til en tjener og spørre om å få utført en tjeneste.
- [ ] Nodene kan være både klint og tjener samtidig.
- [ ] Mange uavhengige noder samarbeider om å tilby en tjeneste.
- [ ] Det finnes ingen sentralisert kontroll over tjenesten.

Aksessmodell: Klient-tjener

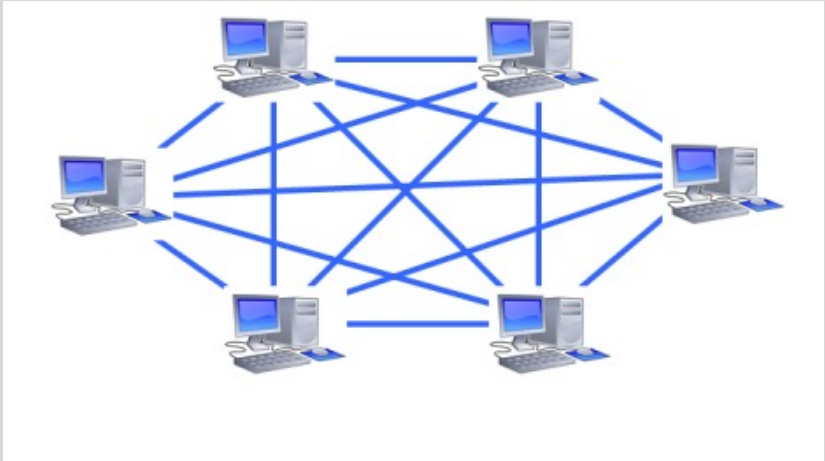
- Aksessmodellen som brukes mest i internett i dag
- Tjeneren lytter etter henvendelser og leverer tjenester på forespørsel
- Klienten initierer utvekslingen ved å koble til en tjener og spørre om å få utført en tjeneste
  - Klienten oppretter forbindelsen

Eksempel: En nettleser (klient) som sender en forespørsel til YouTube sin web-server (tjener)



Aksessmodell: Peer-to-peer (P2P)

- Alle noder i nettet er likeverdige
  - En node/datamaskin kan være både klient og tjener
- Alle noder i nettverket er likestilte og sammenkoblet
  - Alle skal i teorien kunne nå alle
- Distribuert eierskap
  - Data kan aksesseres fra likeverdige noder
    (og ikke fra en server)



Hva er sant for et linjeswitchet nettverk?

Velg ett eller flere alternativer

- [ ] Det må alltid reserveres kapasitet langs hele stien.
- [ ] Data for sending deles opp i mindre deler som sendes uavhengig på nettet.
- [ ] Forskjellige pakker kan ta forskjellige stier fra avsender til mottaker.
- [ ] Det settes opp en dedikert forbindelse mellom sender og mottaker.

Kommunikasjonsformer

- Circuit switching (linjeswitching)
  - Dedikert linje mellom avsender og mottaker
  - Tilkoblingsorientert tjeneste (3 faser)
    - Tilkobling
    - Samtale
    - Nedkobling
- Ikke mye brukt i dag
  - Brukes til å reservere ressurser

- Packet switching (pakkenettverk)
  - Brukes på internett i dag
  - Ingen dedikert linje mellom sender og mottaker.
  - Tilkoblingsløs tjeneste
    - Data sendes videre, og det er neste intermediate-system (router/switch) sitt ansvar å finne veien videre

Du ønsker å laste ned en fil på 150 megabyte, og hastigheten på din Internettlinje er 50 megabit per sekund ned og 20 megabit per sekund opp. Hva er den teoretisk korteste overføringstiden?

Velg ett alternativ:

- 7,5 sekunder
- 60 sekunder
- 24 sekunder
- 150 sekunder
- 3 sekunder

En datamaskin har IP-adressen 172.16.100.18
Nettverksmasken er: 255.255.255.248
Hva er IP-adressen til maskinen i CIDR-notasjon?
Velg ett alternativ:

- 172.16.100.1/26
- 172.16.100.1/29
- 172.16.100.18/26
- 172.16.100.18/29

CIDR og punktnotasjon

- To måter å oppgi en IP-adresse på:
  - **Punktnotasjon**
    - F.eks. 192.168.1.0
      - Må da oppgi nettverksmasken
        - 255.255.255.0
  - **CIDR-notasjon**
    - Punktnotasjon + antall bits nettverksmasken består av
      - 192.168.1.0/24
    - Hvordan ser man antall bits nettverksmasken består av?
      - 255.255.255.0 = 11111111.11111111.11111111.00000000
        - Antall enere er 24

Regne ut subnett fra IP + nettverksmaske

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

AND

For å regne ut subnettet til en maskin med en gitt IP-adresse, så må vi ta en bitvis AND-operasjon mellom IP-adressen og nettverksmasken

Regne ut kringkastningsadresse fra IP + nettverksmaske

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

OR

Om man skal sende ut en melding til alle på subnettet, må man bruke Kringkastningsadressen. Denne regnes ut ved å ved å gjøre en bitvis OR-operasjon mellom IP-adressen og inversen av nettverksmasken.

Hva er hovedmotivasjonen ved å bytte ut IPv4 med IPv6?

Velg ett alternativ

- Lettere å koble sammen IP-adresser og MAC-adresser
- Det blir flere tilgjengelige porter per IP-adresse
- Øke antallet globalt adresserbare IP-adresser
- Gjøre det vanskeligere å utføre "man-in-the-middle" angrep.

Hvilke utsagn stemmer om streaming av video over HTTP?
Velg ett eller flere alternativer

- [ ] Video deles opp i små segmenter og forskjellige kvaliteter.
- [ ] Streaming over HTTP bruker kun UDP til overføring.
- [ ] Med HTTP-streaming må man bufre hele videoen før avspilling kan starte.
- [ ] Det er klienten som bestemmer kvaliteten som lastes ned, ikke tjeneren.

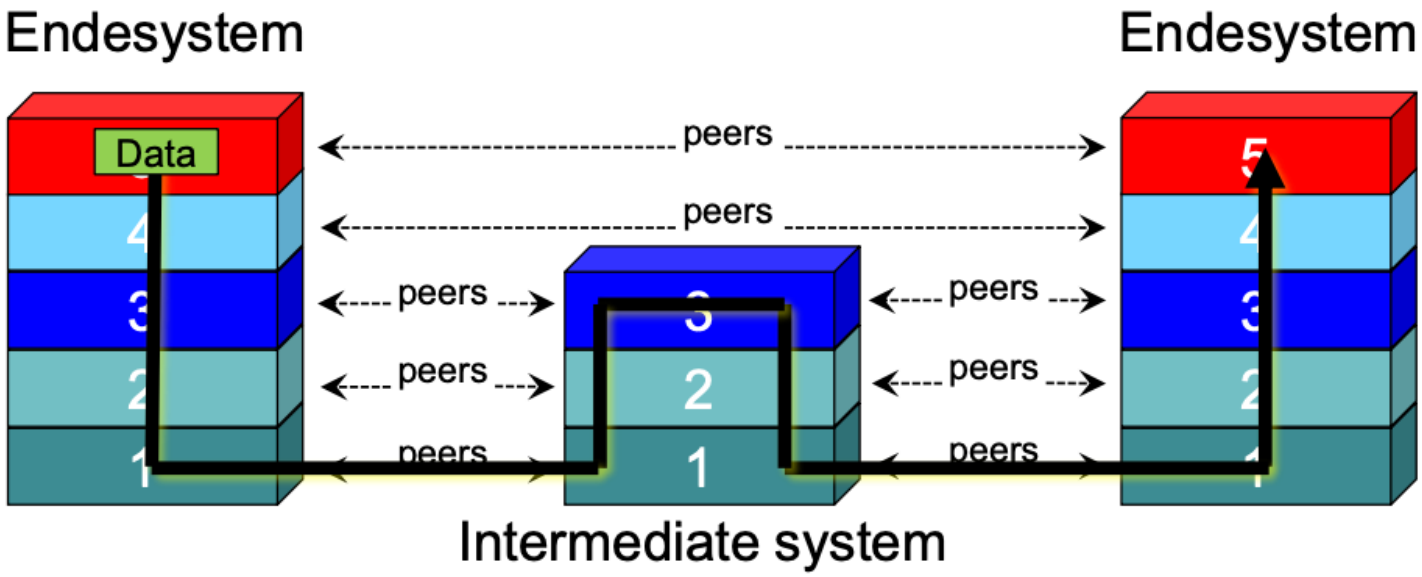
Videostreaming

- Over 50% av trafikken på nettet er videostreaming
- Før måtte man laste ned hele videofiler
  - Treigt
- Derfor har vi videostreaming
  - Gjør at vi kan laste ned videoer fortløpende, mens vi ser på dem
  - Sparer nettverksressurser, fordi man slipper å sende mye data på en gang
- Vi bruker både TCP og UDP til å streame videoer.
- DASH – vanligst form for streaming (HTTP-streaming)
  - Deler videoen i små segmenter
  - Får flere versjoner av samme video, med forskjellige kvalitet
    - Klienten bestemmer hvilken kvalitet som skal brukes basert på nettverksforholdene

Sikkerhetsmål er et sentralt begrep innen informasjonssikkerhet. Hvilke av følgende defineres som sikkerhetsmål:

Velg ett eller flere alternativer

- [ ] Brannmur
- [ ] Orm
- [ ] Tilgangskontroll
- [ ] Tilgjengelighet
- [ ] Uavviselighet
- [ ] Tofaktor-autentisering
- [ ] Integritet
- [ ] Autorisering


Endesystem
Endesystem
Data
peers
peers
peers
peers
peers
peers
peers
peers
5
4
3
2
1
3
2
1
Intermediate system

Sikkerhetsmål

Sikkerhetsmål er egenskaper ved informasjonssikkerhet som vi ønsker å oppnå eller tilby. De opprinnelige sikkerhetsmålene er konfidensialitet, integritet og tilgjengelighet (KIT-målene)

- **Konfidensialitet**
  - Å sikre at ingen andre enn de med rettmessig behov har tilgang til informasjon
- **Integritet**
  - Å sikre at informasjon alltid er korrekt og ikke endres eller slettes utilsiktet
- **Tilgjengelighet**
  - Å sikre at en informasjon alltid er tilgjengelig for rett person til rett tid

Sikkerhetsmål

- **Autentisitet**
  - Hvor sikre vi er på at noe eller noen er det/den de utgir seg for å være (krevende digitalt)
- **Uavviselighet**
  - Å hindre mulighet for fornektelse av at data er sendt eller mottatt, også ovenfor en tredjepart
- **Sporbarhet**
  - Å kunne knytte en gitt identitet til en gitt hendelse
- **Personvern**
  - Retten til privatliv og retten til å bestemme over personopplysninger

Typer skadevare

 Virus	Kjennetegnes ved at det sprer seg hver gang en fil med virus blir åpnet av bruker eller systemet.
 Dataorm	Skadevare som på egenhånd er i stand til å spre seg og finne nye ofre.
 Trojaner	Skadevare som utgir seg for å være et nyttig program.
Bakdør	En ubeskyttet og ukjent åpning inn i et datasystem, laget av en angriper.
 Spionvare	Programvare for å spionere på en bruker i håp om å få tak i f.eks. informasjon, passord, kontoinformasjon.
 Tastelogger (Keylogger)	En enhet eller programvare som er koblet til tastatur eller maskin og som lagrer eller videresender alle tastetrykk. En variant av spionvare.
Rootkit	Skadevare som endrer eller erstatter funksjoner i OS eller applikasjoner, med mål å ta kontroll over en maskin eller et system.
Exploit	Et program, en sekvens med data, eller en oppskrift på hvordan en sårbarhet kan utnyttes.

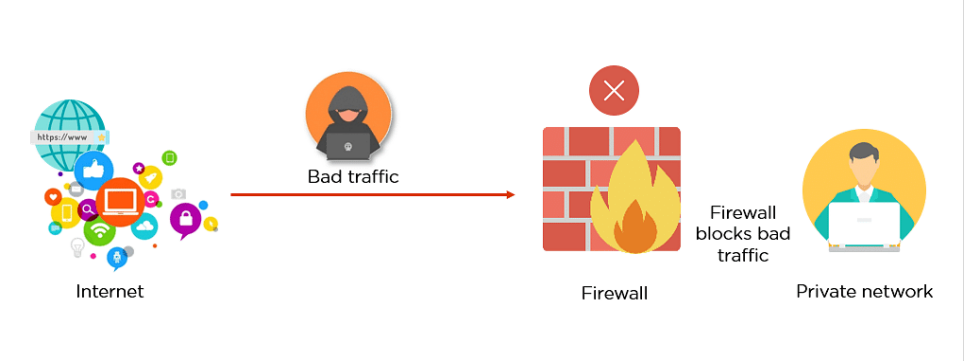
Konfidensialitet er et sentralt krav i norsk personvernlovgivning. Hvilke av følgende sikkerhetstiltak kan sørge for å ivareta konfidensialitet for persondata som behandles i et IT-system:

Velg ett eller flere alternativer

- [ ] Bruk av kryptografi for data lagret i systemet
- [ ] Sørge for redundante tjenester
- [ ] Bruk av skallforsvar
- [ ] Sørge for sikkerhetskopi av alle data
- [ ] Sørge for sikkerhetskopi av all maskinvare
- [ ] Identifisere alle brukere av IT-systemet

Nettverkssikkerhet: Skallforsvar

- **Brannmur**
  - Som en dørvakt
  - Slipper inn eller avviser trafikk inn/ut fra et nettverk
  - Følger et regelsett når den skal avgjøre om noe skal slippe inn/ut
- **Innbruddsdeteksjon**
  - Overvåker nettverkstrafikk for å detektere innbrudd, virus og tjenestenektangrep
  - Vil kunne avdekke mistenkelig aktivitet i nettverkstrafikken som brannmuren tillater



Integritet er et sentralt krav i norsk personvernlovgivning. Hvilke av følgende sikkerhetstiltak kan sørge for å ivareta integriteten for persondata som behandles i et IT-system:

Velg et eller flere alternativer

- [ ] Holde all programvare godt sikkerhetsoppdatert
- [ ] Identifisere alle brukere av IT-systemet
- [ ] Sørge for redundante tjenester
- [ ] Kryptere all nettverkstrafikk til, fra og internt i datasystemet.
- [ ] Sørge for sikkerhetskopi av maskinvaren
- [ ] Ha gode rutiner for tilgangskontroll i IT-systemet

Hvordan kan tofaktor-autentisering med en kombinasjon av autentiseringsfaktorene *Noe man vet* og *Noe man har* for innlogging bidra til å forbedre sikkerheten i et IT-system?

Velg ett eller flere alternativer

- [ ] Det begrenser hva brukerne har lov til å gjøre i et IT-system.
- [ ] Det gjør det vanskelig å lykkes med såkalte brute-force angrep.
- [ ] Det gjør det unødvendig å ytterligere sikre data lagret i IT-systemet.
- [ ] Det gjør det vanskeligere for angripere å utnytte brukerinformasjon lekket i f.eks. phishing-angrep.

Virksomheten du jobber i har behov for sterk sikkerhet og autentisitet til data og meldinger, og har innført en egen, lokal *offentlig-nøkkel infrastruktur* (PKI). Alle ansatte har fått tilordnet et kryptografisk nøkkelpar, bestående av en privat og en offentlig nøkkel. Siden du har tatt emnet IN1020, blir det din oppgave å forklare for kollegaene dine hvilke nøkler som skal benyttes til hvilke operasjoner.

For hver operasjon nedenfor angi riktig nøkkel:

Nøkkel sender benytter til *signering* (digital signatur): Velg alternativ (Senders private nøkkel, Mottagers private nøkkel, Mottagers offentlige nøkkel, Senders offentlige nøkkel).

Nøkkel mottager benytter for *validering (*digital signatur): Velg alternativ (Mottagers offentlige nøkkel, Senders offentlige nøkkel, Mottagers private nøkkel, Senders private nøkkel)

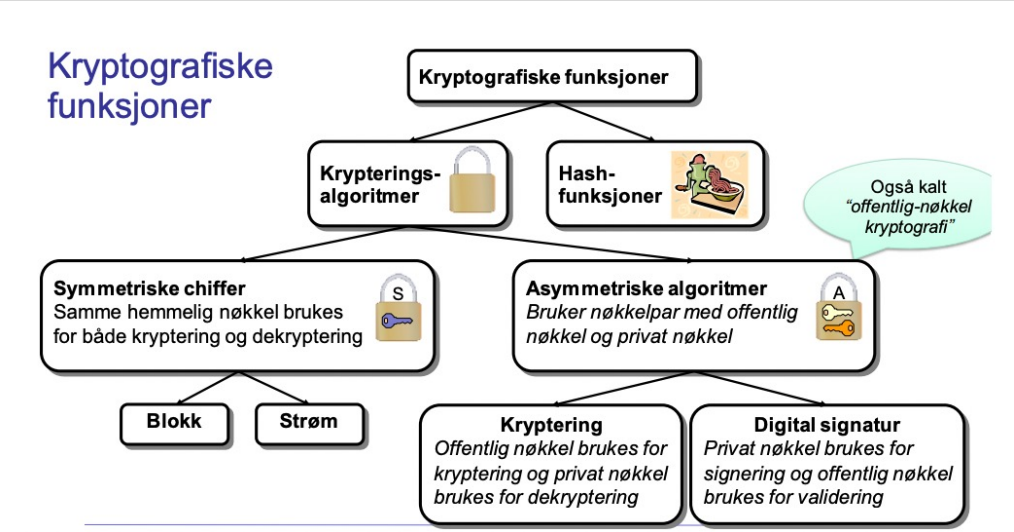
Nøkkel sender benytter for *kryptering* (utveksling av hemmelig melding): Velg alternativ (Mottagers private nøkkel, Senders offentlige nøkkel, Senders private nøkkel, Mottagers offentlige nøkkel)

Nøkkel mottager benytter for *dekryptering* (utveksling av hemmelig melding): Velg alternativ (Mottagers private nøkkel, Senders private nøkkel, Senders offentlige nøkkel, Mottagers offentlige nøkkel)

Sertifikater

- Brukes for å være sikker på at en offentlig nøkkel er ekte
- Består av den offentlige nøkkelen og attributter knyttet til nøkkelen
  - Hvem som eier den, når den er gyldig, utsteder, krypteringsalgoritme, størrelse på nøkkel
- Sertifikater må utstedes av noen vi har tillit til, slik at vi kan stole på at de er ekte
  - Utstedes CA – sertifikat-myndighet
    - Garanterer at nøkler tilhører identiteten som de utgir seg for å være ved å signere sertifikatet. Da har vi et gyldig sertifikat
- **Offentlig-nøkkel infrastruktur (PKI)**
  - Et rammeverk for å binde en offentlig nøkkel til en navngitt enhet
    - Bindingen kan bekreftes av en betrodd myndighet
  - Formål: å garantere autentisitet av offentlige nøkler , og sørge for trygg og forsvarlig nøkkel-distribusjon

Kryptering



Krypteringsnøkkel: tilfeldig streng med tegn

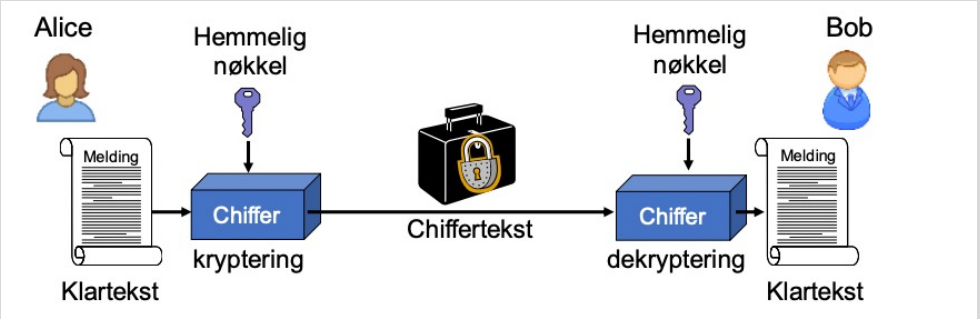
- Generert spesifikt for kryptering/dekryptering (med en krypteringsalgoritme)

Typer kryptering: Symmetrisk

- Samme hemmelige nøkkel brukes for både kryptering og dekryptering.
  - Denne nøkkelen deles mellom sender og mottaker
- Plaintext krypteres til en ciphertext.
- Bruker samme nøkkel til å dekryptere

To typer

- Block cipher (mest brukt)
  - Krypterer blokker av data om gangen
- Stream cipher
  - Krypterer ett og ett bit

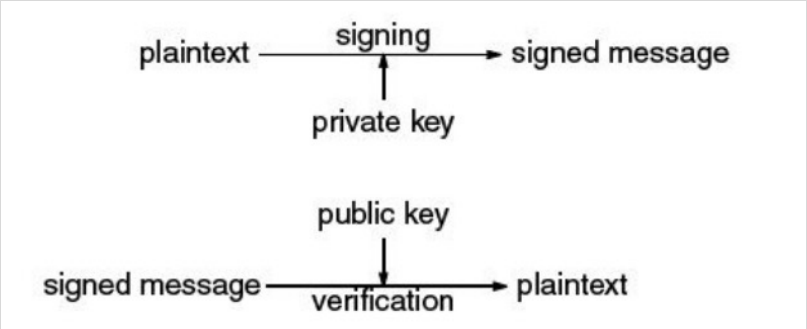


Hvilke sikkerhetsmål kan ivaretas?

- Konfidensialitet?
  - JA
- Integritet
  - JA
- Autentisitet til dataopprinnelse?
  - JA
- Uavviselighet?
  - NEI

Typer kryptering: Digital signatur

- Asymmetrisk kryptering kan brukes i revers for å lage en digital signatur (for å ivareta at meldinger kommer fra en hevdet avsender)
- Man signerer en beskjed med sin private nøkkel, og så kan hvem som helst verifisere at det er signert av deg som person ved å bruke den offentlige nøkkelen din.
- Senderens private nøkkel benyttes til kryptering ved signeringen.
- Senderens offentlige nøkkel benyttes når mottager skal dekryptere medlingen.

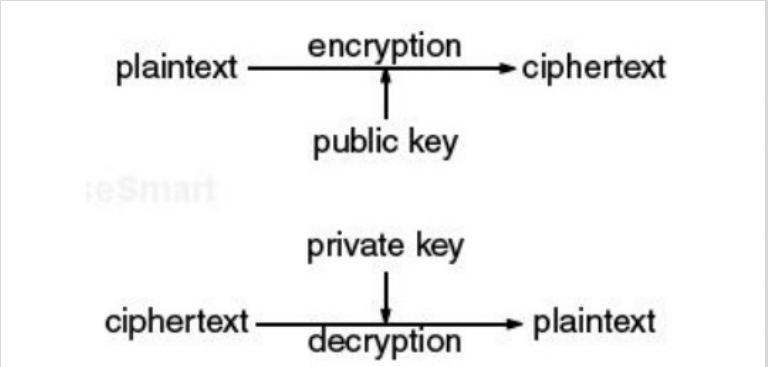


Hvilke sikkerhetsmål kan ivaretas?

- Konfidensialitet?
  - NEI
- Integritet?
  - JA
- Autentisitet til dataopprinnelse?
  - JA
- Uavviselighet?
  - JA

Typer kryptering: Asymmetrisk kryptering

- Benytter seg av et nøkkelpar som består av en privat og en offentlig nøkkel.
- Sender benytter mottagers offentlige nøkkel for å kryptere en melding. Mottager benytter sin private nøkkel for å dekryptere meldingen. Dette anvendes til trygg overføring av data/meldinger.
- Den private nøkkelen er det eneste som kan brukes til å dekryptere en melding som er kryptert med en offentlig nøkkel
- Asymmetriske krypteringsalgoritmer er mye tregere enn symmetriske
  - Derfor har man en hybrid modell
    - Asymmetrisk kryptering brukes for å bli enige om en midlertidig nøkkel
    - Symmetrisk kryptering brukes under resten av kommunikasjonen
    - Det er dette som brukes i HTTPS (HTTP med SSL), som bidrar til kryptert nettverkskommunikasjon

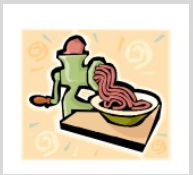


Hvilke sikkerhetsmål kan ivaretas?

- Konfidensialitet?
  - JA
- Integritet?
  - JA
- Autentisitet til dataopprinnelse?
  - NEI
- Uavviselighet?
  - NEI
- Mottagers autentisitet?
  - JA

Typer kryptering: Hash-algoritmer

- Brukes for å teste dataintegriteten
  - Brukes ofte ved nedlasting av programvare
- Kriterier for en god sjekksumalgoritme/hashalgoritme:
  - Enkelt å regne ut sjekksum for en gitt beskjed
  - Ikke mulig å finne en beskjed for en gitt sjekksum
  - Ikke mulig å endre en beskjed uten at sjekksummen blir endret
  - Ikke mulig å finne to forskjellige beskjeder med samme sjekksum



Hashing



Hashing med salting

