

i Eksamensinformasjon

Fakultet: Teknologi, kunst og design

Utdanning: Teknologiske Fag

Emnenavn: Operativsystemer

Emnekode: DATA2500

Bokmål

Dato: 1 august 2025

Tid: kl. 09.00-12.00 (3 timer)

Hjelpemidler:

Ingen hjelpemidler er tillatt.

Andre opplysninger:

Les nøye gjennom oppgavene før du begynner og pass på å besvare alle spørsmålene.

Sett gjerne egne forutsetninger dersom du synes oppgaveteksten er uklar. Beskriv forutsetningene og løs oppgaven utifra dem.

Maks poeng på de fleste av deloppgavene er 10 og de bidrar 3.33 % hver til sluttpoengsummen som maksimalt er 300. Men legg merke til at det er to unntak:

Oppgavene 19 og 21 gir inntil 20 poeng

Innlogging til Apache Guacamole

Øverst i browser-vinduet ved siden av Inspira-fanen finner du en fane der det står "Apache Guacamole". Trykk på denne og så får du opp et innloggingsvindu. Her logger du inn med:

Username: kan25k

Password: finn25opp

Etter 5-10 sekunder får du opp et terminal-vindu til en VM som er av samme type som VM-ene som har blitt brukt i kurset, bortsett fra at denne ikke har nett-tilgang. Det kan ta noen sekunder før du får opp et bash-prompt.

1 Datamaskinarkitektur

Hvilken type logisk port er dette sannhetstabell for? A og B er input og UT er resultatet.

A	B	UT
0	0	0
0	1	1
1	0	1
1	1	1

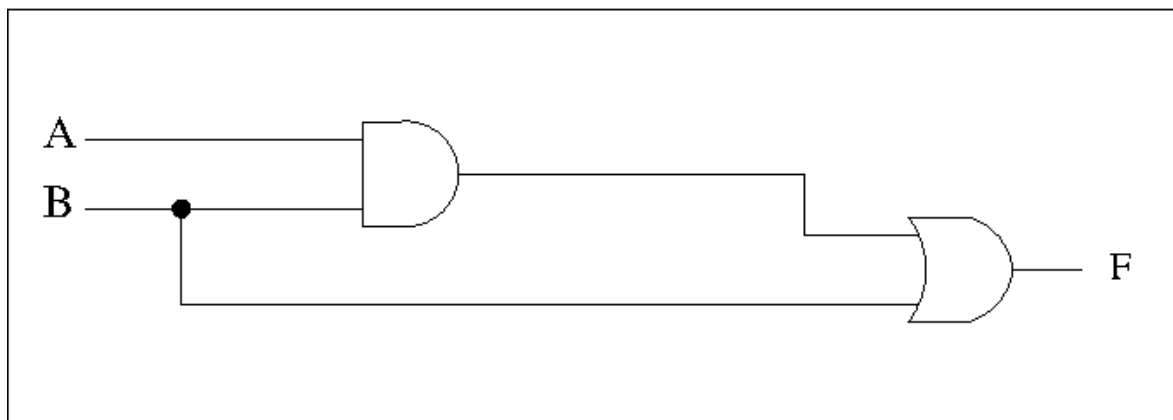
Velg ett alternativ

- ☐ NOR
- ☐ OR
- ☐ NOT
- ☐ PMOS
- ☐ AND
- ☐ NMOS
- ☐ XOR

Maks poeng: 10

2 Datamaskinarkitektur

Hva blir det logiske uttrykket $F(A,B)$ for denne kretsen?



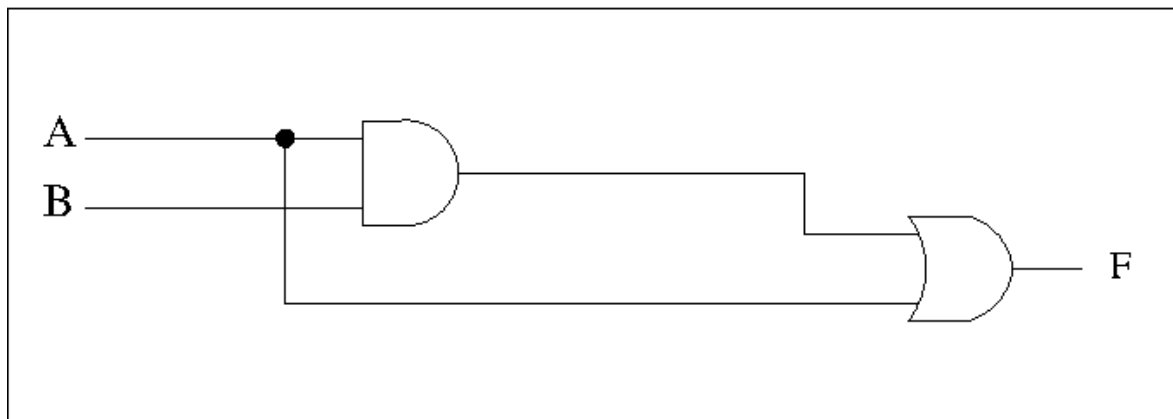
Velg ett alternativ

- ☐ $A \cdot B \cdot B$
- ☐ $\overline{B} \cdot A + \overline{B}$
- ☐ $A \cdot B + B$
- ☐ $\overline{A + B}$
- ☐ $(A + B) \cdot B$
- ☐ $(A + B) \cdot A$
- ☐ $A \cdot B + A$

Maks poeng: 10

3 Datamaskinarkitektur

Uttrykket for den kretsen kan forenkles, utifra sannhetstabellen eller ved å bruke boolsk algebra. Hva kan uttrykket for $F(A,B)$ forenkles til?



Velg ett alternativ

- ☐ 0
- ☐ $A \cdot B$
- ☐ A
- ☐ $\bar{A}$
- ☐ B
- ☐ $\bar{B}$
- ☐ 1
- ☐ $A + B$

Maks poeng: 10

4 Prosesser

Forklar kort hva som menes med PID.

Skriv ditt svar her...

Format

B *I* U $\times_2$ $\times^2$ $\int_x$

Words: 0

Maks poeng: 10

5 Systemkall

Hva er hensikten med et systemkall i et operativsystem? Forklar kort.

Skriv ditt svar her

Format

B *I* U $\times_2$ $\times^2$ I_x

Words: 0

Maks poeng: 10

6 Systemkall

Du utfører Linux-kommandoen **ls /etc/passwd**. Forklar kort om dette vil medføre at det utføres systemkall og isåfall hvorfor dette er nødvendig.

Skriv ditt svar her...

Format ▾

B *I* U x_2 x^2 I_x                    

Words: 0

Maks poeng: 10

7 Prosesser

I denne og de følgende to oppgavene skal vi studere kjøringen av et 100% CPU-avhengig C-program som kun består av en lang for-løkke som regner og regner med heltall. Dette programmet har blitt kompilert og kjørt på en Linux-server, som i teksten bare kalles linux.

Før kjøringen settes timeformat slik:

```
TIMEFORMAT='Real:%R User:%U System:%S %P%%'
```

for at output fra time-funksjonen skal bli mer kompakt.

Følgende viser hvordan programmet kjøres på linux:

```
linux$ gcc -O sum.c
```

```
linux$ time ./a.out
```

```
Real:1.809 User:1.804 System:0.004 99.95%
```

Forklar kort hva kommandoene (vist med uthevet skrift) gjør og hva de fire tallene i output viser.

Skriv ditt svar her...

Format

B *I* U $\times_2$ $\times^2$ $\int_x$

Words: 0

Maks poeng: 10

8 Prosesser

Følgende viser på linux-serveren to forskjellige måter å kjøre programmet på:

```
linux$ for i in {1..4}
> do
> time ./a.out
> done
Real:1.814 User:1.812 System:0.000 99.90%
Real:1.797 User:1.792 System:0.000 99.73%
Real:1.826 User:1.816 System:0.012 100.09%
Real:1.836 User:1.832 System:0.000 99.77%
linux$
```

```
linux$ for i in {1..4}
> do
> time ./a.out&
> done
[1] 1329
[2] 1330
[3] 1332
[4] 1334
linux$
Real:1.803 User:1.796 System:0.000 99.63%
Real:1.816 User:1.812 System:0.000 99.75%
Real:1.854 User:1.828 System:0.020 99.69%
Real:1.866 User:1.848 System:0.012 99.67%
```

Forklar kort hva som skjer i de to kjøringene og forklar spesielt hva som er forskjellen på de to måtene å kjøre programmet på. Vil det være noen forskjell på hvor lang tid de to kjøringene bruker på å fullføres? Forklar kort og gjør en beregning som anslår ca hvor lang tid totalt hver av de to kjøringene tok.

Skriv ditt svar her...

Format

B *I* U $\times_2$ $\times^2$ I_x

Words: 0

Maks poeng: 10

9 Prozessor

Det følgende viser først kompilering og kjøring av det samme 100% CPU-avhengige programmet og så tre forskjellige kjøringene av programmet på serveren linux. Forklar med utgangspunkt i tiden fra den første kjøringen de tidene som er resultatet av de tre kjøringene. Vis ved utregning hvordan størrelsen på Real-tiden omtrentlig kan beregnes utifra resultatet 1.8 sekunder i den første kjøringen. Forklar også prosent-tallene for hver kjøring. Prøv utifra resultatene av kjøringen å finne ut hvor mange cores (med en egen ALU) denne serveren har og om hver core er hyperthreading eller ikke. Forklar kort hvordan du kommer frem til din konklusjon.

```
linux$ gcc -O sum.c
linux$ time ./a.out
Real:1.809 User:1.804 System:0.004 99.95%
linux$ for i in {1..4}
> do
> time ./a.out &
> done
Real:1.819 User:1.812 System:0.000 99.59%
Real:1.833 User:1.812 System:0.004 99.08%
Real:1.833 User:1.812 System:0.000 98.85%
Real:1.855 User:1.836 System:0.004 99.21%
```

```
linux$ for i in {1..6}; do time ./a.out & done
Real:2.510 User:1.800 System:0.000 71.70%
Real:2.617 User:1.800 System:0.000 68.77%
Real:2.661 User:1.804 System:0.008 68.09%
Real:2.686 User:1.828 System:0.004 68.21%
Real:2.806 User:1.800 System:0.008 64.42%
Real:2.832 User:1.840 System:0.000 64.98%
```

```
linux$ for i in {1..8}; do time ./a.out & done
Real:3.553 User:1.816 System:0.000 51.10%
Real:3.594 User:1.844 System:0.000 51.30%
Real:3.625 User:1.828 System:0.004 50.53%
Real:3.623 User:1.812 System:0.016 50.45%
Real:3.642 User:1.820 System:0.008 50.19%
Real:3.650 User:1.812 System:0.004 49.75%
Real:3.671 User:1.820 System:0.008 49.80%
Real:3.746 User:1.836 System:0.004 49.11%
```

Skriv ditt svar her...

[illegible]

Maks poeng: 10

10 Innlogging på Linux-VM

Øverst i browser-vinduet ved siden av Inspira-fanen er det en fane der det står "Apache Guacamole". Når du trykker på denne får du opp et innloggingsvindu der du skal logge inn med

Username: kan25k

Password: finn25opp

og du vil etter 5-10 sekunder få opp et terminal-vindu til en VM som er av samme type som VM-ene som har blitt brukt i kurset, bortsett fra at denne ikke har nett-tilgang. Og det kan så ta noen sekunder før du får opp et bash-prompt.

Du kommer inn i et bash-shell som brukeren "kan25k" i dennes hjemmemappe /home/kan25k på din egen Linux-VM som man som en del av eksamen skal bruke for å løse oppgaver. Denne VMen er nesten helt lik Linux-VMene som dere brukte under kurset, med navn som os13, de er også sysbox Docker-containere, men de har ikke nett-tilgang.

Om man ønsker det, kan Linux-VM under eksamen også brukes til å skrive shell-script og teste ut kommandoer. Men vær oppmerksom på at om man lagrer en fil i Linux-VM og så refresher eller kobler seg til Linux-VM på nytt, vil en ny container-VM startes og filen man har lagret vil være borte; filsystemet vil se slik ut som første gang du koblet deg til.

Det er viktig at du bare har en Linux-VM oppe av gangen, ellers blir ressursene på serverene oppbrukt! Ikke start flere samtidige VMer!

Du kan kopiere tekst i Linux-VM ved å markere den med musen og så taste Shift-Ctrl-Alt. Da kommer det opp et clipboard med den avmerkede teksten og så kan du kopiere med Ctrl-C og paste ut her i Inspira-vinduet med Ctrl-V. Internt inne i Linux-VM kan man kopiere tekst ved å markere den med musen og så paste ved å høyreklikke (som i PowerShell).

Filen **file1** ligger øverst i hjemme-mappen til brukeren du er logget inn som og som har brukernavn "kan25k". Fyll inn innholdet av filen (5 tegn) i boksen under ved å skrive eller kopiere og paste.

Fyll inn:

Maks poeng: 10

11 Filrettigheter

Filen **chmod.txt** ligger øverst i hjemme-mappen til brukeren du er logget inn som og som har brukernavn "kan25k". Filrettighetene til denne filen ser slik ut:

```
-rwxr-x--- 1 kan25k users 0 Jul 24 09:18 chmod.txt
```

Rettighetene til denne filen har blitt satt med kommandoen:

chmod chmod.txt

Fyll inn hva de tre tall-verdiene i kommandoen må ha vært i boksene over.

Maks poeng: 10

12 Docker på Linux-VM

Inne på Linux-VM ligger det tre docker image. Bruk docker til å liste opp alle de tre imageene og paste inn IMAGE ID for hello-world image her (12 tegn):

Hvis docker-service ikke kjører må den først startes med

```
root@os:~# service docker start
```

PS! Hvis du får følgende melding "sudo: unable to resolve host os: Temporary failure in name resolution" trenger du ikke ta hensyn til den, det er bare fordi shellet ikke har nett-tilgang.

Maks poeng: 10

13 Kjøre docker hello-world

Inne på Linux-VM ligger det tre docker image. Bruk docker til å kjøre hello-world image og fyll inn de siste 5 tegnene som output av å kjøre hello-world gir. (ikke ta med mellomrom på slutten av teksten, det siste av de 5 tegnene du skal skrive inn under er "/")

Hvis docker-service ikke kjører må den først startes med

```
root@os:~# service docker start
```

PS! Hvis du får følgende melding "sudo: unable to resolve host os: Temporary failure in name resolution" trenger du ikke ta hensyn til den, det er bare fordi shellet ikke har nett-tilgang.

Maks poeng: 10

14 Kjøre ubuntu docker container

Inne på Linux-VM ligger det også et ubuntu docker image. Bruk dette til å kjøre en ubuntu docker-container, finn inne på containeren filen **/etc/os-release** og les hva som er **VERSION_CODENAME**. Dette er en 5-tegn lang tekst-streng, paste ut eller skriv inn innholdet her (5 tegn):

Maks poeng: 10

15 Kjøre nginx web-container

Inne på Linux-VM ligger det også et image, haugerud/os25, som er ett av imageene som dere skulle laste ned fra Docker Hub i en av konkurranse-oppgavene. Dette imaget er en nginx webserver. Det er en vanlig web-server, men den er spesiell på den måten at man må koble seg til på port 12345 for å se innholdet, og ikke port 80 som er standard for webservere. Kjør en container basert på dette imaget på en slik måte at riktig port er åpnet når du kobler deg til med curl. Når du kobler deg til denne webserveren vil det eneste innholdet på denne være en 10-tegns kode med tilfeldige tegn, paste ut innholdet her (10 tegn):

Hvis du får feilmelding om 'network is unreachable' så har du kanskje ikke spesifisert hvilken versjon (TAG) av imaget du skal bruke. Siden Linux-VM ikke har nett-tilgang, kan du kun kjøre de tre imageene som allerede er lastet ned.

Maks poeng: 10

16 Kommandoer find

Hva gjør følgende find-kommando ?

(bare de 5 første linjene av output er tatt med her)

Denne kommandoer kan være nyttig å bruke i den neste av de praktiske Linux-oppgavene.

```
kan25k@os:~$ find mapper25k -executable -type f -name "run*"
mapper25k/run10
mapper25k/run2
mapper25k/halt.target.wants/udev/run89
mapper25k/halt.target.wants/udev/run65
mapper25k/halt.target.wants/udev/run14
```

Velg ett alternativ:

- ☐ Finner alle mapper i mappen mapper25k.
- ☐ Finner alle mapper i mappen mapper25k og i alle dens undermapper som har et navn som starter med run.
- ☐ Finner alle filer i mappen mapper25k og i alle dens undermapper.
- ☐ Finner alle kjørbare filer i mappen mapper25k og i alle dens undermapper som har et navn som starter med run.
- ☐ Finner alle mapper som kan aksesseres i mappen mapper25k og i alle dens undermapper.
- ☐ Finner alle mapper på hele filsystemet.
- ☐ Finner alle filer i mappen mapper25k som har et navn som starter med run.
- ☐ Finner alle filer på hele filsystemet som har et navn som starter med run.

Maks poeng: 10

17 Kjørbare filer på Linux-VM

Øverst i hjemme-mappen til brukeren du er logget inn som på Linux-VM og som har brukernavn **kan25k** ligger det en mappe som heter **mapper25k**. I denne mappen og nedover i dens undermapper ligger det blant annet 108 filer som heter **run1**, **run2**, ..., **run108**. Dette er program som kan kjøres og som alle gir en streng på 5 tegn når de kjøres. Ett av programmene gir når det kjøres en streng på 5 tegn som inneholder substrengen "WA" (som for eksempel "ahWaf"). *Legg merke til at denne strengen ikke finnes som en streng inne i programmet men er et resultat av å kjøre programmet.* Finn ut hvilken av de 108 strengene du får når du kjører run-programmene som inneholder "WA" og skriv eller kopier og paste ut denne tekststrengen her (5 tegn):

Maks poeng: 10

18 su

På en Linux-VM hvor det finnes en vanlig bruker med navn **user1**, utfører du innlogget som root-brukeren

su user1

Hva skjer?

(Denne kommandoen kan det være nyttig å bruke når du skal teste ulike brukere i de neste oppgavene)

Velg ett alternativ:

- ☐ Du bytter til brukeren user1 uten å måtte oppgi passord.
- ☐ Du bytter til brukeren user1 og må oppgi brukerens passord for å fortsette.
- ☐ Kommandoen feiler fordi root ikke kan bytte til en annen bruker.
- ☐ Du får en feilmelding fordi su ikke er tillatt for root-brukeren.
- ☐ Du sletter brukeren user1 fra systemet.
- ☐ Du oppretter en ny bruker med navnet user1.

Maks poeng: 10

19 ChatGPT-script

Du spør ChatGPT om følgende: *lag et bash script som når det kjøres som root på en ubuntu linux VM lager tre brukere med navn user1, user2 og user3 som alle kan logge inn til hverandres kontoer ved hjelp av ssh-nøkler.*

Som svar får du koden nedenfor som også er lagt inn på Linux-VM du har tilgang til i mappen **/root**. Forklar i store trekk prinsippene som brukes i dette scriptet for å løse oppgaven. Les igjennom hele koden, men forklar **ikke** i detalj hva som skjer linje for linje. Forklar spesielt hva som skjer, men fortsatt i store trekk, i koden etter kommentarlinjen

Konfigurerer SSH-tilgang mellom brukerne

Er det en del av denne siste delen av konfigurasjonen som er overflødig, men som forenkler koden? Forklar kort.

(du blir i neste oppgave bedt om å kjøre scriptet på Linux VM og finne ut om det virker som det skal og du kan gjerne gjøre det nå for å se hvordan det virker i praksis)

```
#!/bin/bash
# Sjekk om scriptet kjøres som root
if [[ $EUID -ne 0 ]]; then
    echo "Dette scriptet må kjøres som root!"
    exit 1
fi
# Bruk en for-løkke for å opprette brukerne user1, user2, og user3
for i in {1..3}; do
    USER="user$i"
    # Opprett brukeren hvis den ikke allerede eksisterer
    if id "$USER" &>/dev/null; then
        echo "Brukeren $USER eksisterer allerede."
    else
        echo "Oppretter bruker $USER..."
        useradd -m -s /bin/bash "$USER"
        echo "$USER:password" | chpasswd
    fi
    # Opprett .ssh-mappen hvis den ikke finnes
    SSH_DIR="/home/$USER/.ssh"
    if [[ ! -d "$SSH_DIR" ]]; then
        echo "Oppretter .ssh-mappen for $USER..."
        mkdir -p "$SSH_DIR"
        chown "$USER:$USER" "$SSH_DIR"
        chmod 700 "$SSH_DIR"
    fi
    # Generer SSH-nøkler for hver bruker
    KEY_FILE="$SSH_DIR/id_rsa"
    if [[ ! -f "$KEY_FILE" ]]; then
        echo "Genererer SSH-nøkler for $USER..."
        sudo -u "$USER" ssh-keygen -t rsa -b 2048 -f "$KEY_FILE" -N ""
    fi
done
# Konfigurerer SSH-tilgang mellom brukerne
for i in {1..3}; do
    USER="user$i"
    AUTH_KEYS="/home/$USER/.ssh/authorized_keys"
    echo "Konfigurerer SSH-tilgang for $USER..."
    > "$AUTH_KEYS" # Tømmer filen (om den finnes) for å starte fra scratch
    for j in {1..3}; do
        OTHER_USER="user$j"
        PUB_KEY_FILE="/home/$OTHER_USER/.ssh/id_rsa.pub"
        if [[ -f "$PUB_KEY_FILE" ]]; then
            cat "$PUB_KEY_FILE" >> "$AUTH_KEYS"
        fi
    done
    # Sett riktige tillatelser
    chown "$USER:$USER" "$AUTH_KEYS"
    chmod 600 "$AUTH_KEYS"
done
echo "Brukerne er opprettet og SSH-konfigurasjonen er fullført!"
```

Skriv svaret ditt her. Endringer blir lagret automatisk.

Format

•
B
I
U
x₂
x²
I_x
↺
↻
↶
↷
↺
↻
↶
↷
Ω
⌂



Words: 0

Maks poeng: 20

20 Kjøre user-script

På Linux-VM du har tilgang til ligger det i mappen **/root** et script **user.sh** som er scriptet du fikk generert ved hjelp av ChatGPT i forrige oppgave. Kjøre scriptet på Linux-VM og gjøre en begrenset test av at det virker som det skal ved å først som root-bruker logge på som **user1** og så vise at du kan gjøre ssh til localhost som **user2** og så videre til **user3** og derfra direkte tilbake til **user1**, uten å måtte skrive passord.

Før du kan teste å gjøre ssh mellom brukerne, må du starte ssh serveren på følgende måte:

```
root@os:~# service ssh start
```

```
* Starting OpenBSD Secure Shell server sshd
```

Vis at testene fungerer ved å paste inn output fra Linux-VM her:

Skriv svaret ditt her. Endringer blir lagret automatisk.

Format

B *I* U $\times_2$ $\times^2$ I_x

Words: 0

Maks poeng: 10

²¹ Lager ChatGPT en 'hemmelig' bakdør?

Når man spør en AI-bot om å generere kode, er det veldig viktig å undersøke koden nøye for å forsikre seg om at den faktisk gjør akkurat det den skal før man kjører koden på en produksjons-server med internett-tilgang. Hvis du klarte å kjøre og teste scriptet **user.sh** i forrige oppgave, fant du forhåpentligvis ut at det fungerer som ønsket og gjør det mulig å gå mellom user-kontoene uten å skrive passord.

Men hvis man bruker dette til å konfigurere en server som kan nås fra internett med ssh, er det en unødvendig konfigurasjon som er tatt med som du ikke ga instruksjoner om i ChatGPT-promptet. Feilen vil raskt kunne føre til at fremmede potensielt kan klare å logge seg på serveren. Studer koden og prøv å finne ut hvilken kodelinje som introduserer denne kritiske sikkerhetsfeilen; forklar kort hva kommandoen gjør, hvorfor den reduserer sikkerheten for systemet og hvordan scriptet må endres for å gjøre systemet sikrere mot inntrenging utenfra. Vis til slutt med output fra Linux-VM at det faktisk er mulig for brukeren **kan25k** å logge seg på som **user1** med ssh til localhost, etter at scriptet har blitt kjørt.

(Denne koden ble faktisk generert av Sikt KI-chat og det er ikke opplagt av hvilken grunn feilen ble introdusert. Enkelte AI-skeptikere vil nok kunne påstå at det kan være et ledd i bot'ens endelige mål om å ta over verden...)

Skriv svaret ditt her. Endringer blir lagret automatisk.

Maks poeng: 20

22 Kritisk avsnitt

Hva er et kritisk avsnitt?

Velg ett alternativ

- ☐ Kode i OS-kjernen som utfører en context switch
- ☐ Kode som er nødvendig for at deler av OS-kjernen skal kunne serialiseres
- ☐ Kode som avgjør om en prosess skal avsluttes på grunn av en deadlock
- ☐ Kode som må utføres veldig hurtig for at en trap til OS-kjernen ikke skal ta for lang tid
- ☐ Kode som må fullføres uten at andre prosesser bruker samme felles ressurs
- ☐ Kode som en prosess kun kan utføre når CPU'en har switchet modus bit til privilegert modus

Maks poeng: 10

23 Busy-waiting

Hva er busy-waiting?

Velg ett alternativ:

- ☐ Metode for å unngå at prosesser som er veldig viktige ikke ligger i samme kø som andre prosesser
- ☐ At en prosess utfører CPU-intensive kode mens den ligger i en I/O-kø
- ☐ At en prosess venter i en while-løkke til en hendelse inntreffer
- ☐ At en prosess venter på et interrupt som er generert av data-bussen
- ☐ Metode for å unngå at det oppstår deadlock
- ☐ At en prosess blir lagt i en wait-kø

Maks poeng: 10

24 lock

Hva gjør x86-instruksjonen lock?

Velg ett alternativ

- ☐ Stopper alle interrupts til CPUen som lock utføres på
- ☐ Stopper CPUen lock utføres på
- ☐ Låser en minneadresse slik at den ikke kan skrives til
- ☐ Stopper trafikken fra alle andre CPUer enn den som lock utføres på til RAM
- ☐ Stopper trafikken fra CPU til RAM for den CPUen lock utføres på

Maks poeng: 10

25 Internminne

Hva er betydningen av begrepet **Random Access Memory**?

Velg ett alternativ

- ☐ At det tar like lang tid å hente en byte fra hvor som helst i minnet
- ☐ At adressene til hver byte i minnet allokeres tilfeldig av operativsystemet
- ☐ At det går like fort å skrive til minnet som å lese fra minnet
- ☐ At det er tilfeldig hvilken byte som hentes først om to byte leses samtidig
- ☐ At data som lagres kan bli lagt hvor som helst i minnet
- ☐ At alle enheter har tilgang til minnet

Maks poeng: 10

26 Interniminne

Mange datamaskiner har et cache-minne mellom CPU og RAM. Vil det kunne føre til at det er forskjell i hvor lang tid det tar for CPU å hente inn to forskjellige bytes fra RAM til registerne i CPU'en? Forklar kort.

Skriv ditt svar her...

Format

B

I

U

x_2

x^2

I_x

Words: 0

Maks poeng: 10

27 Internminne

Et C-program definerer en heltalls-matrise med

```
int mat[5000][5000];
```

Hvis et heltall (integer, int) bruker 4 byte lagringsplass, hvor mange Megabyte består denne matrisen av? (La her Mega bety en million i SI-betydningen)

(se også teksten om hvordan en matrise er bygd opp i neste oppgave)

Skriv ditt svar her...

Format

B *I* U x_2 x^2 I_x

Words: 0

Maks poeng: 10

28 Internminne

En matrise, også kalt et todimensjonalt array, er et sett av elementer ordnet i rader og kolonner. For eksempel kan en 2x2 matrise $A[2][2]$ defineres i et C-program og den vil da ha 2x2 elementer: $A[0][0]$, $A[0][1]$, $A[1][0]$ og $A[1][1]$. Når disse elementene lagres i RAM, lagres de etter hverandre som vist i eksempelet. I en 3x3 matrise lagres først $A[0][0]$, $A[0][1]$ og $A[0][2]$ etter hverandre, så $A[1][0]$, $A[1][1]$ og så videre.

På en maskin med 2 GByte RAM har man følgende C-program:

```
int mat[5000][5000];
for(i = 0; i < 5000; i++){
    for(j = 0; j < 5000; j++){
        mat[i][j] = 5;
    }
}
```

Programmet kompiles og kjøres:

```
$ time a.out
Real:0.113 User:0.020 System:0.092 99.39%
```

Så endres kun en linje i programmet; linjen der matriseverdier legges inn endres til

```
mat[j][i] = 5;
```

Når programmet så kompiles og kjøres, tar det nesten tre ganger så lang tid å kjøre det:

```
$ time a.out
Real:0.303 User:0.216 System:0.084 98.95%
```

Hvordan kan dette forklares?

Skriv ditt svar her...

Format
|
B
I
U
 $\times_2$
 $\times^2$
 $\mathcal{I}_x$
|
📄
📋
|
↶
↷
↺
|
≡
≡
≡
≡
|
Ω
📊
|
✎
|
Σ
|
✖

Words: 0

Maks poeng: 10