

i Eksamensinformasjon

Fakultet: Teknologi, kunst og design

Utdanning: Teknologiske Fag

Emnenavn: Operativsystemer

Emnekode: DATA2500

DEMOPRØVE

Dato: 5. mai til og med 9. mai 2025

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.

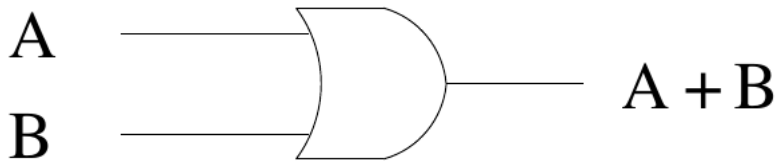
Maks poeng på de fleste av deloppgavene er 10 og de bidrar 3.33 % hver til sluttpoengsummen som maksimalt er 300. Men legg merke til at det er tre unntak:

Oppgave 19 og 24 gir inntil 20 poeng

Oppgave 23 gir inntil 30 poeng

1 Datamaskinarkitektur

Hva viser denne figuren?



Velg ett alternativ

- ☐ En D-vippe
- ☐ En NOT-port
- ☐ En Full Adder
- ☐ En PMOS
- ☐ En AND-port
- ☐ En D-lås
- ☐ En OR-port

Maks poeng: 10

2 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	0
1	0	0
1	1	1

Velg ett alternativ

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

Maks poeng: 10

3 Datamaskinarkitektur

Hvilke konstruksjoner i et høynivåspråk er avhengig av Branch-Control for å kunne utføres?
(det er mulig å velge mer enn ett svar, feil svar gir trekk, men man kan ikke få en negativ score)

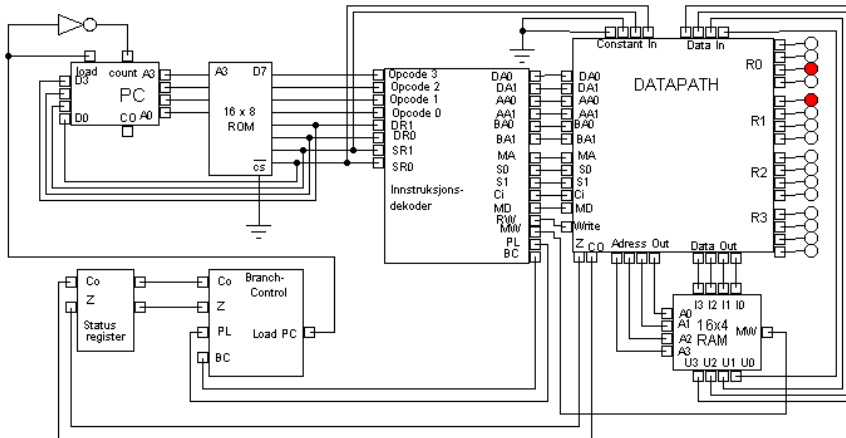
Velg ett eller flere alternativer

- ☐ If-tester
- ☐ Initialisering av variabler
- ☐ multiplikasjoner
- ☐ while-løkker
- ☐ Deklarasjon av variabler
- ☐ addisjoner
- ☐ for-løkker

Maks poeng: 10

4 Datamaskinarkitektur

Hvilken del av denne CPU'en inneholder registerne?



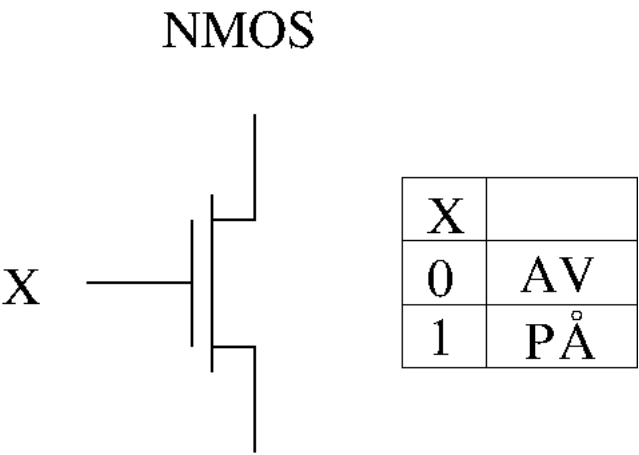
Velg ett alternativ

- ☐ ROM
- ☐ Instruksjonsdekoder
- ☐ Ingen av delene
- ☐ Branch-control
- ☐ PC
- ☐ DATAPATH
- ☐ RAM

Maks poeng: 10

5 Datamaskinarkitektur

Hva viser denne figuren?



Velg ett alternativ

- ☐ En transistor
- ☐ En D-lås
- ☐ En D-vippe
- ☐ En ALU
- ☐ En Full Adder
- ☐ En NOT port
- ☐ En flip-flop

Maks poeng: 10

6 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: kan24k

Password: sale4anon

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. (hvis du får opp en side hvor teksten "All Connections" står midt på siden, trykk på "osex24k" under denne teksten.)

Du kommer inn i et bash-shell som brukeren "kan24k" i dennes hjemmemappe /home/kan24k 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 hver 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; filesystemet 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 "kan24k". Fyll inn innholdet av filen (5 tegn) i boksen under ved å skrive eller kopiere og paste.

Fyll inn:

Maks poeng: 10

7 Docker på Linux-VM

Inne på Linux-VM har det tidligere kjørt en Docker-container som nå er stoppet. Finn ut hva containeren heter (under kolonnen NAMES) og fyll inn navnet her (5 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

8 Starte docker container

Inne på Linux-VM har det tidligere kjørt en container som du i forrige oppgave skulle finne navnet til. Start denne containeren, gå inn på den og finn innholdet av filen /root/xfile2 (en annen mulighet er å først kopiere den fra containeren). Den vil da gi deg en tekst-streng, paste ut innholdet her (5 tegn):

Maks poeng: 10

9 Starte ny docker container

Inne på Linux VM ligger det et ubuntu-image som containeren du skulle starte i forrige oppgave også ble startet med. Kjør en kommando på Linux VM som bruker dette imaget og starter en ny ubuntu-container med navn CONTE. Skriv inn kommandoen under ved å copy-paste fra Linux VM, inkludert output fra terminalvinduet når du kjører kommandoen. Ta også med output fra en docker-kommando som viser at den nye containeren kjører.

Hint: du har ikke tilgang til internett, så hvis du ikke prøver å starte nøyaktig den versjonen av imaget som allerede er lastet ned, vil containeren ikke starte. Det kan også hjelpe å prøve å finne ut hvilken kommando som tidligere ble brukt for å starte den andre containeren som du skulle starte på nytt i forrige oppgave.

Skriv ditt svar her

Format

B *I* U $\times_2$ $\times^2$ $\mathcal{I}_X$

Words: 0

Maks poeng: 10

10 SSH

Du skal med ssh gå passordløst fra server A til server B. Hvor skal privat nøkkel (id_rsa) og offentlig nøkkel (id_rsa.pub) ligge?

Velg ett alternativ:

- ☐ id_rsa på A og id_rsa.pub på B
- ☐ id_rsa på B og id_rsa.pub på A
- ☐ Begge på server A
- ☐ Begge på server B

Maks poeng: 10

11 sudo

Hva er sudo i Linux?

Velg ett alternativ:

- ☐ En teksteditor for filredigering.
- ☐ En kommando som lister kataloginnhold.
- ☐ En kommando for å endre filrettigheter.
- ☐ En pakkehåndterer for programvare.
- ☐ En kommando for å kompilere C-programmer.
- ☐ En kommando som gir root-privilegier.

Maks poeng: 10

14 Sudo ved hjelp av ssh

Du har på Linux-VM en vanlig bruker kalt kan24k. Brukeren har muligheten til å bli root ved å utføre kommandoen **sudo su**. Du bestemmer deg for å legge til en annen måte å bli root på ved å gjøre det mulig for brukeren kan24 å logge seg på localhost som root ved hjelp av ssh og på denne måten bli root uten behov for sudo-rettigheter. Forklar med litt teknisk detalj hvordan du i teorien ville gjort dette på en Linux-host fra kommandolinjen.

Se forrige oppgave om å logge seg på localhost med ssh. I neste oppgave vil du bli bedt om å faktisk konfigurere denne andre metoden for å bli root på Linux-VM du har tilgang til, så du kan gjøre dette parallelt og teste ideene dine. Men merk at for å logge deg på localhost ved hjelp av ssh, må du først starte ssh-tjenesten, se neste oppgave.

Skriv ditt svar her

Format

B *I* U ~~x~~ ^x _x | *I*_x |   |   |    

    |   |  |  | 

Words: 0

Maks poeng: 10

15 Sudo ved hjelp av ssh i praksis

I Linux VM som du har tilgang til (se eksamensinformasjonssiden), er det en normal Linux-bruker kan24k som du logger deg på som. Denne brukeren har muligheten til å bli root ved å gjøre kommandoen **sudo su**. Konfigurer kan24k-kontoen og root-kontoen på en slik måte fra kommandolinjen på Linux VM, at det er mulig for brukeren kan24k å logge seg på localhost som root ved hjelp av ssh. Dette bør gjøre det mulig for kan24k å bli root uten å trenge sudo-privilegiene til kan24k-brukeren.

Dokumenter metoden du bruker for å oppnå dette ved å kopiere og lime inn kommandoene du gjør og utdataene av disse kommandoene fra Linux VM til tekstfeltet nedenfor. Du kan også henvise til den mer teoretiske forklaringen du har gitt i den forrige oppgaven. Selv om du ikke lykkes helt, bør du dokumentere forsøkene dine på å løse denne utfordringen.

Merk at for å logge deg på localhost ved hjelp av ssh, må du først starte ssh-tjenesten.

```
root@os:/home/kan24k# service ssh start
* Starting OpenBSD Secure Shell server sshd
```

Merk også at det å kopiere utdata fra terminalen til en tekstfil i Linuv-VM terminalen, eller til en editor, kan føre til linjeskift i teksten som du ikke ønsker og som potensielt kunne få løsningen din til å mislykkes. Dette kan løses ved å redigere filene eller ved strengt å holde seg til å gjøre kommandoer som unngår kopiering og liming av tekst.

Skriv ditt svar her

Format

B *I* U ~~x₂~~ ^{x²} *I_x*

Words: 0

Maks poeng: 10

16 Plattformavhengighet

På en PC med et Ubuntu 16.04 Linux operativsystem og Intel-prosessor, kompilerer og kjører du et Hello world C-program:

```
rex:~/hello$ gcc hello.c
rex:~/hello$ ./a.out
c: Hello world!
```

Derefter kopierer du filen **a.out** med **scp** til fire forskjellige plattformer. Velg en eller flere av plattformene nedenfor som dette programmet vil kunne kjøre på og produsere samme output (det bidrar med en negativ score om du velger et galt svar, men sammenlagt score for spørsmålet kan ikke bli mindre enn null).

Velg ett eller flere alternativer

- ☐ Ubuntu 16.04 med AMD-prosessor
- ☐ Mac OS X 10.6 med Intel-prosessor
- ☐ Windows 10 med Intel-prosessor
- ☐ FreeBSD med Intel-prosessor

Maks poeng: 10

17 Tråder

Hvilke av de følgende kan deles av tråder som er del av samme prosess?
(det er mulig å velge mer enn ett svar, feil svar gir trekk, men man kan ikke få en negativ score)

Velg ett eller flere alternativer

- ☐ CPU-kjerne
- ☐ Program Counter
- ☐ Variabler
- ☐ Registerverdier
- ☐ Kode
- ☐ Cache
- ☐ Stack
- ☐ Data
- ☐ Scheduling prioritet

Maks poeng: 10

18 Fork

Hvor mange ganger blir HURRA! skrevet ut av dette programmet?

```
#include <stdio.h>
#include <sys/types.h>
#include <stdlib.h>
#include <unistd.h>
int main(){
    int pid;
    pid = fork();
    if(pid == 0){
        printf("1. HURRA fra child!\n");
        fflush(stdout);
        /* fork() returnerer 0 for child-process */
    }

    printf("2. HURRA!\n");
    fflush(stdout);
    fork();
    printf("3. HURRA!\n");
    fflush(stdout);
}
```

(*fflush(stdout)* er tatt med for å hindre kopiering av et buffer som ikke er skrevet ut ennå. Sørger for samme output også om man sender output videre til en pipe eller fil)

Velg ett alternativ:

- | | | | | | | | | |
|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 16 |
| ☐ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |

Maks poeng: 10

19 Parallele tråder

I forelesningene så vi på et Java-program med navn Calc.java som kjører to tråder som regner intensivt og bruker all CPU de kan få tak i. Programmet Calc1.java kjører bare en tråd av samme type.

De fire scriptene a.sh, b.sh, c.sh og d.sh som vises nedenfor kjøres på en slik Linux-VM som dere hadde på Linux-lab'en (ikke Linux VM som du har er på eksamen). Linux-VMen som scriptene kjøres på har to cores, det vil si, to uavhengige CPU-regneenheter.

Hvilket eller hvilke av scriptene blir først ferdig?

(Om to script bruker omtrent like lang tid, regnes de som like raske)

Forklar kort hvordan du kom frem til svaret.

```
group131@osl131:~$ cat a.sh
#!/bin/bash
java Calc
java Calc
group131@osl131:~$ cat b.sh
#!/bin/bash
java Calc&
java Calc
group131@osl131:~$ cat c.sh
#!/bin/bash
java Calc1
java Calc1
java Calc1
java Calc1
group131@osl131:~$ cat d.sh
#!/bin/bash
java Calc1&
java Calc1&
java Calc1&
java Calc1
```

Skriv ditt svar her

Format

B *I* U ~~x_e~~ ^{x²} *I_x*

Words: 0

Maks poeng: 20

20 PowerShell

Hvordan finner du ut hvilke properties og metoder som objektene som returneres av kommandoen **ls** (som egentlig er et alias for **Get-Childitem**) har?

Velg ett alternativ

- ☐ ls | Get-Member
- ☐ ls | grep Property
- ☐ Get-Properties | ls
- ☐ ls.GetProperty
- ☐ (ls).GetProperty

Maks poeng: 10

21 PowerShell

I PowerShell gjør du følgende kommando:

PS C:\Users\levav\mydir> Get-Childitem

Directory: C:\Users\levav\mydir

Mode	LastWriteTime	Length	Name
-a----	22.06.2017 21.14	26807	dok.pdf
-a----	01.12.2015 08.24	59481	License.pdf
-a----	26.06.2017 11.15	609017	Room.pdf

Hvilken av de følgende kommandoene vil gi kun

dok.pdf

som output?

Velg ett alternativ

- ☐ (Get-Childitem dok.pdf).Name
- ☐ Get-Childitem *.pdf
- ☐ Get-Childitem | grep dok.pdf
- ☐ Get-Childitem.dok.pdf.Name
- ☐ Get-Childitem | grep dok.pdf | cut -f 5
- ☐ Get-Name \$(Get-Childitem dok.pdf)
- ☐ Get-Childitem dok.pdf

Maks poeng: 10

22 PowerShell: Årstall for fil

Gi en PowerShell-kommando som gir året for når filen `/etc/gai.conf` sist ble endret.

Du kan teste ut kommandoen i PowerShell i Linux VM med

```
kan24k@os:~$ pwsh
PowerShell 7.4.4
PS /home/kan24k>
```

men vær oppmerksom på at **ls** ikke er et alias for **Get-ChildItem** i denne versjonen av PowerShell.

Skriv ditt svar her

Format

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

Words: 0

Maks poeng: 10

23 Disk-eater

Du er en systemadministrator for et selskap som bruker Docker for sine applikasjoner. Nylig har du lagt merke til at noen Docker-containerer bruker en betydelig mengde diskplass, spesielt containere som overstiger 10 GByte. Disse overdimensjonerte containerne forårsaker ytelsesproblemer og må håndteres.

Du spør Chat-GPT om hjelp og ber om følgende:

Skriv et Bash-skript som:

- Itererer over alle Docker-containerer i katalogen `/var/lib/docker/containers/`.
- Beregner størrelsen på hver Docker-container.

Hvis en Docker-container overstiger 10 GByte, skal den utføre følgende handlinger:

- Logg ID-en til den overdimensjonerte containeren, og en melding som indikerer at containeren blir avsluttet på grunn av størrelsen. Denne loggen skal legges til en loggfil som ligger på `/root/diskeaterLog.txt`.
- Avslutt den overdimensjonerte Docker-containeren.

Du får følgende bash-script som svar (det ligger også på `/home/kan24k/diskeater.sh` hvor du kan teste det):

```
#!/bin/bash
# Definer variabler
CONTAINER_DIR="/var/lib/docker/containers"
LOG_FILE="/root/diskeaterLog.txt"
SIZE_LIMIT=10737418240 # 10 GByte i bytes
# Lag loggfilsmappe hvis den ikke eksisterer
mkdir -p $(dirname "$LOG_FILE")
# Iterer over alle Docker-containerer
for CONTAINER in $(docker ps -aq); do
    # Finn containerens størrelse
    CONTAINER_SIZE=$(du -sb $CONTAINER_DIR/$CONTAINER | awk '{print $1}')

    # Sjekk om containeren overstiger størrelsesgrensen
    if [ "$CONTAINER_SIZE" -gt "$SIZE_LIMIT" ]; then
        # Logg informasjon
        echo "ID: $CONTAINER stoppet, $CONTAINER_SIZE bytes" >> "$LOG_FILE"

        # Avslutt containeren
        docker stop "$CONTAINER"
    fi
done
```

Men når du kjører det skjer følgende:

```
root@os:/home/kan24k# ./diskeater.sh
du: cannot access '/var/lib/docker/containers/811d0e96ceab': No such file or directory
./diskeater.sh: line 17: [: : integer expression expected
du: cannot access '/var/lib/docker/containers/3c992739a39f': No such file or directory
./diskeater.sh: line 17: [: : integer expression expected
```

Du undersøker litt nærmere og finner ut følgende:

```
root@os:/home/kan24k# ls -l /var/lib/docker/containers
total 8
drwx--x--- 4 root root 4096 Aug  1 09:16 3c992739a39f1e0a9127476ac0039d51140b56b033f1a8210674b32869c19f4b
drwx--x--- 4 root root 4096 Aug  1 09:18 811d0e96ceabba9cefe515f58185078036bb468f6cc2ccbcf8e761ce11ce2f7d
root@os:/home/kan24k#
```

Forklar kort hva som er feil med scriptet og endre scriptet slik at det virker. Du trenger ikke å skrive inn hele scriptet, bare endringen. Du kan også teste ut dette på Linux-VM, scriptet ligger under `/home/kan24k`.

Skriv ditt svar her

Format
B
I
U
x₂
x²
I_x
📄
📁
↶
↷
↺
↻
☰
☲
☱
☴
Ω
📊
🖋️
Σ
✖️

Words: 0

Maks poeng: 30

24 Disk-eater i praksis

Du ønsker å teste den korrigerte versjonen av `/home/kan24k/diskeater.sh` fra forrige oppgave på Linux-VM, men containeren som har ligget der og nye containere som du eventuelt har startet, bruker ikke mer enn 10GByte disk. Forklar kort hvordan du kan endre scriptet slik at du kan teste at det fungerer som det skal, stopper containere og skriver til loggen.

Utfør så dette i praksis på Linux-VM og kopier inn output fra Linux-VM i svar-feltet under som viser at scriptet virker som det skal.

Forklar kort om scriptet gjør forskjell på containere som kjører og containere som har blitt stoppet. Forklar også kort om dette er et alvorlig problem med scriptet og hvordan problemet kan løses (men du trenger ikke å eksplisitt skrive kode som løser problemet).

Skriv ditt svar her

Format | **B** *I* U $\times_e$ $\times^2$ | $\int_x$ |   |    |     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 

 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 

 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 

 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 

 |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 

25 Internminne

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

Velg ett alternativ

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

Maks poeng: 10

26 RAM

I hvilke tilfeller lever ikke begrepet **Random Access Memory (RAM)** opp til sin egentlige betydning?

(det bidrar med en negativ score om du velger et galt svar, men sammenlagt score for spørsmålet kan ikke bli mindre enn null).

Velg ett eller flere alternativer

- ☐ Når en server har så mange CPUer at de er delt inn i numa-nodes.
- ☐ Når datamaskinen har vært på for lenge uten en omstart.
- ☐ Når prosessoren kjører i kernel mode.
- ☐ Når det har oppstått en deadlock.
- ☐ Når det kjøres tunge databehandlingsoppgaver som stort sett bruker CPU.
- ☐ Når noen av dataene som hentes inn finnes i cache.

Maks poeng: 10