

i Eksamensinformasjon

Fakultet: Teknologi, kunst og design

Utdanning: Teknologiske Fag

Emnenavn: Operativsystemer

Emnekode: DATA2500

Bokmål

Dato: 9 august 2023

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 noen unntak:

Oppgave 13 og 18 gir inntil 20 poeng

Oppgave 16 gir inntil 30 poeng

1 Logiske porter

Hvilke input til en OR-port vil gi output lik 1?

(Ett eller flere riktige svar. Trekk for gale svar)

Velg ett eller flere alternativer

- ☐ 0
- ☐ 01
- ☐ true
- ☐ 00
- ☐ 11
- ☐ 10
- ☐ 1
- ☐ false

Maks poeng: 10

2 Logiske porter

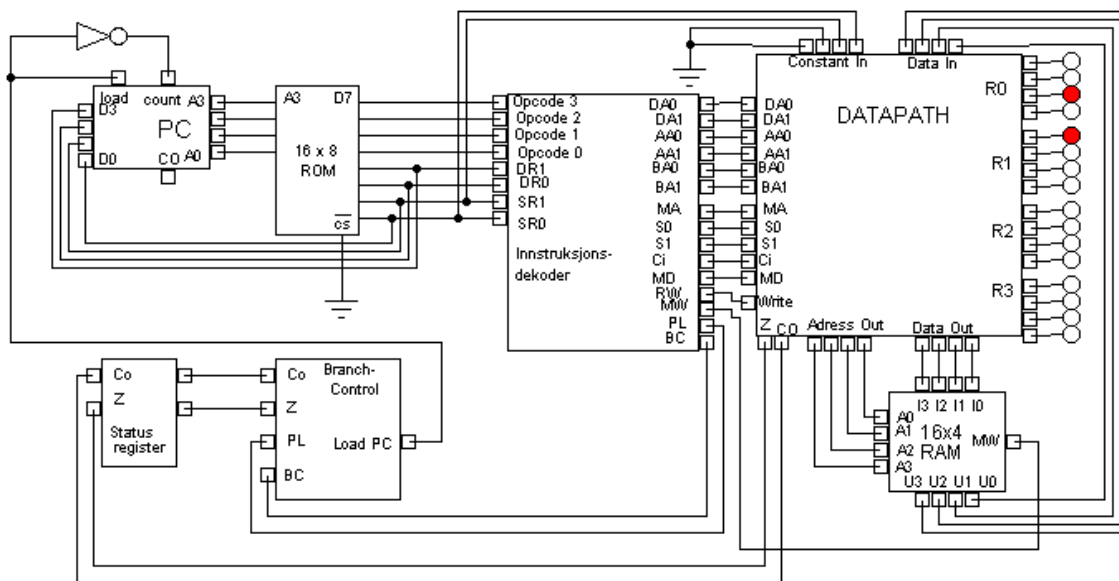
Skriv inn sannhetstabellen for en AND-port med input A og B:

A	B	UT
0	0	
0	1	
1	0	
1	1	

Maks poeng: 10

3 Datamaskinarkitektur 22k

Hvilken del av denne CPU'en sørger for at det er mulig å hoppe i koden, slik at man kan lage if-tester og løkker i programmet?



Velg ett alternativ

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

Maks poeng: 10

4 Innlogging på Linux-VM

Nederst i Inspira-vinduet er det en knapp der det står "Linux VM". Når du trykker på denne første gang får du opp et "APACHE GUACAMOLE" innloggingsvindu der du skal logge inn med

Username: kan

Password: konte22

Da får du opp en side hvor teksten "All Connections" står midt på siden. Trykk på "os" under denne teksten og du vil etter 6-7 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.

Du kommer inn i et bash-shell som brukeren "kan" i dennes hjemmemappe /home/kan 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 avmerkde 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 "kan". Fyll inn innholdet av filen (5 tegn) i boksen under.

Fyll inn:

Maks poeng: 5

5 Fil på Linux-VM

I filesystemet til brukeren du logget på med på Linux-VM ligger det 200 filer i en mappe som heter **mangefiler**. En av filene i denne mappen inneholder tekststrengen "rett fil" i andre linje, finn denne filen, kopier og paste ut innholdet av første linje i denne filen her (5 tegn):

Maks poeng: 10

6 sys.txt på Linux-VM

I en mappe på dette systemet ligger det en fil som heter **sys.txt**. Finn den og kopier og paste ut innholdet av denne tekstfilen her (5 tegn):

Maks poeng: 10

7 Filrettigheter k22

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

```
-rw-r----- 1 kan users 0 May 18 14:33 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

8 Tidsstempel på Linux-VM

Øverst i hjemme-mappen til brukeren du er logget inn som og som har brukernavn "kan" ligger det en mappe som heter tusenfiler. I denne mappen er det tusen filer som alle ble endret på forskjellige tidspunkt. En av disse filene ble endret 11 Februar 2013. Finn den og kopier og paste ut innholdet av denne tekstfilen her (5 tegn):

Hint: Et søk av typen

```
find . -type f -newermt "2 Jan 2019 19:55" -ls
```

kan hjelpe, men det finnes også andre metoder for å kunne lete seg frem til filen.

Maks poeng: 10

9 Docker på Linux-VM

Inne på Linux-VM ligger det et Docker-image. Ved å bruke en kommando for å liste alle Docker-image kan du finne dette images IMAGE ID. Finn denne og fyll den inn her (med alle 12 tegnene):

Hvis docker-service ikke kjører må den først startes med

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

Maks poeng: 10

10 Fil på docker-container

Inne på Linux-VM har det tidligere kjørt en container som er basert på imaget fra forrige oppgave. Start 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

11 Samtidige tråder i Linux-VM

I denne oppgaven er det nyttig å vite at containeren som kjører det vi kaller Linux-VM er startet med **run --cpus="2"** slik at programmer som kjører her har tilgang til to CPUer.

På Linux-VM (under ressurser) er det en mappe **/home/kan/lock** hvor det ligger tre filer. Kompiler to av disse filene med

```
kan@os:~/lock$ gcc -O -pthread thread.c en.c
```

og kjør det resulterende programmet noen ganger. Forklar kort (uten for mye detaljer) hva programmet gjør og hvorfor sluttresultatet blir forskjellig fra gang til gang. Forklar kort hva opsjonen **-O** betyr (og legg merke til at denne opsjonen vil ha betydning for kjøringen i neste oppgave).

Skriv ditt svar her

Format

B **I** **U** x_e x^2 I_x

Σ 

Words: 0

Maks poeng: 15

12 Tråder på samme CPU

Hvis du nå kjører programmet, det du kompilerte i forrige oppgave, mange ganger på følgende måte:

```
kan@os:~/lock$ taskset -c 0 ./a.out
```

vil du se at resultatet alltid blir 20000000.

Hvis du derimot kompilerer programmet uten **-O** opsijonen

```
kan@os:~/lock$ gcc -pthread thread.c en.c
```

og igjen kjører det mange ganger som vist over med taskset, vil du se at resultatet stort sett blir 20000000, men at det noen ganger blir et svar som varierer (du må ikke eksplisitt demonstrere at det blir sånn).

Forklar kort hvorfor det blir forskjell på resultatene med og uten opsijonen **-O** (*det kan være nyttig å gjøre neste oppgave først for å kunne forklare forskjellen*). Sammenlign også med resultatene i forrige oppgave der du kjørte uten taskset og forklar kort forskjellen på resultatene.

Skriv ditt svar her

Format

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

Σ 

Words: 0

Maks poeng: 15

13 Fra C til Assembly

Kompiler nå programmet `en.c` med følgende kommando:

```
kan@os:~/lock$ gcc -O -S en.c
```

Forklar kort hva som skjer når du kompilerer `en.c` på denne måten. Kompiler så programmet `en.c` uten opsijonen `-O`.

```
kan@os:~/lock$ gcc -S en.c
```

Sammenlign resultatet du får med og uten opsijonen `-O` og forklar kort hvordan de resulterende filene kan brukes til å forklare hvorfor resultatet av kjøringene i forrige oppgave blir forskjellige.

Skriv ditt svar her

Format

B

I

U

x_2

x^2

I_x

Σ

Words: 0

Maks poeng: 15

14 Threads og cores

På en Linux server med hyperthreading får du følgende output fra en kommando:

```
$ grep "" /sys/devices/system/cpu/cpu*/topology/thread_siblings_list
/sys/devices/system/cpu/cpu0/topology/thread_siblings_list:0,4
/sys/devices/system/cpu/cpu1/topology/thread_siblings_list:1,5
/sys/devices/system/cpu/cpu2/topology/thread_siblings_list:2,6
/sys/devices/system/cpu/cpu3/topology/thread_siblings_list:3,7
/sys/devices/system/cpu/cpu4/topology/thread_siblings_list:0,4
/sys/devices/system/cpu/cpu5/topology/thread_siblings_list:1,5
/sys/devices/system/cpu/cpu6/topology/thread_siblings_list:2,6
/sys/devices/system/cpu/cpu7/topology/thread_siblings_list:3,7
```

Utifra dette resultatet kan man konkludere med at serveren har

cores og

thread(s) per core.

Fyll inn riktig tall i boksene over.

Maks poeng: 10

15 En CPU-avhengige prosess

En 100% CPU-avhengig prosess kjører alene og direkte på en CPU på en Linux-server uten virtualisering. Du ser på statistikk fra `/proc/stat` for CPUen den kjører på. I hvilken modus er da neste alle ticks som rapporteres?

Velg ett alternativ:

- ☐ irq
- ☐ system
- ☐ steal
- ☐ guest_nice
- ☐ nice
- ☐ user
- ☐ softirq
- ☐ iowait
- ☐ idle
- ☐ guest

Maks poeng: 10

16 Lagringsenheter

En byte skal kopieres til et register i CPU. Hvilken av følgende enheter går det raskest å kopiere fra?

Velg ett alternativ

- ☐ SSD disk
- ☐ L2 cache
- ☐ RAM
- ☐ HDD disk
- ☐ L3 cache
- ☐ L1 cache

Maks poeng: 10

17 RAID 3

Anta at du har et RAID 3 oppsett med 4 disker og en paritets-disk. Alle dataene på disk 3 har gått tapt etter en disk-crash. Bruk informasjonen fra paritets-disken til å gjenopprette det som var på disk 3 og skriv inn 0 eller 1 for hver rad på disk 3.

RAID 3				
disk 1	disk 2	disk 3	disk 4	paritets-disk
0	0		1	0
0	1		1	1
1	0		0	1
0	0		1	1
1	1		0	1
0	1		1	0

Maks poeng: 10

18 Linux kommandolinje

NB! Kommandoene som brukes i denne og de neste tre oppgavene er nyttige for å løse script-oppgaven som kommer etter disse. Eksempelene er kjørt i Linux-VMen du har tilgang til i Inspira og du kan teste ut kommandoene der om du ønsker å gjøre det.

Hva gjør følgende Linux-kommando?

```
kan@os:~/mem$ echo "xxx" >> mem1.c
```

Velg ett alternativ:

- ☐ Legger til teksten "xxx" til slutt i filen mem1.c.
- ☐ Legger til teksten "xxx" helt først i filen mem1.c.
- ☐ Skriver teksten "xxx" i terminalvinduet.
- ☐ Skriver teksten "xxx" i filen mem1.c. Eventuelt tidligere innhold slettes.
- ☐ Fjerner alle forekomster av teksten "xxx" i filen mem1.c.
- ☐ Endrer navnet på filen mem1.c til "xxx".

Maks poeng: 10

19 Kommandoer find

Hva gjør følgende find-kommando?

```
kan@os:~/mem$ find . -type f
./dir/mem3.c
./dir/mem4.c
./mem1.c
./mem2.c
kan@os:~/mem$
```

Velg ett alternativ:

- ☐ Finner alle mapper i mappen du står i.
- ☐ Finner alle mapper på hele filsystemet.
- ☐ Finner alle filer i mappen du står i og i alle dens undermapper.
- ☐ Finner alle mapper i mappen du står i og i alle dens undermapper.
- ☐ Finner alle filer i mappen du står i.
- ☐ Finner alle filer på hele filsystemet.

Maks poeng: 10

20 MD5-sum av en fil

Forklar kort hva som skjer når de følgende tre kommandoene kjøres. Forklar spesielt hvordan kommandoen **md5sum** kan brukes til å finne ut om en fil har blitt endret.

```
kan@os:~/mem$ md5sum mem1.c
1fd4c21162f21633e862b3af2e0511a8  mem1.c
kan@os:~/mem$ echo "xxx" >> mem1.c
kan@os:~/mem$ md5sum mem1.c
8e923f34fcb16c23f6627896054a2c81  mem1.c
```

Skriv ditt svar her

Format

B

I

U

x_2

x^2

I_x

Σ

Words: 0

Maks poeng: 10

21 MD5sum for flere filer

Forklar kort hva som skjer når de følgende fire kommandoene kjøres. Se ved behov på manualsiden for denne kommandoen. Forklar spesielt hvordan kommandoen **md5sum** kan brukes til å finne ut om en eller flere filer har blitt endret.

```
kan@os:~/mem$ md5sum mem1.c > sum.txt
kan@os:~/mem$ md5sum mem2.c >> sum.txt
kan@os:~/mem$ echo "xxx" >> mem1.c
kan@os:~/mem$ md5sum -c sum.txt
mem1.c: FAILED
mem2.c: OK
md5sum: WARNING: 1 computed checksum did NOT match
```

Skriv ditt svar her

Format

B

I

U

x₂

x²

I_x

Σ

Words: 0

Maks poeng: 10

22 Bash-script som sjekker om filer har endret seg

I denne oppgaven skal du laget et bash-script **check.sh** som ved hjelp av kommandoen **md5sum** sjekker om noen av filene i en mappe og alle dens undermapper har endret seg. For å vise hvordan scriptet skal virke, vises nedenfor hvordan det vil se ut om scriptet kjøres i mappen **mem** på Linux-VMen du har tilgang til i Inspira. Etter at scriptet har blitt lagt i mappen vil kommandoen **tree -A** gi følgende:

```
kan@os:~/mem$ tree -A
```

```
.
├── check.sh
├── dir
│   ├── mem3.c
│   └── mem4.c
├── mem1.c
└── mem2.c
1 directory, 5 files
```

Når scriptet **check.sh** kjøres, skal det lage en fil **sum.txt** i mappen det kjører i som inneholder md5summen til alle filer i denne mappen og i undermapper, etter første kjøring skal det se slik ut:

```
kan@os:~/mem$ ./check.sh
```

```
kan@os:~/mem$ tree -A
```

```
.
├── check.sh
├── dir
│   ├── mem3.c
│   └── mem4.c
├── mem1.c
├── mem2.c
└── sum.txt
1 directory, 6 files
```

Filen **sum.txt** skal bare lages hvis den ikke finnes fra før. Hvis den finnes fra før skal scriptet istedet sjekke om det har skjedd noen endringer i noen av filene i mappen og dens undermapper. Når **check.sh** kjøres for andre gang uten at noen endringer har skjedd, skal det derfor se slik ut:

```
kan@os:~/mem$ ./check.sh
```

```
./dir/mem3.c: OK
./dir/mem4.c: OK
./mem1.c: OK
./mem2.c: OK
./check.sh: OK
```

NB! Legg merke til at output automatisk blir slik som beskrevet når du bruker kommandoen **md5sum slik som demonstrert i de foregående oppgavene.**

Hvis man har gjort en endring på en eller flere av filene, skal scriptet rapportere om det som i følgende eksempel hvor vi først endrer på filen **dir/mem3.c**:

```
kan@os:~/mem$ echo "xxx" >> dir/mem3.c
```

```
kan@os:~/mem$ ./check.sh
```

```
./dir/mem3.c: FAILED
./dir/mem4.c: OK
./mem1.c: OK
./mem2.c: OK
./check.sh: OK
md5sum: WARNING: 1 computed checksum did NOT match
```

Dermed kan man ved å kjøre **check.sh** ha kontroll på om noen filer har blitt endret siden første gang man kjørte scriptet.

Skriv ditt svar her

1	
---	--

Maks poeng: 40

23 Liste filer i PowerShell

På Linux-VM kan det se slik ut når man starter PowerShell med kommandoen pwsh:

```
kan@os:~$ pwsh
PowerShell 7.2.4
Copyright (c) Microsoft Corporation.
https://aka.ms/powershell
Type 'help' to get help.
PS /home/kan> cd win
PS /home/kan/win>
```

PowerShell-kommandone som brukes i denne og den neste oppgaven kan det være nyttig å kjenne når den siste PowerShell script-oppgaven skal løses.

Forklar kort hva de tre følgende PowerShell-kommandoene (vist i boldface) gjør:

```
PS /home/kan/win> Get-ChildItem -Recurse -File
    Directory: /home/kan/win
UnixMode      User          Group          LastWriteTime         Size Name
-----
-rw-r--r-- kan      kan           08/02/2022 13:32      724 md5.ps1
-rw-r--r-- kan      users        08/02/2022 13:13      192 mem1.c
-rw-r--r-- kan      users        08/02/2022 13:13      192 mem2.c
    Directory: /home/kan/win/dir
UnixMode      User          Group          LastWriteTime         Size Name
-----
-rw-r--r-- kan      users        08/02/2022 13:31      194 mem3.c
-rw-r--r-- kan      users        08/02/2022 13:13      192 mem4.c
PS /home/kan/win> (Get-ChildItem -Recurse -File).Name
md5.ps1
mem1.c
mem2.c
mem3.c
mem4.c
PS /home/kan/win> Get-ChildItem -Recurse -File | Get-Member
    TypeName: System.IO.FileInfo
Name      MemberType      Definition
-----
Attributes Property        System.IO.FileAttributes Attributes {get;set;}
CreationTime Property        datetime CreationTime {get;set;}
CreationTimeUtc Property        datetime CreationTimeUtc {get;set;}
Directory Property        System.IO.DirectoryInfo Directory {get;}
DirectoryName Property        string DirectoryName {get;}
Exists Property        bool Exists {get;}
Extension Property        string Extension {get;}
FullName Property        string FullName {get;}
IsReadOnly Property        bool IsReadOnly {get;set;}
LastAccessTime Property        datetime LastAccessTime {get;set;}
LastAccessTimeUtc Property        datetime LastAccessTimeUtc {get;set;}
LastWriteTime Property        datetime LastWriteTime {get;set;}
LastWriteTimeUtc Property        datetime LastWriteTimeUtc {get;set;}
Length Property        long Length {get;}
LinkTarget Property        string LinkTarget {get;}
Name Property        string Name {get;}
```

(kun et utvalgt del av output er vist fra den siste kommandoen) På grunnlag av disse kommandoene, skriv ned en PowerShell-kommando som gir følgende output:

```
PS /home/kan/win> PowerShell-kommando  
/home/kan/win/md5.ps1  
/home/kan/win/mem1.c  
/home/kan/win/mem2.c  
/home/kan/win/dir/mem3.c  
/home/kan/win/dir/mem4.c
```

Skriv ditt svar her

Format | **B** | *I* | U | x_e | x^e | I_x | | | | | | | | |

Σ |

Words: 0

Maks poeng: 10

24 Md5sum i PowerShell

I PowerShell kan man regne ut MD5-summen for en fil på følgende måte:

```
PS /home/kan/win> Get-FileHash mem1.c -Algorithm MD5
Algorithm      Hash                                     Path
-----
MD5            1FD4C21162F21633E862B3AF2E0511A8      /home/kan/win/mem1.c
```

Prøv å finne ut på Linux-VM om dette er den samme algoritmen som brukes i Linux. Forklar kort hvorfor du konkluderer som du gjør.

Skriv så ned en PowerShell-kommando som gir følgende output:

```
PS /home/kan/win> PowerShell-kommando
1FD4C21162F21633E862B3AF2E0511A8
```

Skriv ditt svar her

Format
|
B
I
U
x₂
x²
I_x
|
📄
📋
|
↶
↷
↺
|
☰
☷
|
Ω
📱
|
✎

Σ
|
✖

Words: 0

Maks poeng: 10

Words: 0

Maks poeng: 20