Eksamensinformasjon

Fakultet: Teknologi, kunst og design
Utdanning: Teknologiske Fag
Emnenavn: Operativsystemer
Emnekode: DATA2500

Bokmål
Dato: 30 mai 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 3, 15 og 19 gir inntil 20 poeng
Oppgave 9, 11 og 16 gir inntil 30 poeng

1 Linux kommandolinje

Hvilken Linux-bruker kan endre på alt i Linux-filsystemet?

Velg ett alternativ

- daemon
- systemd
- administrator
- nobody
- root
- superuser
- bin

Maks poeng: 10

2 Linux kommandolinje

Hvilken Linux-kommando lister filer og mapper?
Velg ett alternativ

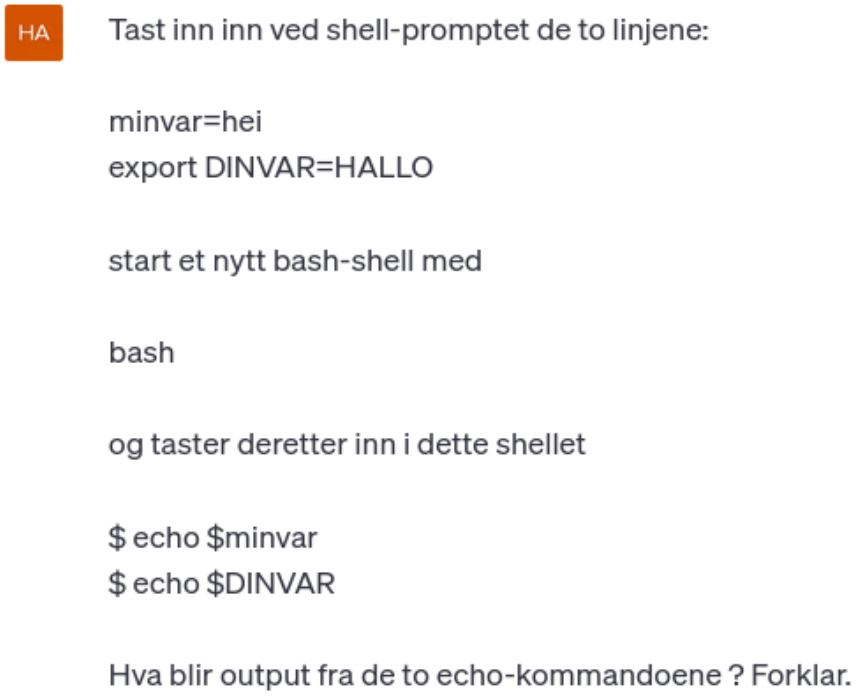
- show
- pwd
- ls
- chmod
- cp
- list
- mv
- dir

Maks poeng: 10

3 Arv av variabler

ChatGPT har tatt verden med storm og den gir raske og velformulerte svar på det meste av spørsmål. Svarene den gir er ofte korrekte, men ikke alltid og det er viktig å kunne vurdere hva som korrekt enten det er svar fra ChatGPT eller noe man finner på Stack Overflow.

Følgende spørsmål fra en obligatorisk oppgave i dette kurset ble gitt til ChatGPT:



Og den gav følgende svar. Vurder kort hvor riktig dette svaret er, rett opp eventuelle feilaktige påstander og kom med din egen forklaring.

Når du skriver inn de to linjene i shellet:

shell Copy code

```
minvar=hei
export DINVAR=HALLO
```

Setter du verdien "hei" til variabelen `minvar` og eksporterer variabelen `DINVAR` med verdien "HALLO" som en global variabel.

Deretter starter du et nytt bash-shell ved å skrive `bash` i terminalen. I det nye shellmiljøet vil de tidligere definerte variablene bli arvet.

Når du skriver følgende kommandoer i det nye bash-shell-miljøet:

shell Copy code

```
$ echo $minvar
$ echo $DINVAR
```

vil du få følgende output:

shell Copy code

```
hei
HALLO
```

Forklaring:

- `echo $minvar` skriver ut verdien til `minvar`, som er "hei".
- `echo $DINVAR` skriver ut verdien til den globale variabelen `DINVAR`, som er "HALLO". Denne variabelen ble eksportert og er tilgjengelig i det nye shellmiljøet.

Skriv ditt svar her

Format

Words: 0

Maks poeng: 20

4 Innlogging på Linux-VM

Nederst i Inspera-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: 8ex2023

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; 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 avmerkde teksten og så kan du kopiere med Ctrl-C og paste ut her i Inspera-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: 10

5 Fil 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 **tusenmapper**. I denne mappen er det 1000 mapper i tre nivåer med 10 mapper på hvert nivå og inne i en av disse mappene ligger det en fil.

Finn denne filen, skriv eller kopier og paste ut innholdet av filen her(5 tegn):

Maks poeng: 10

6 Docker på Linux-VM

Inne på Linux-VM har det tidliger 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

7 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

8 Kjøre mange program på Linux-VM

Øverst i hjemme-mappen til brukeren du er logget inn som på Linux-VM ressursen og som har brukernavn **kan** ligger det en mappe som heter **run**. I denne mappen er det fem hundre filer som heter **run1**, **run2**, ... , **run500**. Dette er program som kan kjøres og som alle gir en streng på 5 tegn når de kjøres. Ett av programmene gir en streng på 5 tegn som inneholder substrengen "R3" (som for eksempel "aR3kX"). Finn ut hvilken av de 500 strengene som inneholder "R3" og kopier og paste ut denne tekststrengen her (5 tegn):

Legg merke til at du i neste oppgave blir bedt om å skrive et bash-script som løser dette problemet og at du dermed kan få noe igjen for innstatsen selv om du ikke har et script som fungerer så bra at det i praksis finner denne tekststrengen.

Maks poeng: 10

9 Script som kjører program

I denne oppgaven skal du laget et bash-script **find.sh** som på en systematisk måte løser problemet som ble gitt i forrige oppgave. Hvis du finner en fungerende løsning, kan den brukes til å løse forrige oppgave, men hvis den ikke virker helt som den skal kan du få noe uttelling for arbeidet du har gjort her.

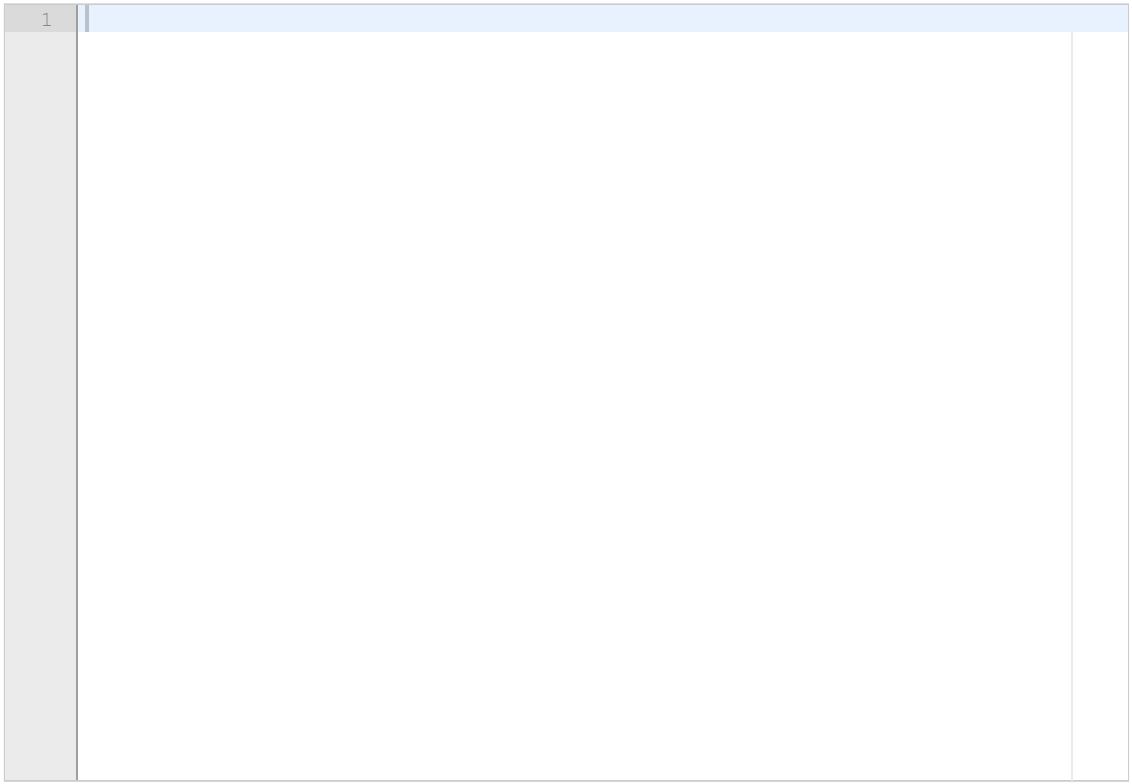
I en mappe som for eksempel kan hete **run** ligger det fem hundre filer som heter **run1**, **run2**, ... , **run500**. Dette er program som kan kjøres og som alle gir en streng på 5 tegn når de kjøres. Ett av programmen gir en streng på 5 tegn som inneholder substrengen "R3". Skriv et bash-script som systematisk kjører alle disse 500 programmene og finner ut hvilket av dem som gir et output som inneholder strengen "R3". Scriptet skal forvente å få et argument som er navnet på mappen som run-programmene ligger i, inkludert riktig path til mappen. Scriptet skal skrive ut den riktige strengen og navnet på filen som gir denne strengen. En kjøring kan for eksempel se slik ut:

```
kan@os:~/$ ./find.sh run
run/run377 gir strengen 5R3xG
```

Hvis det skulle være flere program som gir et output som inneholder strengen "R3", skal det tilsvarende skrives ut for hvert av dem (men i forrige oppgave er det kun ett program). Og hvis det ikke er noen slike strenger skal ingenting skrives ut. Hvis argumentet som gis ikke er en mappe skal programmet avsluttes med en feilmelding som vist i de to eksemplene under:

```
kan@os:~/$ ./find.sh run/run500
"run/run500" er ikke en mappe!
kan@os:~/$ ./find.sh
"" er ikke en mappe!
```

Skriv ditt svar her



Maks poeng: 30

10 Fredags-script

Øverst i hjemme-mappen til brukeren du er logget inn som på Linux-VM og som har brukernavn **kan** ligger det en mappe som heter **tusenfiler**. I denne mappen ligger det omtrent tusen filer som alle har forskjellige tidspunkt for når de sist ble endret. I denne mappen ligger det også et bash-script med navn **fredag.sh** som skriver ut navnet på alle filer i mappen som sist ble endret på en fredag. Hvis du kjører det i Linux-VM vil du se hvordan det virker.

Skriv en bash-kommando som kjører scriptet **fredag.sh** på en slik måte at kommandoen gir som output antall filer i mappen scriptet kjøres i som sist ble endret på en fredag. Kjør scriptet i mappen **tusenfiler** på Linux-VM og tell hvor mange sliker filer det er der og skriv både kommandoen og det totale antallet her (hvis du er i tidsnød og ikke rekker å kjøre kommandoen i Linux-VM, skriv bare inn kommandoen du ville kjørt nedenfor).

Skriv ditt svar her

Format

Words: 0

(*Hvis du ser på innholdet av **fredag.sh**, vil du se at dette er relativt komplisert script. I neste oppgave skal du selv skrive et PowerShell-script som gjør dette, da det er enklere å få til med PowerShells objekter.*)

Maks poeng: 10

11 PowerShell-script/oneliner

På Linux-VM kan det se slik ut når man lister ut en fil etter at man starter PowerShell med kommandoen pwsh:

```
kan@os:~$ pwsh
PowerShell 7.3.4
PS /home/kan> cd tusenfiler
PS /home/kan/tusenfiler> Get-ChildItem fredag.sh
    Directory: /home/kan/tusenfiler
UnixMode   User     Group         LastWriteTime     Size Name
--------   ----     -----         -------------     ---- ----
-rwxr-xr-x kan      users      05/23/2023 14:06     197 fredag.sh
PS /home/kan/tusenfiler>
```

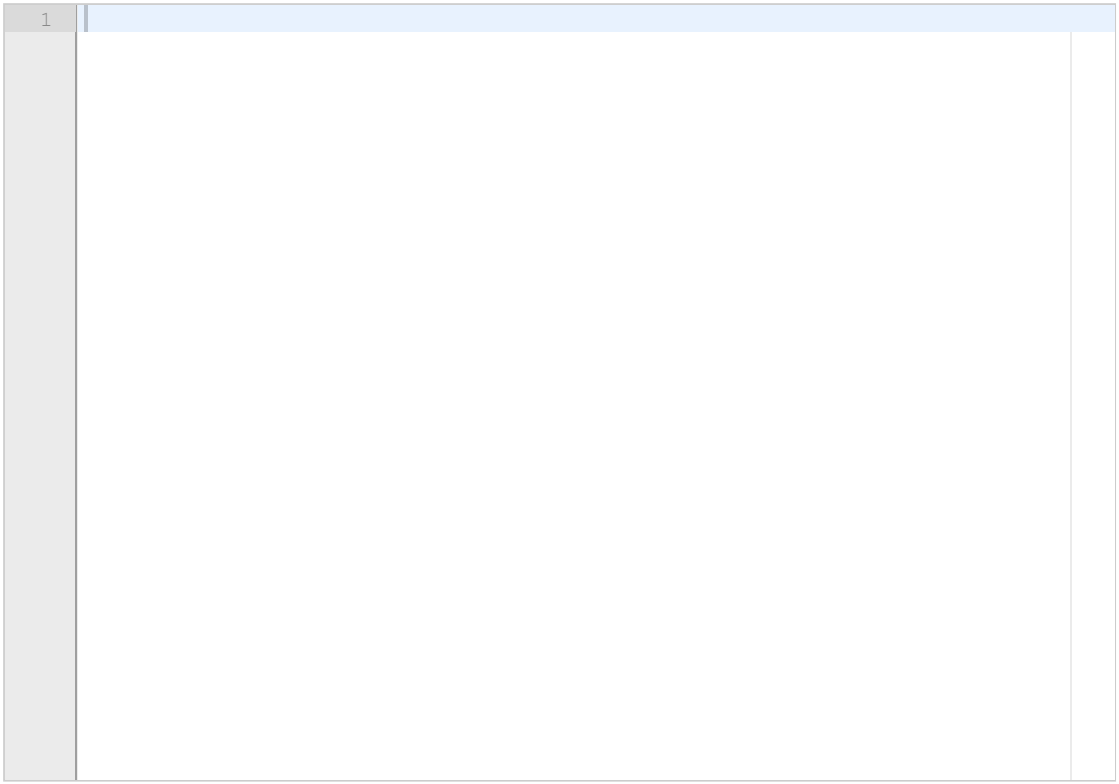
Vi så i forrige oppgave at det er relativt komplisert å finne frem til filer som sist har blitt endret på en gitt ukedag i bash. I PowerShell er dette vesentlig enklere fordi mens kommandoen **ls** i bash kun gir en tekststreng, returnerer den tilsvarende PowerShell kommandoen Get-ChildItem et objekt med mange properties som man kan trekke ut informasjon fra. For eksempel kan man få ut ukedagen på følgende måte:

```
PS /home/kan/tusenfiler> (Get-ChildItem fredag.sh).LastWriteTime.DayOfWeek
Tuesday
```

som viser at filen fredag.sh sist ble endret på en tirsdag (Tuesday). Skriv på grunnlag av dette en PowerShell oneliner eller et PowerShell script som teller opp antall filer i en mappe og skriver ut resultatet. Kjør det så i mappen tusenfiler i Linux-VM og tell opp antall filer som sist har blitt endret på en fredag ("Friday" i utskriften fra **Get-ChildItem**). Dette skal gi samme antall som når du kjørtet **fredag.sh** scriptet i bash i forrige oppgave og talte opp antallet filer.

Du kan om du ønsker det bruke jed, nano eller andre editorer til å skrive scriptet inne i pwsh-shellet eller utvikle en oneliner på pwsh kommando-linjen.

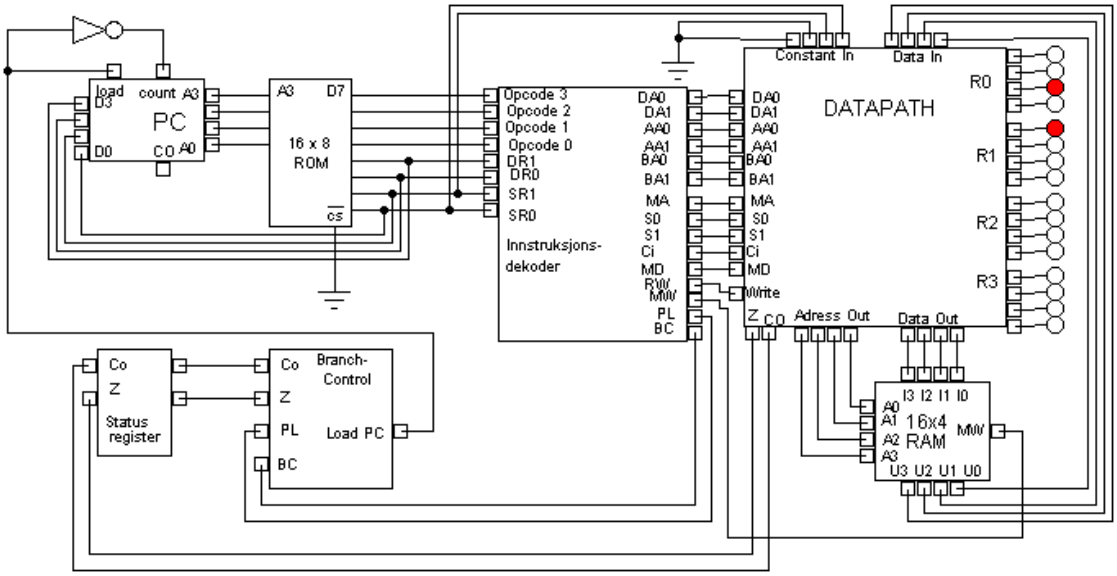
Skriv ditt svar her



Maks poeng: 30

12 Datamaskinarkitektur

Hvilken del av denne CPU'en inneholder ALU'en hvor innstruksjonene utføres?

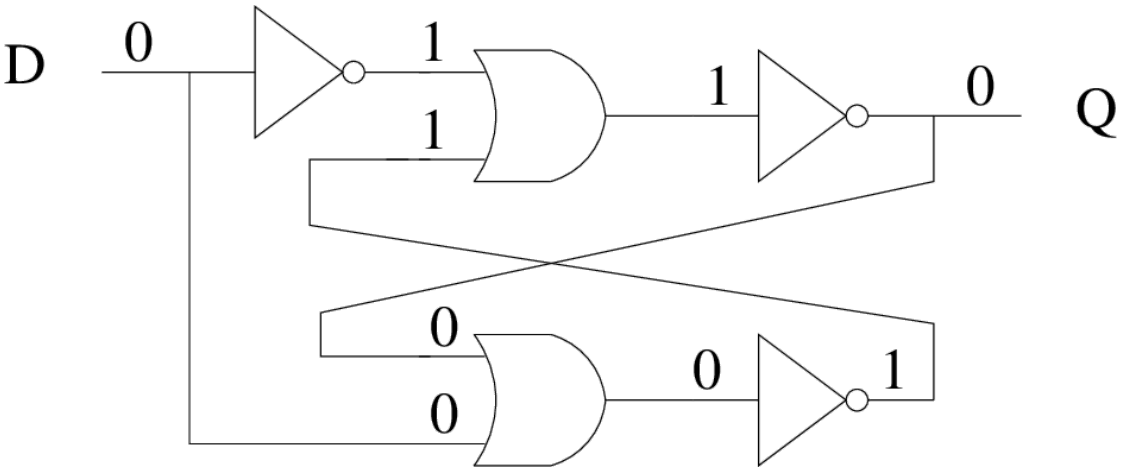


Velg ett alternativ

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

Maks poeng: 10

13 Konstruksjon av D-vippe



Figuren viser et forsøk på å lage en D-vippe som skal kunne brukes til å lagre ett bit i et register. Hva er problemet som gjør at denne konstruksjonen må utvides noe for å få en velfungerende D-vippe?

Velg ett alternativ:

- Q vil hele tiden flippe mellom 0 og 1 på grunn av at signalet sendes bakover i kretsen
- Siden det er tre NOT-porter vil D = 0 gi Q = 1
- Om man sender inn D = 1 vil det også resultere i Q = 0
- Man kan ikke velge å beholde verdien til Q selvom D endres
- Verdien til D kan ikke endres
- To enere inn i en OR-port gir kortslutning
- Verdien til Q kan ikke endres
- Kretsen kortslutter når ledningene krysser hverandre

Maks poeng: 10

14 Assembly

Hvilket tall returneres her av denne assembly-funksjonen?

```
.globl sum
# C-signatur:int sum ()
# 64 bit assembly
sum:
mov   $3, %rax
mov   $1, %rbx
add   %rax, %rbx
ret
```

Velg ett alternativ:

- 0
- 64
- 2
- 1
- 31
- 13
- 4
- 3

Maks poeng: 10

15 Datamaskinarkitektur

På Linux-VM (under ressurser) er det en mappe **/home/kan/cpp** hvor det ligger en C++-fil som heter **b.cpp**. Dette er et program som lager et stort array av usorterte tifeldige heltall mellom 0 og 256 og summerer ved hjelp av en if-test opp alle tallene i arrayet som er større enn 127.

Kompiler denne filen og mål hvor lang tid programmet bruker på å kjøre med:

```
kan@os:~/cpp$ g++ b.cpp
kan@os:~/cpp$ time ./a.out
```

I dette shellet er variabelen TIMESTAMP definert slik at kommandoen **time** kun vil rapportere "real time", det vil si hvor lang tid programmet reelt sett bruker på å kjøre. I starten av kjøringen skrives de 10 første tallen ut og du vil se at de kommer i tilfeldig rekkefølge. Åpne så filen i en editor og fjern de to kommentartegnene // fra følgende linje:

```
// sort(data, data + arraySize);
```

Når kommentaren fjernes fører det til at arrayet sorteres før if-testen kjøres. Kompiler programmet og ta på nytt tiden på hvor lang tid det bruker på å kjøre.

Ta utgangspunkt i det du har lært om datamaskinarkitektur og forklar kort hvorfor programmet går raskere i det andre tilfellet til tross for at programmet i begge tilfeller utfører nøyaktig like mange instruksjoner.

Skriv ditt svar her

Format

Words: 0

Maks poeng: 20

16 Serialisering av Java-tråder i Linux-VM

På Linux-VM (under ressurser) er det en mappe **/home/kan/java** hvor det ligger en fil **SynchThread.java**. Kompiler denne filen 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 til tross for at det er gjort et forsøk på å serialisere de to trådene som programmet kjører. Hva bør sluttresultatet for variabelen **saldo** bli om trådene er serialisert slik at de ikke ødelegger for hverandre? Hva er feil ved forsøket på å serialisere trådene og hvordan kan denne feilen rettes opp? Forklar kort. Rett opp feilen i **SynchThread.java**, kompiler og kjør det korrigerte programmet på nytt, sjekk at du nå får riktig svar og skriv svaret her.

Skriv ditt svar her

Format

Words: 0

Maks poeng: 30

17 Modusbit

Hva er en prosessors modusbit?
Velg ett alternativ:

- Et bit som avgjør om det skal utføres en context switch
- Et bit som kan begrense aksess til minne og instruksjoner
- Et bit som settes når det kommer et interrupt
- Et bit som endrer CPU'ens klokkefrekvens
- Et bit som gir et program root-rettigheter selvom det kjøres av en vanlig bruker
- Et bit som sier om prosessoren er aktiv eller ikke

Maks poeng: 10

18 CPU-avhengige prosesser

Hvis en 100% CPU-avhengig prosess bruker 60 sekunder på å fullføres på en CPU-kjerne når den kjører alene, hvor mange sekunder bruker 4 slike prosesser når de kjører på 3 CPU-kjerner?

Velg ett alternativ:

- 100
- 90
- 60
- 150
- 80
- 120
- 110
- 180
- 70
- 240

Maks poeng: 10

19 Prosesser

På Linux-VM (under ressurser) er det en mappe **/home/kan/regn** hvor det ligger et bash script som heter **regn**. Dette er et CPU intensivt program som i en lang for-løkke regner ut en sum. I mappen ligger det også to bash script **run1** og **run2** som på litt forskjellige måter kjører scriptet **regn** to ganger.

I dette shellet er variabelen TIMESTAMP definert slik at kommandoen **time** kun vil rapportere "real time", det vil si hvor lang tid programmet reelt sett bruker på å kjøre. Ta tiden på de to run-scriptene på følgende måte

```
kan@os:~/regn$ time ./run1
kan@os:~/regn$ time ./run2
```

og forklar kort hvorfor det ene bruker omtrent halvparten så lang tid som det andre. Kjør scriptene på nytt med følgende kommandoer

```
kan@os:~/regn$ time taskset -c 0 ./run1
kan@os:~/regn$ time taskset -c 0 ./run2
```

og forklar kort hvorfor de nå bruker omtrent like lang tid.

Skriv ditt svar her

Format

Words: 0

Maks poeng: 20

20 Internminne

I et C-program er det et integer (int) array som har 1024*1024 elementer. Hvor mange byte utgjør dette arrayet?

Velg ett alternativ:

- 256 KiB
- 512 KiB
- 1 MiB
- 4 MiB
- 8 MiB
- 16 MiB
- 32 MiB
- 1 GiB
- 4 Gib
- 8 GiB
- 16 GiB
- 32 GiB

Maks poeng: 10

21 Internminne

Hvorfor kan ikke MMU implementeres i software og styres av OS?
Velg ett alternativ:

- Fordi minne-aksess da ville ha tatt alt for lang tid
- Fordi også vanlige brukerprosesser må kunne aksessere RAM
- Fordi det ikke ville være plass i RAM til hele page-tabellen
- Fordi RAM må kontrolleres av hardware, OS kan kun styre CPU
- Fordi det da ikke ville være mulig å bruke cache
- Fordi man da ikke kunne finne mappingen for en virtuell adresse siden den ligger i fysisk RAM

Maks poeng: 10