

Ukesoppgaver 4 – Binære tall og full adder; filbehandling og filrettigheter

1. Utfør addisjonen $5 + 5 = 10$ med binære tall for hånd ved å bruke den samme addisjonsmetoden som man bruker når man legger sammen desimaltall med penn og papir.

ukesoppgaver ④

$$5 + 5 = 10$$

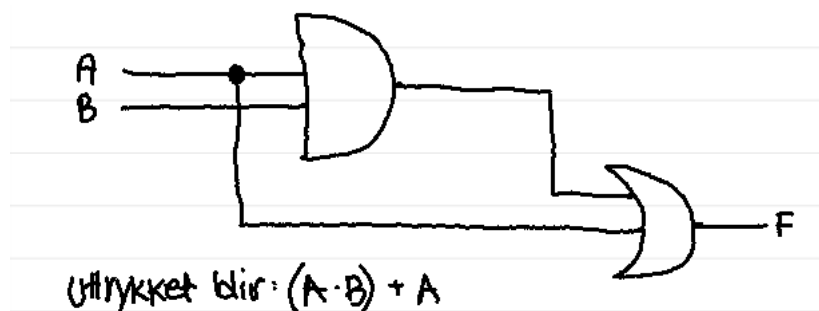
5 i binært tall er 101

$$\begin{array}{r} 101 \\ + 101 \\ \hline = 1010 \end{array}$$

2. sannhetstabell

C X Y	X	Y	Z	S	C
$+ Y$	0	0	0	0	0
$= S$	0	0	1	1	0
	0	1	0	1	0
	0	1	1	0	1
	1	0	0	1	0
	1	0	1	0	1
	1	1	0	0	1
	1	1	1	1	1

3. hva blir uttrykket?



4. Uttrykket $(A * B) + A$. Ved store verdier vil uansett A dominere. Hvis vi trekker felles leddet A utenfor parantesen blir $B * 1$ inni parantesen. Uttrykket kan dermed forkortes til A.

5. gjøre

6. gjøre

7. Tegn en krets med AND, OR og NOT-porter som gir funksjonen $F = A + A \cdot B$.
Samme som i bildet fra oppgave 3.

8. Hva er path? Finn ut hva som er full path til programmet som startes når du gir kommandoen diff og les manualsiden for denne kommandoen.

Path er en variable som instruerer et linux system til hvilke kataloger å søke i for kjørbare filer. Path variabelen lar en bruker kjøre en kommando uten å spesifisere en vei/bane/path. Diff står for difference og er en variabel som viser forskjeller i filer oppgitt, linje for linje. I motsetning til cmp og comm som vil fortelle om hva som trengs for å gjøre de to filene identiske.

9. Gå til katalogen /usr/bin ved å bruke absolutt path.

Cd /usr/bin

10. Gå til hjemmekatalogen din. Gå deretter til katalogen /usr/bin ved å bruke relativ path.

Man må først inn i rotkatalogen ellers vil det ikke fungere og bruke relativ path direkte.

```
$ cd /  
$ ls  
$ cd usr/bin  
$ pwd  
$ /usr/bin
```

11. Kommandoen diff sammenlikner filene for hver linje, lister forskjellene i filene og forteller hva som må til for å få de identiske.

```
> 1a 2a 3a 4b 5a
[s364520@data2500:~$ cat fa.txt
1a 2a 3a 4a 5a
[s364520@data2500:~$ cat fb.txt
1a 2a 3a 4b 5a
[s364520@data2500:~$ diff fa.txt fb.txt
1c1
< 1a 2a 3a 4a 5a
---
> 1a 2a 3a 4b 5a
[s364520@data2500:~$
```

C står for change.

12. Lag en katalog med navn 'test' i oblig1-katalogen du lagde i forrige uke. Kopier deretter katalogen oblig1 med alle filer og underkataloger til katalogen oblig2 (oblig2 skal ikke eksistere fra før). Sjekk at alt ble kopiert til den nye katalogen oblig2. Hva skjer om du utfører kopieringen av oblig1 til oblig2 en gang til (med samme kommando)?

```
cp -r oblig1 oblig2
```

Hele oblig1 mappen vil bli kopieres inn i oblig 2 hvis det kopieres en gang til fordi nå finnes det en mappe som heter oblig2. Da oppretter den ikke en ny mappe.

13. Først lagde jeg to mapper tex og oblig. Så flyttet jeg oblig under tex. Så lagde jeg en ny mappe oblig. Så flyttet jeg tex (som allerede har oblig under seg) under den nye oblig. Da får jeg slik struktur som vist under.

```
[s364520@data2500:~$ mkdir oppgave13
[s364520@data2500:~$ cd oppgave13
[s364520@data2500:~/oppgave13$ mkdir tex oblig
[s364520@data2500:~/oppgave13$ mv oblig tex
[s364520@data2500:~/oppgave13$ mkdir oblig
[s364520@data2500:~/oppgave13$ mv tex oblig
[s364520@data2500:~/oppgave13$ tree
.
|-- oblig
|   |-- tex
|   |   |-- oblig
3 directories, 0 files
[s364520@data2500:~/oppgave13$
```

14. `cd ~`, `cd` og `cd /home/s364520`

```
s364520@data2500:~$ cd  
[s364520@data2500:~$ cd ~  
[s364520@data2500:~$ cd /  
[s364520@data2500:/$ cd /home/s364520  
s364520@data2500:~$
```

15. Skriv kommandoen `echo bla bla bla > newfile`, og bruk `more` for å se på filens innhold. Kan du forklare hva som har skjedd? Dette kalles redirection.

Vi setter inn innhold i `newfile` og `echo` kommandoen blir redirected til tekstfila. Da skjer ikke det på skjermen.

```
[s364520@data2500:~$ echo bla bla bla > newfile  
[s364520@data2500:~$ ls -l  
total 20  
-rw-r--r-- 1 s364520 20364520 15 Jan 25 22:56 fa.txt  
-rw-r--r-- 1 s364520 20364520 15 Jan 25 22:56 fb.txt  
-rw-r--r-- 1 s364520 20364520 12 Jan 25 23:49 newfile  
drwxr-xr-x 3 s364520 20364520 4096 Jan 25 23:24 oppgave13  
drwxr-xr-x 2 s364520 20364520 4096 Jan 16 22:04 oppgave15  
[s364520@data2500:~$ more newfile  
bla bla bla  
s364520@data2500:~$
```

MANGLER 16-23

21. Lag først en fil `hemmelig.txt`. Utfør en Linux-kommando som setter filrettighetene for filen `hemmelig.txt` slik at eieren av filen (du) kun kan lese den, mens alle andre ikke har noen rettigheter.

```
[s364520@data2500:~$ touch hemmelig.txt  
[s364520@data2500:~$ chmod 400 hemmelig.txt  
[s364520@data2500:~$ ls -l  
total 20  
-rw-r--r-- 1 s364520 20364520 15 Jan 25 22:56 fa.txt  
-rw-r--r-- 1 s364520 20364520 15 Jan 25 22:56 fb.txt  
-r----- 1 s364520 20364520 0 Jan 26 18:00 hemmelig.txt  
-rw-r--r-- 1 s364520 20364520 12 Jan 25 23:49 newfile  
drwxr-xr-x 3 s364520 20364520 4096 Jan 25 23:24 oppgave13  
drwxr-xr-x 2 s364520 20364520 4096 Jan 16 22:04 oppgave15  
s364520@data2500:~$
```

Her, markert med rødt, kan man se rettighetene til fila og kun jeg som admin kan lese den, mens ingen av de andre gruppene har noen rettigheter. Den første bindestreken understreker kun at typen er en fil.

22. Lag først en fil `fil.txt`. Utfør en Linux-kommando som setter filrettighetene for fila `fil.txt` slik at eieren av fila (du) har alle rettigheter, medlemmer av filens gruppe har alle rettigheter untatt å skrive til fila, og alle andre kun kan lese den.

```
[s364520@data2500:~$ touch fil.txt
[s364520@data2500:~$ chmod 754 fil.txt
[s364520@data2500:~$ ls -l
total 20
-rw-r--r-- 1 s364520 20364520 15 Jan 25 22:56 fa.txt
-rw-r--r-- 1 s364520 20364520 15 Jan 25 22:56 fb.txt
-rwxr-xr-- 1 s364520 20364520 0 Jan 26 18:07 fil.txt
-r----- 1 s364520 20364520 0 Jan 26 18:00 hemmelig.txt
-rw-r--r-- 1 s364520 20364520 12 Jan 25 23:49 newfile
drwxr-xr-x 3 s364520 20364520 4096 Jan 25 23:24 oppgave13
drwxr-xr-x 2 s364520 20364520 4096 Jan 16 22:04 oppgave15
s364520@data2500:~$
```

23.

24. Hva blir rettighetene til en ny fil du lager med touch? Blir rettighetene de samme om du bruker en editor til å lage en ny fil? Bruk kommandoen `umask` til å sørge for at nye filer som lages kun kan leses og skrives til av eieren, mens alle andre ikke får noen rettigheter.

Jeg vil få de samme rettighetene. Det utgjør ingen forskjell om det er laget med touch eller editor.

Umask 0077 gjør at kun eier kan lese og skrive imens de andre gruppene inne får noen rettigheter.

28-33 må gjøres