Ukeoppgaver Maskinvare
(Koder møter kretser)

(1) Bruk UiO GPT til å utforske følgende begreper:
   a. Minnekoherens – Hvilke minneelement er dette viktig for og hvorfor?
   b. Hva er forskjellen på Cache-Miss og RAM-miss?
   c. Hva er en CPU og en ALU? Hva er hovedforskjellene mellom dem?
   d. Hvordan regner man ut frekvens for en CPU?

Er du enig i alle svarene som UiO GPT gir?

(2) Beregning av Cache-miss:
Anta at en prosessor vil prosessere en instruksjon per klokkesykel og skal totalt prosessere 1.000.000 instruksjoner som typisk vil ha behov for å hente data fra minne for 60% av instruksjonene, hvor det uheldigvis vil være 50% cache-miss. Anta videre at det typisk vil være en penalty (økt forsinkelse) for cache-miss på 10 klokkesykler. Hva lang mange klokkesykler vil det hele ta?

1.000.000 instruksjoner = 1.000.000 klokkesykler
60% minneaksess = 600.000 instruksjoner
som vil ha 50% cache-miss = 300.000 instruksjoner
som vil ta 10 sykler lenger tid = 3.000.000 instruksjoner

Totalt 1.000.000 + 3.000.000 = 4.000.000 klokkesykler.

(3) NØTTNØTT! CPUen trenger å prosessere data som ikke finnes i register, cache eller RAM, men ute på harddisken. Gitt at hvert ledd av minnehierarkiet må lagre dataen før de kan sende den videre. Hvor lang tid vil det ta fra harddisken til CPUen?

LAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAANG tid.....

Veldig forenklet så kan man anta at hastigheten fra register til cache blir 10 ganger forsinket. Cache til RAM 50 ganger forsinket og RAM til harddisk med 10 ganger. Hvis man må vente til at det er lagret før man leser da blir det dobbelt av hvert ledd som gir da

Lese fra
register: 100 ps
cache: 1 000 ps
RAM: 50 000 ps
SSD: 500 000 ps

Svaret blir da:
(100 + 1.000 + 50.000) *2 + 500.000 = 602.200 ps
Altså
Ca 6.000 ganger så lang tid som det vil da lese ut fra et register.

Hvis man legger til grunn at en prosessor kan gjennomføre en operasjon pr klokkeperiode, så taper man 6.000 operasjoner for å hente noe ute på harddisken.