



Faglig kontakt under eksamen:
Institutt for datateknikk og informasjonsvitenskap
Heri Ramampiaro, 73593440

**MIDTEMESTEREKSAMEN I EMNE
IT1101 INFORMATIKK BASISFAG
2. Forsøk**

Tirsdag 31. oktober 2006.
KL. 12.15 – 14.00

Hjelpemidler: Ingen trykte eller håndskrevne tillatt. **INGEN** kalkulator er tillatt.

NB: På alle oppgavene er det ett riktig svar. *Dersom du anser flere av alternativene som riktige, velg det som er mest riktig.* Hvert riktig svar gir **2 poeng**, flere avkrysninger eller feilsvar gir **-1 poeng (trekk)**, blankt svar gir **0 poeng**. Stort sett alle svarene teller likt bortsett fra der dette er særskilt spesifisert. Maksimalt oppnåelig poeng er 40 (100%).

Studentnr.: _____

Oppgave I

- Kun ett av følgende utsagn er riktig om ”*toerskomplementnotasjon*”. Kryss av hvilket.
☐ A: I 4-bits toerskomplementnotasjon vil tallene **-8** og **8** begge være representert.
☐ B: For å konvertere et (spesifikt) positivt tall til negativt er det nok å sette bit lengst til venstre (leftmost bit) til 1.
☐ C: Toerskomplementnotasjon brukes alltid sammen med flyttallnotasjon
☐ D: Addisjon av 2 toerskomplementverdier kan i noen tilfelle i et 4-bits toers-komplement system gi overflow men ikke når vi legger sammen (med addisjon) 1101_{tkn} og 1110_{tkn} .
- Hva er 8-bits toers komplement representasjon til tallet -180 (10-tallsystemet)?
☐ A: 11110110_{tkn}
☐ B: 01110110_{tkn}
☐ C: -01110110_{tkn}
☐ D: 10001010_{tkn}

3. Hvilket av utsagnene nedenfor er *sant*?
- ☐ A: I en 8-bits flyttallsnotasjon er det satt av 4 bits til å ta vare på *råtallene* (mantissa), resten er til å fortelle hvor komma er, dvs. kommaposisjon eller *eksponent*.
 - ☐ B: *Eksponenten* i en flytallsnotasjon kan representeres enten vha. av *eksessnotasjon* eller *toerskomplementnotasjon*.
 - ☐ C: Avrundingsfeil i flyttallsnotasjon oppstår ofte når det er for liten plass til råtallene (mantissa).
 - ☐ D: En økning i antall bits i eksponenten (i flyttallsnotasjon) medfører at avrundingsfeil ved addisjon av flyttall vil oppstå sjeldnere.
4. Hvilket av følgende er feil?
- ☐ A: $-4^{3/8}$ er umulig å konvertere til binært tall vha en 8-bits flyttallsnotasjon pga avrundingsfeil ("truncation error")
 - ☐ B: $-4^{3/8}$ er lik 11000110 i 8-bits flyttallsnotasjon
 - ☐ C: $-5^{1/2}$ er lik 11111011 i 8-bits flyttallsnotasjon
 - ☐ D: $2\frac{1}{2} + 1/8 + 1/8$ kan gi oss avrundningsfeil dersom vi velger "feil" rekkefølge på addisjonen
5. Datakomprimering – Hvilket av følgende utsagn er *sant*?
- ☐ A: Huffman koder er et godt eksempel på *frekvens-avhengig enkoding* ("frequency-dependent encoding"), men den er ikke "*lossless*"
 - ☐ B: *Oppslagsverk enkoding* ("Dictionary encoding") er basert på oppslag og derfor aldri kan være en "*lossy*" enkoding.
 - ☐ C: GIF er komprimering av bilder som bruker teknikker bygd på *LZW*.
 - ☐ D: LZ77 er et eksempel på en komprimeringsmetode som kan kalles "*relative encoding*"
6. Komprimering av "xyx xyz yxx xyz" kan gjøres som følgende. Velg hva som er rett.
- ☐ A: xyx xyz yxx (8,2,z) hvis vi benytter LZ77
 - ☐ B: 1213124321135 hvis vi bruker en hvilken som helst oppslagsverk enkoding
 - ☐ C: 1213125321134 ved hjelp av LZW
 - ☐ D: 1213(4,2,4)321135 ved hjelp av LZ77
7. Følgende utsagn handler om håndtering av kommunikasjonsfeil. Velg hvilket er *usant*:
- ☐ A: "Hamming distance" er et eksempel på teknikker som kan brukes til å korrigere datafeil i forbindelse med kommunikasjonsfeil.
 - ☐ B: For å oppdage kommunikasjonsfeil kan vi bruke paritets bit. Både "odde paritetskode" (odd parity) og "partalls paritetskode" (even parity) kan benyttes til dette formålet.
 - ☐ C: "Huffman kode" er kun en komprimeringsmetode og derfor ikke er ment for å håndtere kommunikasjonsfeil
 - ☐ D: "CRC" (Cyclic redundancy checks) er en form for feilkorrigerende kode (error-correcting code)

Oppgave II

- Hvilet av følgende utsagn er **IKKE** riktig?
 - ☐ A: Med CISC prosessorer er det ikke (aldri) mulig å gjennomføre en spesifikk kompleks instruksjon uten å måtte kombinere to eller flere mindre komplekse instruksjoner fra et begrenset instruksjonssett
 - ☐ B: En Intel Pentium CPU er et eksempel på CISC prosessorer mens Apple sin gamle PowerPC G5 og G5 CPUer er eksempler på RISC prosessorer
 - ☐ C: Det finnes 3 forskjellige gruppe CPU instruksjoner, uavhengig om prosessoren er RISC eller CISC: (1) gruppe for data overføring, (2) aritmetisk/logisk gruppe og (3) kontroll gruppe.
 - ☐ D: AppC-instruksjonene (som i vedlegg 1) er et komplett sett med instruksjoner som ville være nok for en RISC CPU
- Hva består en CPU-arkitektur av?
 - ☐ A: Instruksjonsdekoder og minnehåndterer (memory manager)
 - ☐ B: Aritmetisk/logisk enhet (ALU) og kontroll enhet.
 - ☐ C: Programteller, instruksjonsminne, og hovedminne
 - ☐ D: Flere registre, instruksjonslager og bus
- Gitt følgende program lagret i hovedminnet der programtelleren på forhånd er satt til A6.

Adresse	Innhold
A6	25
A7	06
A8	24
A9	F2
AA	53
AB	54
AC	C0
AD	00

- Hva er innhold i register 3 når programmet avslutter? Se vedlegg 1.
- ☐ A: Oppgaven er uløselig
 - ☐ B: FF (hex) og 11111111 (bin)
 - ☐ C: F8 (hex) og 11110000 (bin)
 - ☐ D: AD (hex) og 10101101 (bin)
- Hvilket av følgende utsagn om er **sant**?
 - ☐ A: Dersom en prosessor benytter seg av såkalt *pipelining* kan ALU utføre/kjøre 3 instruksjoner samtidig
 - ☐ B: ALU har som hovedoppgave å lagre flyttall samt utføre logiske operasjoner som OR, XOR og AND
 - ☐ C: For at ALU skal kunne behandle verdier som ligger i hovedminnet må verdiene først lastes inn i registre
 - ☐ D: CPU'en benytter såkalte *semaforer* for å håndtere konflikter i forbindelse med instruksjonskjøring

Oppgave III

1. Hvilket av følgende utgang er passer **best** om et operativsystem:
☐ A: Operativsystemets roller er å sørge for at det ikke oppstår minneallokeringsproblemer vha en *minnehåndterer* (memory manager)
☐ B: Operativsystemets roller er å sørge for at alle prosessene har tilgang til filer og filinformasjoner vha *filhåndterer* (file manager)
☐ C: Operativsystemets roller er å sørge for at ingen prosesser er i konflikt med hverandre vha en *konflikthåndterer* (conflict manager)
☐ D: Både A og B er riktig.
2. Gitt følgende utsagn: "En semafor er et redskap som operativsystemet benytter som en sikker måte å koordinere prosesskjøringer på". Avgjør hvilket av følgende passer **best** til dette utsagnet.
☐ A: Utsagnet er ikke sant da semaforer er bare en form for flagg som *kan forstyrres* under setting (seting) og fjerning (clearing).
☐ B: Utsagnet er ikke sant fordi semaforer benyttes til å håndtere *vranglås*
☐ C: Utsagnet er sant fordi semaforer implementerer såkalte "*test-and-set*" instruksjon
☐ D: Utsagnet er sant fordi semaforer bestemmer når en prosess skal slutte å kjøre slik at andre prosesser også får lov til å bli ferdig.
3. Avgjør når begrepet "*spooling*" er mest aktuelt:
☐ A: I operativsystemsammenheng, i forbindelse med håndtering av vranglås
☐ B: I prosesskjøringssammenheng, når prosessene bes om å kjøre på nytt
☐ C: I forbindelse med at operativsystemet aksesserer en harddisk
☐ D: I forbindelse med nullstilling av semafor-verdier

Oppgave IV

1. Hvilken av følgende påstand er **feil**?
☐ A: Internett er et netteverk av nettverk
☐ B: To nettverk av samme topologi kan som hovedregel koples sammen ved hjelp av en router
☐ C: Dersom to nettverk er koblet sammen vha en bro vil all kommunikasjon (eksempelvis meldingssending mellom enhetene eller "hosts") i det ene nettverket være synlig i det andre nettverket til enhver i tid
☐ D: Et nettverk med irregulær topologi er ikke nødvendigvis mer pålitelig enn et nettverk med stjernetopologi
2. Hvilket av følgende utsagn er **sant**?
☐ A: At noen er autentifisert betyr at de har rett/lov til å aksessere informasjon
☐ B: Å komprimere og kryptere betyr det samme
☐ C: Hvilken vei (gjennom hvilke maskiner) en melding sendes over Internett avgjøres i nettverkslaget
☐ D: Internett ble utviklet av Microsoft på oppdrag fra det amerikanske forsvaret i 1992.
3. Gitt noen påstår dette overfor deg: "IP-adresser med 32 bits lengde kan vi ha et *ubegrenset* antall unike adresser". Hvilket av følgende passer for deg å svaret?
☐ A: Dette er selvfølgelig bare tull fordi vi kan "kun" ha 1024 (2^2) IP-adresser
☐ B: Dette er sant fordi 32 bits gir så mange kombinasjoner at det er nærmest umulig å beregne
☐ C: Dette er sant fordi vi vil få 10^{32} antall adresser, hvilket gir et tilnærmet ubegrenset antall unike adresser
☐ D: Dette er ikke sant fordi det riktige antallet vil være 4 294 967 296 (2^{32}), noe som ikke er ubegrenset.

Studentnr: _____

4. Du har en sortert liste med $N = 1\,048\,576$ elementer. Du skal avgjøre hvor mange sammenlikninger du trenger dersom du skal prøve å finne noe som ikke ligger i lista. Hvilket av følgende utsagn er da *sant*?
- ☐ A: Lista er for stor derfor er oppgaven umulig basert på de søkealgoritmene vi har lært i faget.
 - ☐ B: Bruker vi binærsøk trenger vi alltid flere sammenligninger enn om vi skulle brukt sekvensielt søk
 - ☐ C: Bruker vi sekvensielt søk trenger vi $N/2$ dvs 524 288 sammenligninger
 - ☐ D: Bruker vi binærsøk trenger vi ikke mer enn 20 sammenlikninger.
5. Gitt denne listen: **Arne, Birger, Bjørn, Danner, Ding, Fredrik, Frode, Gunnar, Heidi, Karl, Kjell, Petter, Sigrid, Svein, Åge**. Vi skal søke etter **Bjørn**. Hvilken algoritme vil gi færrest sammenligninger (elementer sjekket)?
- ☐ A: Sekvensielt søk
 - ☐ B: Like mange sammenligninger ved binært og sekvensielt søk
 - ☐ C: Umulig å avgjøre
 - ☐ D: Binærsøk
6. Hvilken av følgende påstand mener du er *feil*?
- ☐ A: JSP er basert på en objektorientert programmeringsspråkparadigme
 - ☐ B: Java, SmallTalk og C++ hører til samme språkparadigme
 - ☐ C: C og C++ er to beslektede programmeringsspråk men hører egentlig til i to forskjellige paradigmer
 - ☐ D: Basic og Fortran og Lisp tilhører alle i funksjonelle programmeringsspråkparadigmet

Studentnr: _____

Appendix C maskinspråk

Op-code	Operand	Beskrivelse
1	RXY	LOAD: Legg innholdet i adresse XY inn i register R.
2	RXY	LOAD: Legg verdien XY inn i register R.
3	RXY	STORE: Legg innholdet i register R inn i adressen XY.
4	ORS	MOVE: Flytt innholdet i register R til register S.
5	RST	ADD: Legg sammen innholdet i registerene S og T og legg resultatet i R. Tallene i S og T er heltall representert i 2'ers komplement notasjon.
6	RST	ADD: Legg sammen innholdet i registerene S og T og legg resultatet i R. Tallene i S og T er flyttall representert i flyttallsnotasjon.
7	RST	OR: Gjør en OR på bitmønstrene i S og T og legg resultatet i R.
8	RST	AND: Gjør en AND på bitmønstrene i S og T og legg resultatet i R.
9	RST	EXCLUSIVE OR: Gjør en XOR på bitmønstrene i S og T og legg resultatet i R.
A	R0X	ROTATE: Roter bitmønster i R en bit til høyre X ganger.
B	RXY	JUMP: Hopp til den instruksjonen som ligger i adresse XY hvis innholdet i R er lik bitmønsteret som ligger i register 0. Hvis ikke fortsett den normale sekvensielle utføringen.
C	000	HALT: Avslutt

VEDLEGG 2 av 2: 3-bit excess notasjon

Bit pattern	Value represented
111	3
110	2
101	1
100	0
011	-1
010	-2
001	-3
000	-4

Studentnr: _____

KLADD – riv av