

Denne kolonne er
forbeholdt sensor.Oppgave 1)(og/eller ~~app~~ applika-
sionene)

a) Er grensesnitt mellom brukeren og maskinvaren. Tildeler ressurser ~~til~~ til maskinvare, styrer og kontrollerer prosesser, sørger for ut-data (skjerm, printer o.l.), tar imot inn-data (f.eks. mus, tastatur), knytter maskiner til andre maskiner, tildeler ressurser og legger premisser for applikasjoner.

b) Del av harddisken som brukes som RAM. Øker maskinens ledige minne, men er ~~læst~~ tregere enn vanlig RAM.

Swapping kan brukes i flere sammenhenger:
Hot-swapping: Bytte ut harddisker mens maskinen kjøres, ~~for~~ noe man kan med SATA-disker og diskene i NAS og enkelte RAID-løsninger.

Swapping: Brukes om linux-partisjoner. En swap-disk er en virtuell disk.

Denne kolonne er
 forbeholdt sensor.

forts oppgave 1)

c) To måter:

1. Lossy: F. eks. musikk, MP3-fil. Områder i filen fjernes uten at det gir et særlig hørbart resultat. For MP3-filer betyr det at ikke-hørbare frekvenser tas vekk (høye + veldig lave) og filstørrelsen krymper mye. ~~For bilder:~~ For bilder: f. eks. JPG. Her fjernes ~~noe~~ noe av dybdeskarphet og fargeinfo, ~~alt~~ alt eller komprimeringsgrad. Felles er uansett at dess mer man komprimerer, dess mer fjernes. Og det som er fjernet kan ikke erstattes hvis man skulle ønske å gjendanne filen til originalstand.

2. Lossless: F. eks. bilde.

En bilde-fil består av nuller og ener, og en ~~linje~~ linje kan f. eks. se slik ut:
 00101101001100100000000000000000000000110110

Når man komprimerer ~~et~~ vil man fjerne all repeterende info eller "tom" kode og erstatte det med info om hvor som ble fjernet.

Eks.: 001011010011001-20 null'er - 110110.

Fjerne redundans.

Dermed kan man gjenopprette filen om man skulle ønske det, eller beholde den

Denne kolonne er
forbeholdt sensor.

forts. 1 c)

~~stør~~ komprimert uten tap av data.

Komprimeringen blir mindre, men altså
tapfri.

Ekse. på tapfri ~~et~~ bildefil : PNG-fil

— " — musikkfil : APE, FLAC

D) BIOS - Basic Input Output System.

Styringsystem som ligger i en ROM-brikke
på hovedkortet.

Hovedoppgaver :

- CPU søker ved oppstart etter et bestemt minneområde i ROM, BIOS starter.
- BIOS kjører POST (Power on Self Test) som sjekker at nødvendige enheter er der, hva enheter som finnes og om de fungerer som forventet.
- Starter bootstrap.
- Bootstrap starter loader ~~som~~ som i sin tur starter ~~et~~ operativsystemet.

Denne kolonne er
forbeholdt sensor.

Forts. oppg. 1

e)

A) Full backup tar kopi av alle data på den disken eller dataområdet som backup-jobben definerer. I Windows-systemer får filene den nye filattributtet "~~ATA~~" "A" som viser at de er arkivert.

B) Inkrementell backup tar ^{kopi} backup av alle data som er endret eller siste fulle backup, eller siste inkrementelle.

C) Differensiell backup tar kopi av alle data som er endret eller siste fulle ~~(eller inkrementelle)~~ backup.

Følgene er :

Ved backup går inkrementell backup raskest siden det kanskje bare er få data som er endret siden siste backup. Mens ~~if~~ differensiell tar lengre tid siden alle ~~st.~~ data som er endret siden fulle backup skal kopieres.

Men ved gjenopprettning tar differensiell backup ~~lengre~~ kortest tid siden man kun trenger siste fulle + siste diff-sett. Ved inkrementell trenger man siste fulle + alle inkrem.-sett. Dus. mye lengre tid.

Denne kolonne er
forbeholdt sensor.

Oppgave 2

a) For personregister uten sensitive opplysninger er det nok med melding til datatilsynet. For sensitive opplysninger må datatilsynet gi konsesjon. Ved ~~til~~ tilstilfelle søker man ber man datatilsynet om å vurdere om ~~kravene~~ kravene for konsesjon er tilstede eller ikke.

Skal man registrere sensitive opplysninger må de registrerte gi sin tilslutning til at data registreres. Unntaket gjelder for enkelte typer statlige registre.

B) ~~Den som~~ Opphavsbeskyttet materiale er ethvert materiale av kunstnerisk art, herunder tekst, bilder, musikk, video, dataprogram. Opphavsmann og rettighetshaver er den/ de som har laget materialet, med mindre opphavsmann har overdratt rettighetene til tredjepart. Opphavsmann har rettighetene i 70 år etter sin død. Er deler av et verk rettighetsbeskyttet og noen deler fritt, er hele verket samlet sett beskyttet til sist beskyttelse er gått ut på dato.

%

Denne kolonne er
forbeholdt sensor.

forts. oppgave 2 a)

Det er ett unntak fra loven og det gjelder utvikling av dataprogrammer i ~~de~~ bedrift hvor utvikleren er ansatt. Her er det bedriften som har opphavsretten.

Statlige offentlige dokumenter er ikke opphavsbeskyttet.

c) For mange er internett et hav av informasjon som er fritt tilgjengelig, uansett hva. Men alt som legges ut eller produseres for internett er underlagt de samme lover for opphavsrettigheter.

Tekst på vg.no kan f.eks. ikke brukes uten tillatelse fra VG, og man kan heller ikke lenke til stoff uten å oppgi hvem stoffet er ~~hva~~ hentet fra og uten at tillatelse er gitt.

På grunn av at det er så enkelt å kopiere, manipulere og spre elektroniske data blir det veldig utfordrende å spore, kontrollere og finne ut av hvem opphavsrettigheter.

Det er dessuten vanskelig å håndheve opphavsretten fordi data så lett kan endres og kopieres, uten at man enkelt kan si hvem som var egentlig eier. F.eks.: Dersom noen lager en per nettside med et flott design : %

Denne kolonne er
forbeholdt sensor.

forts. oppgave 2 c).

Hvordan kan man finne informasjon om personen som laget den dersom man ønsker å ~~finne~~ bruke samme design eller kildekode selv? Hvordan skal man fastsette godtgjørelse og overføring av godtgjørelser? (Personen sitter kanskje i Kina). Og hvem skal bestemme om man bare har blitt inspirert av det designet, eller om man har kopiert det?

(Kunne ha skrevet i dagens om bare dette temaet, men stopper her!)

Denne kolonne er forbeholdt sensor.

Oppgave 3

a) I Kravspesifisering og rammebetingelser

- Finne ut av hva som skal lages, økonomiske og praktiske rammer, målsettinger, krav til løsningen, når det skal ferdigstilles, funksjonalitet som skal være med, integrasjon til andre systemer o.l.

II Design og analyse.

- Analyser hva som trengs å lages, ev. i hvilken rekkefølge. Utarbeide ulike alternativer som er nyttige for å forstå det fremtidige systemet og dens brukere.
- Designe ulike deler eller systemet som ~~ett~~ helhet.

III Koding - sier seg selv ...

IV Integrasjon / implementering.

- Systemet rullet ut / og installert.

V system test

- Foretar testing av systemet i seg selv, og systemet i forhold til andre systemer, brukere o.l.

Denne kolonne er
forbeholdt sensor.Oppgave 3

B) Inkrementell : Minifossefallsmetoden ~~oppdel~~
som gjentas i inkrementer/
iterasjoner.

Man deler opp et større prosjekt i mindre
deler og utfører fossefallsmetoden i
~~hver~~ på hver del. Dermed oppnår man
tidligere ferdig kode ~~og~~ Hver iterasjon
skal resultere i ferdig testet funksjonalitet
før man starter på neste
iterasjon.

Prototypentvikling : Man lager en prototype av
hele systemet som deretter
testes ut overfor beslutnings-
taker og brukere.

Først når prototype er ferdig (f.eks. en
statisk HTML-side uten praktisk funk-
sjonalitet, eller diagrammer / bilder av
systemet) begynner man videre analyse
og coding!

Denne kolonne er
 forbeholdt sensor.

Oppgave 3

c) Prosjektets størrelse :

store prosjekter er ofte så kompliserte at det er enklere (og billigere) å dele dem opp i mindre sub-prosjekt som ~~leses~~ hver og en leses etter tur. Dermed blir det ~~større~~ også lettere å raskere lage ferdig ~~funksjonalitet~~ som kan presenteres for ~~oppkøper~~ den utålmodige opphavsgeberen.

Økonomi

Fossefall : Meget dyrt å endre dess lengre man kommer utover i prosjektets

Inkrem. : Mindre dyrt å endre ting underveis.

Prototype : I mange tilfeller minst dyrt å endre ting underveis.

Kompleksitet

Vil sammen med økonomi og prosjektets størrelse virke inn på hva ~~te~~ metode som brukes.

Brukere av systemet

Der man er veldig avhengig av ~~utgang~~ brukersespekt vil kanskje prototyping være ~~best~~ best.

%

Kurskode/Fag : IS-100
Kandidatnr. : 6151
Dato : 3/12-07
Ark nr. : 11 av 20

Denne kolonne er
forbeholdt sensor.

~~forts.~~ forts. 3 c)

Risiko for endrede betingelser underveis.
Der hver delt er stor spenning for at
rammebetingelser og kravspesifikasjon kan
endres seg underveis, er ofte inkrementell og
kanskje epis. prototyping & foretrekke.

Denne kolonne er
forbeholdt sensor.Oppgave 3

D) Systemutvikling fungerer ikke ~~er~~ uavhengig av organisasjonsutvikling. Et nytt system vil kunne påvirke sosiale strukturer innen en bedrift eller avdeling, vil kunne endre organisasjonsstruktur og ansvarsområder. Mennesker misliker i stor grad endringer. Har man først lært å vent med en måte å gjøre ting på innen en sosial- og organisatorisk struktur vil det alltid være en betydelig endringsmotstand. Endringsmotstanden (og effekten av den) vil variere med kultur og ~~er~~ tid siden siste endring, samt om endringen påtrykkes utenfra eller ~~er~~ er resultat av brukermedvirkning. Systemutvikling må derfor sees i kulturell kontekst. I skandinavia har ~~vi~~ ^{vi} f.eks. tradisjon for lik lønn, aktiv brukermedvirkning, ~~fl~~ flate organisasjoner. Tonen er ufermel, ideer blir vektlagt selv om den kommer fra nyansatte, unge personer. Dette gjør seg utslag både i selve systemutviklingen, og i måten man tenker å implementere systemet i en organisasjon.

Denne kolonne er
forbeholdt sensor.

Oppgave 3

e) ~~Argumenter mot:~~ Argumenter mot:

* Dyrt å endre underveis. Dess lengre man er ute i utviklingsprosessen, dess mer komplisert og dyrere er det å endre ting.

* Testing mot brukere kommer for sent i utviklingen. Ved prototyping vil brukere komme mye tidligere inn i prosessen, og det uten at man har kodet noe særlig. Det reduserer tidsforbruk ved kodig og også (ofte i alle fall) reduserer testing ~~og~~ og endringer eller implementering.

* Lang tid før ferdig produkt. Ved inkrementell eller prototyping tar det mindre tid å produsere ferdige kode, delvis funksjonalitet og/eller ferdig design. Det medfører at korrigeringer fra ledelse og brukere lettere kan gjøres underveis, og de som skal bruke og de som skal betale for systemet i større grad både kan følge med utviklingen og ha et eierforhold til den.

* Vanskeligere å få brukeraksept når systemet er ferdig.

* Vanskeligere å fange opp endrede kravspesifikasjoner.

Denne kolonne er forbeholdt sensor.

Oppgave 3 ~~4~~

F) Del av en bedrifts struktur eller system som er viktig for bedriftens virksomhet. f. eks. lønssystem, fakturamodul, ordresystem o.l.

Denne kolonne er
forbeholdt sensor.

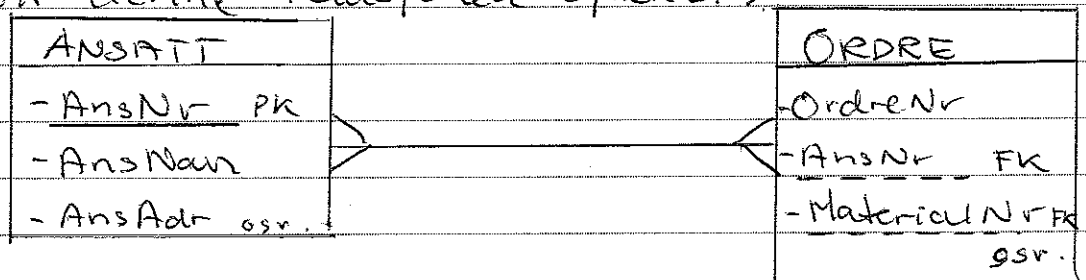
Oppgave 4

a) Relasjonsdatabaser er egentlig ikke annet enn en tabell med rader og kolonner. Radene og kolonnene er relatert til hverandre, en primærnøkkel i én rad relateres til verdier i raden, og ~~tabell~~ ~~kan~~ ~~rela~~ tabeller/tabellverdier kan relateres til andre tabeller/tabellverdier ved hjelp av tabellnavn, attributtnavn og primærnøkkel/jemmednøkkel.

Organisert i rader og kolonner.

b) Eks. En ansatt kan ha deltatt i mange ordre. En ordre kan ~~inneholde~~ inneholde bruk av mange ansatte (f.eks. én ordre som produseres ved hjelp av ansatte og materialer og hver man ønsker å se hvem ansatte som var med på å lage den spesifikke ordren.

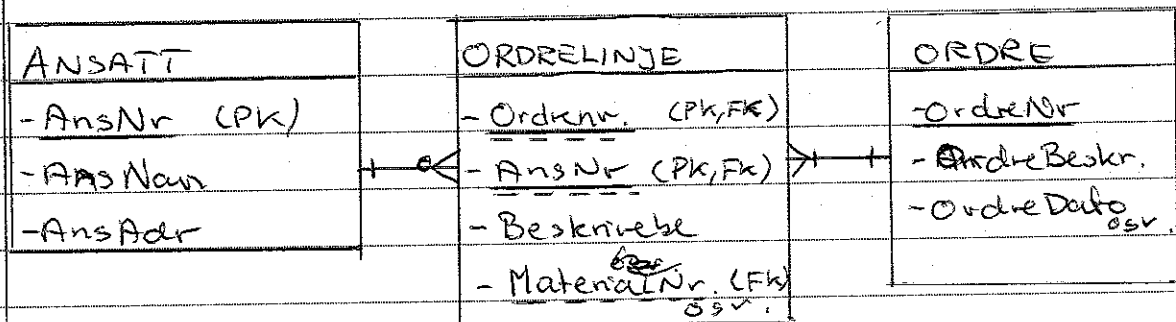
Giv denne relasjonen (f.eks.)



Denne kolonne er
forbeholdt sensor.

Forts. oppg. 4 B)

I en relasjonedatabase løser dette ved at man setter inn en ekstra tabell mellom disse slik at man ikke får mange-til-mange relasjon i hver ende, if. eks. slik.



Forbehold: Hver ordrelinje utføres av en ansatt.

c) ~~EH attributt~~ som entydig

En determinant (ett attributt eller flere attributter) som entydig identifiserer en rad
Eks. postnr → poststed.

D) EH attributt som ~~er~~ enten har verdien NULL, eller eksisterer som primærnøkkel i moder tabellen. (Er primærnøkkel i en annen tabell)

E) Det samme som ~~er~~ i oppgave D.

Den må enten ha verdien NULL eller ha verdien til primærnøkkel i moder tabellen eller per til.

Denne kolonne er
forbeholdt sensor.

Oppgave 5

a) At det er overensstemmelse mellom planlagte investeringer ^{-dntt} og bedriftens ~~over~~ overordnede IKT strategi ~~for~~ forretningsstrategi.

B) Data

Allt mulig er i prinsippet data. Men ikke all data er informasjon, strukturert data er informasjon.

~~Men selv strukturert data~~

Men for noen kan selv ustrukturert data gi informasjon og for andre ~~kan~~ ^{kunne} strukturert data gi informasjon bare mottakeren forstår den.

Metadata

Data om data. Informasjon om hvordan data er strukturert/organisert. Brukes f.eks. i databasesammenheng for å ~~kan~~ beskrive systemdata, f.eks. rettigheter, indeksering, constraints (forretningsregler) ~~og~~ m.m.

Informasjon

Data som er strukturert slik at det gir meningsfull informasjon (i alle fall for noen)



Denne kolonne er
forbeholdt sensor.

Oppgave 5

c) ERP: Enterprise Resource Planning
store systemer som inkluderer
kunde funksjoner, økonomifunksjoner og interne
bedrifts funksjoner / stabsfunksjoner (eks. personalavd.)
På denne måten prøver man å unngå
at data lagres mer enn et sted, ~~um~~ at
man har mange programmerandere å
forholde seg til, nyttiggjøre seg synergieffektene
(f.eks. lavere kostnader, raske system,
lettere å finne nødvendig informasjon).
Gir (ofte) mindre arbeid for ~~IT~~ IT-avd.
Bruke bedriftens ressurser på en mest
mulig optimal måte.

Denne kolonne er forbeholdt sensor.

Oppgave 6

- 1: Angir at det kommer PHP-kode
- 2: Oppretter kontakt med databasen ^{serveren} ~~med~~ ^{via} ~~via~~ parametrene som er oppgitt (forsøker å opprette kontakt)
- ~~3: Forsøker å koble seg til~~
Parametrene er: server, brukernavn, passord i nevnte rekkefølge.
- 3: Forsøker å koble til valgte database, her: 18100-olav.
- 4: Tom linje
- 5: Lager en query-string, ber databasen skrive ut alle rader fra tabellen "ansatt"
- 6: Lager en result-string som inneholder funksjonen "mysql_query" med query-stringen som parameter.
- 7: Avslutter databaseforbindelsen.
- 8: Tom linje
- 9: Går variabelen ~~num~~ "num" en int-verdi som er hvor mange rader det er i result-stringen.
- 10: Initiere en telle-(int) som settes til 0
11. Tom linje
- 12: starter en ~~for~~ WHILE-løkke. Så lenge telleren er mindre en num-verdien vil løkkeinnhold kjøres.
- 13: ~~løkkeinnhold starter~~ ~~for~~ For hver rad i ~~string~~ resultstringen gjøres denne: "fornavn" hentes fra stringen og tildeles string-variabelen "fornavn". %

Denne kolonne er
 forbeholdt sensor.

forts oppgave 6 a)

 14: Det samme, men her ~~titel~~ hentes / tildeles verdi-
 en av "etternavn"

15: samme: Her "e-post"

16: echo (skriver til skjerm):

- linjeskift

- Naem: "fornavn" "etternavn" Epost: "e-post"

17: cker teller med 1

18: Løkkeshutt

19. PHP-kode slutt (angir slutt på php-koden)

 B) For videre å lage, endre og distribuere
 design/stil på/mellom HTML-sider. Man
 trenger bare å ~~endre stil ett sted.~~
 endre / legge til / fjerne stilinformasjon ett
 sted.

 1: ~~Body tag~~ Act innhold i body-tag
 skal ha grå bakgrunnsfarge. (divs.
 hele nettsiden får grå bakgrunnsfarge)

2: Alle overskrifter med <H1>-tag får rød skrift.

 3: " " " " <H2> " " " " får grønn
 skrift, og er fet.

 4: Alle <p>-tagger med id = viktig får
 full vidde, fet skrift, rød font, gul
 bakgrunnsfarge

 5: Alle <p>-tagger med id = morsom får
 full vidde, normal fontvekt, rosa farge og
 font-størrelse satt til 25.