# Notater ingeniørfaglig innføringsemne eksamen

Eksamen og faget er delt opp i fire ulike deler:

1. Teamteori, prosjektteori, presentasjonsteknikk
2. IT-historie
3. IT og samfunn
4. Etikk

## Teamteori, prosjektteori, presentasjonsteknikk

**Belbin**

Belbin er et hjelpemiddel for å kartlegge hvilke kvaliteter ulike personer har i teamet. Ved å bruke belbin kan man enklere fordele inn roller siden man vet svakhetene og styrkene til teammedlemmene. Det finnes to måter å dele inn i roller:

1. **Effektivt arbeid**: Alle deles inn i rollene de er best egnet til. Dette vil sørge for at arbeidet blir mest mulig effektiv.
2. **Rollestyrking**: Man kan gi personer roller som de har scoret lavt på i belbin testen. Dette vil gjøre slik at personen kan styrke seg selv i den aktuelle rollen.

Personer med spesialkompetanse kan inneha en mentorrolle og bidra til å løfte hele resten av teamet innenfor det feltet der man selv er god. Man kan også få kartlagt eventuell kompetanse som mangler i teamet. Da må man vurdere om man skal lære noen opp til å håndtere denne rollen, eller om man skal hente inn ekstern kompetanse.

**Arbeidskontrakt**

En arbeidskontrakt skal gi et grunnlag til hvordan teamet planlegger å samarbeide. Arbeidskontrakten må skrives under av alle teammedlemmene og vil være juridisk bindende. Punkter som en arbeidskontrakt vil inneholde:

1. **Felles mål**: Effektiv samarbeid, få innsikt i hvordan det er å samarbeide i et team.
2. **Rollefordeling**: Teamleder, innleverings og kvalitetssikring, møteorganisering (innkalling, møteleder, referent)
3. **Prosedyrer**: møteinnkalling, krav til innsats og oppmøte, dokumenthåndtering, innlevering.
4. **Interaksjon**: sanksjoner over uregelmessigheter, utkasting av team-medlem, oppmøte og forberedelse, prosedyrer til uenighet innad i gruppa.

Konflikthåndtering er en viktig del av arbeidskontrakten:

Man må avklare hvilke prosesser som skal gjennomføres i en situasjon med konflikt. Man bør på forhånd finne ut hvilke konflikter som kan oppstå, hvem som har ansvar for å ordne opp, og fremgangsmåte for å håndtere situasjonen.

Hensikten med arbeidskontrakten er at hvis det oppstår f.eks. uenigheter så har man prosedyrer og måter man vil bruke for å håndtere situasjonen. Ved å gjøre dette kan det ikke oppstå uenigheter i fremtiden angående hvordan situasjonen ble håndtert, alle i teamet er enig i håndteringen på forhånd.

**FIRO (Fundemental Interpersonal Relations Orientation)**

En helhetlig teori om teamutvikling og modningsprosess.

1. Forstå menneskelig atferd og hva som skjer i møtet mellom mennesker
2. Inspirere til prosesser for utvikling i grupper
3. Utvikle mer effektive ledere
4. Bygge samarbeidene allianser og partnerskap

Modellen kan bidra til:

1. Skape trygghet og tilhørighet
2. Skape motivasjon
3. Håndtere ulikheter
4. Stimulere til bruk av effektiv feedback

Modellen deles inn i ulike faser:

1. **Tilhørighet fase:**

Modellen starter i denne fasen. Alle ønsker å bli en del av gruppen og bli akseptert. Alle er veldig høflige med hverandre. Gruppen tar sjelden opp alvorlige konflikter. Man forsøker å finne ut hvilken opptreden som er akseptabel.

1. Etablere struktur i gruppen
2. Avklare forventninger
3. Definere målsettinger
4. Leder må vise tydelig lederskap, gi godt definerte oppgaver og sette overnevnte temaer på agendaen.
5. **Mellomfase**

Denne fasen preges av overfladisk hygge og harmoni, konflikter skyves under teppet. Alt er bare velstand, ingen vil bruke energi på å sprekke boblen. Overgangen kommer KUN dersom noen punkterer denne boblen.

1. **Rollesøking / Kontroll**

I denne fasen avklares hele makt og ansvarsfordelingen. Preges av styring, kontroll, makt, lederskap, autoritet og ansvar. For mange en ubehagelig fase. Man tar gjerne opp spørsmål som:

1. Ansvarsfordeling (hvem gjør hva)
2. Maktfordeling (hvem bestemmer hva)
3. Konflikter og konflikthåndtering
4. Kompetanseavklaring (hvem kan hva og kan det best BELBIN?)
5. Fortrolighet og akseptering (hvem får vite hva)
6. Løsing av oppståtte problemer.
7. **Samhørighet fasen**

Grupper i denne fasen kjennetegnes av gjensidig respekt, aksept, og effektivt arbeid.

I denne fasen kan man også sette fokus på samhandling med andre team. Leder skal være ikke kontrollerende og kan integrere nye medarbeidere og oppgaver.

**Oppsummert**

Se for deg en bil. Alle stormer til bilen (tilhørighet), sloss om plassene (kontrollfasen), og kjører av gårde (samhørighetsfasen).

**Prosjekt V.S annet arbeid**

Definisjon: *Et prosjekt er en midlertidig oppgave igangsatt for å produsere et unikt produkt eller tjeneste.*

Et prosjekt vil ha en start og en slutt, altså det er en engangsutførelse, i motsetning til andre typer arbeid som ikke har en definert start og slutt. Et prosjekt vil også ha spesifikke mål med gitte rammer i form av tid, penger og andre ressurser.

Eksempler på andre typer arbeid: drift av systemer.

**Ulike mål i et prosjekt**

**Effektmål**: Skal beskrive hva oppdragsgiver vil oppnå med prosjektet.

* + Redusere portoutgiftene med 60&, gjennomføre og dokumentere et legoprosjekt.

**Resultatmål**: Skal beskrive det konkrete resultatet fra prosjektet.

* + Legorobot som spiller fire på rad mot en menneskelig motspiller.

**Prosessmål**: Skal beskrive resultatet av prosessen det er å jobbe med prosjektet.

* + Bestå i ingeniørfaglig innføringsemne,

## IT – historie

**Utvikling av programvare har en livssyklus på fem deler:**

1. Spesifisere systemet
2. Designe systemet
3. Programmere systemet
4. Teste systemet
5. Vedlikeholde systemet

Utvikling av disse på 70-tallet:

1. **Spesifisere systemet**
   * Formelle språk og modeller (tidlig 70-tall)
   * Allerede da innså man at endring av krav underveis høyst sannsynlig ville komme.
2. **Designe systemet**

* Strukturert programmering (Hoare og Dijkstra) til erstatning for «spaghettikode».
  + Delte store systemer i små på flere nivåer (FORTRAN 66, «subrutine»).
* Moduler (Parna), informasjonsskjuling
  + Moduler skal være mest mulig avhengige

1. **Programmere systemet**

* Programmeringsspråkene var allerede på plass
  + Men nye kom til: C (1973), Pascal (1971), Ada (1975), m.fl.
* Koden skal være billig å utvikle, effektiv å kjøre og lett å forstå.
* Hvordan utdanne og lede programmerere?

1. **Teste systemet** 
   * PACE (Produce Assurance and Confidence Evalator)
   * “Program testing can be used to show the presence of bugs, but never their absence.” (Dijkstra).
   * Allerede da innså man at testingen bør planlegges parallelt med designet av systemet.
2. **Vedlikeholde systemet** 
   * Vedlikeholdskostnadene overstiger ofte utviklingskostnadene (Barry, 1979)

**Internett**

Da internettet fikk sitt gjennombrudd fikk programvare utviklerne flere utfordringer:

1. **Nye produkter**: distribuerte systemer, nettverksprogramvare, søkemotorer, maskinoversettelse av naturlige språk, web-applikasjoner med alle typer data.
2. **Flere utfordringer**: kommunikasjon mellom ulike typer maskiner, protokoller, epost osv.

Web 2.0

Web 2.0 er ikke en teknologi, men måten internett etter hvert kunne brukes på ved at brukerne ble viktige leverandører av innhold. Sosiale medier er et godt eksempel på denne utviklingen man har sett som førte til web 2.0.

Internett

Da russland sendet opp satelitten «Sputnik» ble USA bekymret for landets sikkerhet og lanserte DARPA ( Defence Advanced Research Projects Agency) i slutten av 50-tallet. DARPA skulle utvikle ny og moderne teknologi for militæret. Den første tanken bak internettet var at militæret skulle kunne kommunisere etter et atomangrep med et desentralisert nettverk/kommunikasjonssystem. Dette ble først kalt ARPANET, og ble først konstruert i 1968. Etter hvert som flere maskiner ble koblet på nettet utviklet man protokoller, den første var NCP (Network Control Protocol).

I 1972 ble applikasjonen email lagd. Ulempen med NCP var at kun noen maskiner kunne kommunisere, derfor lagde man en ny protokoll TCP/IP i 1973 (brukt fra 1974). ARPANET vokste etter hvert til noe større enn et militærprosjekt, SATNET brukte satellitt for å koble USA til Norge, også deretter til Europa. På 80-tallet ble CSNET (Vitenskapelig nettverk) lagd, og koblet sammen med ARPANET etter hvert. I tillegg ble DNS (Domain Name System) lagd.

**Tidslinje**

1. **Slutten av 50-tallet**: DARPA ble lansert.
2. **1968**: ARPANET ble lagd.
3. **1972**: Email ble lagd.
4. **1973**: TCP/IP ble innført som nye protokollen.
5. **80-tallet**: CSNET ble lagd og koblet sammen med ARPANET etter hvert.
   * DNS ble innført.
   * Hastigheten økte, 25 ganger raskere enn før (fra 25kb/s til 1,5mb/s).
   * Nettverket eksploderte, 1985 (1000 hosts) til 1989 (200.000 hosts).
6. **1985**: Over 1000 (pc-er/hoster) på nettverket. ARPANET ble skilt fra MILNET (militære bruk).
7. **1989**: WWW ble lagd av Sir Tim Berners-Lee, Sir Sam Walker og Robert Caillau.
8. **1991**: WWW ble introdusert til offentligheten, starten på økningen av interenettet.
9. **90-tallet**: Internettet fortsatte å vokse, slutten av 90-tallet (flere hundre millioner hosts).
10. **1993**: Mosaic (verdens første nettleser) ble lansert.
11. **1995**: Yahoo ble introdusert (første søkemotor) av Jerry Yang og en annen student.

Viktige applikasjoner på internettet: Email, forum, online dating, sosiale nettverk/medier, blogger.

Internettet har også medført en enorm økonomisk vekst, trolig billioner av dollar har blitt skapt. Google har vokset til å bli en av de største og mest lønnsomme selskapene i verden.

**Kommunikasjonsprotokoll**

En kommunikasjonsprotokoll er et sett med regler for hvordan en datastrøm (strøm av bytes) skal sendes fra avsender og tolkes hos mottaker. Altså enhetene må snakke samme «språk». Eks. Tallet 39 kan være data, men 39 grader celcius er informasjon. Eksempler på slike protokoller er IP og http.

Problemet med NCP som ble brukt tidlig i utviklingen av internettet var at denne protokollen var proprietær. Tenk hvis f.eks. kun du kunne kommunisere med maskiner som ble lagd av sony eller microsoft, og ingen andre. Løsningen på dette ble TCP/IP som erstattet NCP.

## 3. IT og Samfunn

**Kilder til energiforbruk i systemutviklingsprosessen:**

Programvareutvikling foregår som oftest i et selskap, derfor kan kilde til energiforbruk være kontorlokaler, møter, firmareiser, kommunisering over telenett, sosiale aktiviteter.

* *Måten man jobber på i utviklingsfasen*: mange personer, distribuerte team, tid
* *Automatisering av ledd i prosessen*: f.eks. automatisert testing
* *Måten software jobber på og hardware energiforbruk* – dvs. energiforbruk i drift
* *Generelt om utnyttelse av hardware*: sleep-mode, dynamisk lastdeling, virtualisering.
* *Livsløpet:* produksjon, distribusjon, bruk, deaktivering og «disposal».

Disse kan deles inn i tre hoveddeler: utvikling, bruk og «end of life».

**Hvordan teknologi påvirker oss som brukere og som samfunn i både hverdagsliv og arbeid:**

Endring i teknologi kan endre samfunnet og vår adferd. Endringer i adferd kan også føre til ny teknologi (nye markeder). Dette har både positive og negative effekter:

* Informasjonstilgang
* Behagelig liv, enklere arbeid
* Teknologi former handlingsrom (f.eks. bruer, funksjonalitet)
* Bedre kvalitet
* Færre fysiske møter, mer sosiale medier
* Utfordringer – stress (alltid på, alltid synlig), tidsbruk på nett
* Større rekkevidde, mer effektive
* Potensiell arbeidsløshet, automatisering og effektivisering.

**Hvordan man designe og utvikle teknologi uten at det får uheldige effekter for brukerne:**

Hvis man setter brukerne i fokus ved design og utvikling av en teknologi så finnes det flere hensyn man kan ta. Blant annet så man først undersøke hva brukerne ønsker og eventuelle konsekvenser en teknologi kan ha for brukeren. Etter dette må man ta hensyn til disse undersøkelse når man designer og utvikler en ny teknologi. Det kan tenkes at spesifikasjonene til hva som brukerne ønsker eller konsekvenser som bør unngås endres underveis i prosessen av utviklingen. Det kan f.eks. forekomme endringer etter testing av teknologien til utvalgte brukere.

**Rebound effect**

Når man øker effektiviteten i en teknologi vil man se en utvikling hvor det brukes mer energi og ressurser enn tidligere.

1. Biler med mer effektivt drivstofforbruk
   * Biler blir større og raskere, flere biler, vi kjører mer
2. Printere vil føre til «papirløse kontorer»
   * Billigere, mer effektivt å printe enn før.

Ulemper: E-avfall, energiforbruk, ressursfordeling

**Noen begreper**

*Grønn IKT*: Mer energieffektiv produksjon og bruk gjennom hele livsløpet.

1. Grønne datasentre og skytjenester, optimalisering:
   * Samlokalisere servere
   * Synenergi strømforsyning, brebåndstilgang, kjøling, utnyttelse av overskuddsvarme
   * Reduserer strømforbruket 15-80%.
   * Virtualisering

*Avmaterialisering*: Digitale alterativer til fysiske produkter, og transport av varer og personer.

*Smart IKT*: IKT som verktøy for et bedre miljø, f.eks. minske klimagassutslippene.

**Fafo-rapporten «Det norske IKT-samfunnet» Tunge trender**

1. Digitalisering
   * Omforme noe analogt til noe binært
2. Integrering
   * Alle datamaskiner kommuniserer med hverandre via protokoller
3. Miniatyrisering
   * Moores lov, «mer kan gjøres raskere med mindre plass»
4. Globalisering
   * Takker være internett
5. Multitasking og multiservicing
   * Direkte distribusjon fra opphav til forbruker
6. Individualisering
   * Den enkelte bestemmer selv hva den vil se av filmer eller når nyhetene skal spilles av.
7. Avprivatisering
   * Nettaktivitet legger igjen spor
8. Interaktivitet
   * Alle kan kommunisere med alle, direkte kommunikasjon
9. Demokratisering
   * I prinsippet ingen begrensninger, alle kan få hva som helst ved noen klikk.
   * Digitale skiller: geografiske, sosiale, aldersmessige, økonomiske.
   * Økende overvåkning av nettaktiviteter.
10. Sosialisering
    * Normer for oppførsel på nettet.