Løsningsforslag/sensorveiledning ST V23

NB! Svarene er kun veiledende, og kan ikke holde feil!

1 Part 1 - Multiple response

Hvilke to av de følgende er IKKE en review-type, ifølge ISTQB-ordlisten?

Velg ett eller flere alternativer

- [ ] Inspeksjon
- [ ] Inntrengingsdeteksjon ✔
- [ ] Teknisk review
- [ ] Gjennomgang
- [ ] Eksaminasjon ✔

Maks poeng: 2

2 Part 1 - Multiple choice

Hvilket av følgende utsagn beskriver best White-box testingsteknikker?

Velg ett alternativ:

- En prosedyre for å utlede og / eller velge test caser basert på testernes erfaring, kunnskap og intuisjon.
- En prosedyre for å utlede og / eller velge test caser basert på en analyse av spesifikasjonene, enten funksjonelle eller ikke-funksjonelle, til en komponent eller et system uten referanse til dens eller dets interne struktur.
- En prosedyre for å utlede og / eller velge test caser basert på en analyse av den interne str[illegible]uren til en komponent eller et system. ✔
- En rekke test caser i eksekveringsrekkefølge, og eventuelle tilknyttede handlinger som kan være nødvendige for å sette opp de første forutsetningene og eventuelle avsluttende aktiviteter etter kjøringer.

Maks poeng: 1

3 Part 1 - Multiple choice

Begrepet "Impact Analysis" (konsekvensanalyse) er i ISTQB's pensum knyttet til spesiell type testing. Hvilken?

Velg ett alternativ

- Strukturert testing
- Funksjonell testing
- Regresjonstesting ✔
- Ikke-funksjonell testing

Maks poeng: 1

4 Part 1 - Multiple choice

Hva er hensikten med bekreftelsestesting?

Velg ett alternativ:

- For å bekrefte at ingen utilsiktede feil har blitt introdusert som følge av endringer som er gjort.
- For å bekrefte at en feil er rettet på riktig måte. ✔
- For å bekrefte brukernes tillit til at systemet vil møte deres forventninger og behov.
- For å bekrefte at den detaljerte logikken til en komponent samsvarer med spesifikasjonen.

Maks poeng: 1

5 Part 1 - Multiple choice

Fullfør setningen slik at den gir den mest presise definisjonen av alfatesting.

Alfatesting ...

Velg ett alternativ:

- ... utføres av et uavhengig testteam.
- ... utføres av sluttbrukere hos utviklerne. ✔
- ... utføres av sluttbrukere på deres eget sted.
- ... er nyttig for å teste programvare utviklet for en spesifikk kunde eller bruker.

Maks poeng: 1

6 Part 1- Matching / pairing

Avgjør hvilken type teststrategi som er brukt i hvert av tilfellene.

Finn de som passer sammen:

	Metodisk	Konsultativ	Analytisk	Modellbasert
Testene er basert på synspunktene til forretningseksperter.		✔		
Testene er basert på systematisk bruk av et forhåndsdefinert sett med testbetingelser.	✔			
Testene er basert på et tilstandsdiagram av produktet.				✔
Testene er konstruert og prioritert ut fra risikonivå.			✔	

Maks poeng: 3

7 Part 1 - Multiple choice

Hvordan skiller *verifisering* seg fra *validering* i forbindelse med testing?

Velg ett alternativ:

- De skiller seg ikke fra hverandre. Verifisering og validering betyr det samme i testingskontekst.
- Verifisering er relatert til hvorvidt programmet tilfredsstiller brukernes behov, mens validering er relatert til om programmet fungerer i henhold til spesifikasjonene og kravene.
- Verifisering er relatert til om programmet fungerer i henhold til spesifikasjonene og kravene ✔ens validering er relatert til om programmet tilfredsstiller brukernes behov.
- Verifisering er relatert til 100% kodedekning, mens validering er relatert til 100% use case-dekning.

Maks poeng: 1

8 Part 1- Matching / pairing

Hvilke av følgende er en fordel og hvilke utgjør en risiko ved bruk av verktøystøtte i testing?

	Risiko	Fordel
Undervurdering av innsatsen som kreves for å vedlikeholde testressursene som genereres av verktøyet	✔	
Større konsistens og repeterbarhet		✔
Redusering av repeterende arbeid		✔
Enklere tilgang til informasjon om testaktiviteter og testresultater		✔
Urealistiske forventninger	✔	
Objektiv vurdering		✔
Overdreven tillitt til verktøyet	✔	

Maks poeng: 3

9 Part 1 - Multiple choice

Hvilket av følgende alternativer beskriver best testdrevet utvikling (TDD - Test driven developement)?

Velg ett alternativ:

- TDD er en utviklingsmetodikk der enhetstestene skrives før koden for å bestå dem implem ✔ eres.
- TDD er en utviklingsmetodikk der uavhengige testere og utviklere jobber sammen i team.
- TDD er en smidig utviklingsmetodikk der sluttbrukeren tester programmet for hver iterasjon.
- TDD er en annet betegnelse for brukersentrert utvikling.

Maks poeng: 1

10 Part 1- Matching / pairing

Match kolonneoverskriftene med riktig testnivå:

	Brukerhistorier	Funksjonelle og ikke-funksjonelle krav	Kode	Global design
Akseptansetesting	✔			
Enhetstesting			✔	
Systemtesting		✔		
Integrasjonstesting				✔

Maks poeng: 3

11 Part 1 - Multiple response

Statisk kodeanalyse identifiserer alle unntatt to av følgende problemer. Hvilke?

Velg ett eller flere alternativer

- [ ] Brudd på programmeringsstandarder
- [ ] Minnelekkasjer ✔
- [ ] Deklarerte variabler som aldri blir brukt
- [ ] Avvik mellom aktuell og forventet output ✔
- [ ] Kode som ikke kan nås

Maks poeng: 2

12 Part 1 - Multiple choice

Hvilket av følgende utsagn beskriver best Black-box designteknikker?

Velg ett alternativ:

- ○ En rekke test caser i eksekveringsrekkefølge, og eventuelle tilknyttede handlinger som kan være nødvendige for å sette opp de første forutsetningene og eventuelle avsluttende aktiviteter etter kjøringer.
- ○ En prosedyre for å utlede og / eller velge test caser basert på en analyse av den interne strukturen til en komponent eller et system.
- ○ En prosedyre for å utlede og / eller velge test caser basert på en analyse av spesifikasjonene, enten funksjonelle eller ikke-funksjonelle, til en komponent eller et system uten referanse ✔ ens eller dets interne struktur.
- ○ En prosedyre for å utlede og / eller velge test caser basert på testernes erfaring, kunnskap og intuisjon.

Maks poeng: 1

13 Part 1 - Multiple response

Hvilke av følgende kan være et mål på dekningsgrad for tilstandsbasert testing? Velg ett eller flere alternativer. (NB! -0.5 for feil svar)

- [ ] Responstiden for hver overgang.
- [ ] Antall overganger som er blitt testet. ✔
- [ ] Antall tilstander som er nådd. ✔
- [ ] Antall grenseverdier som er blitt testet.
- [ ] Spesifikke sekvenser av overganger som er blitt testet.

Maks poeng: 2

14 Part 1- Matching / pairing

Match følgende prosesser med tilhørende aktiviteter:

	Fastsette omfanget og målene for testingen	Gruppere tester i testskript	Skrive en test-oppsummerings-rapport for interessentene	Omgjøre testmålene til testbetingelser
Testanalyse	○	○	○	○ ✔
Testplanlegging	○ ✔	○	○	○
Testrapportering	○	○	○ ✔	○
Testimplementasjon	○	○ ✔	○	○

Maks poeng: 3

Part 2 – EP / BVA

a) I denne oppgaven kan du forta ekvivalensklasseinndelingen på flere måter, avhengig av hva du fokuserer på og hvordan du velger å foreta oppdelingen. Beskriv minst to ulike måter input-verdiene kan deles inn i ekvivalensklasser.

Svar – 2 poeng

Alternativ 1: Dele inn året etter stjernetegnene.

Alternativ 1: Dele inn året etter årets måneder.

Alternativ 3: Dele inputverdiene i to partisjoner: gyldige/ugyldige Datoer.

Alternativ 4: Dele inputverdiene i to partisjoner: Numeriske/ikke-numeriske verdier.

b) Hvilken betingelse må være oppfylt for at ekvivalensklassene skal inneholde grenseverdier?

Svar: - 1 poeng

Elementene må være sortérbare slik at man kan avgjøre hvilke elementer som ligger nærmest grensene mellom ekvivalensklassene

c) For hver av de to ekvivalensklasseinndelingene du foretok i oppgave a) skal du angi eventuelle grenseverdier, både gyldige og ugyldige. Da antall grenseverdier kan bli veldig høyt trenger du ikke å ramse opp alle. Det viktigste er at du beskriver dem slik at det framgår hvilke de er.

Svar: - 2 poeng

Alternativ 1: eks. maks-verdi: 19. april, min-verdi: 20. april

Alternativ 2: eks. maks-verdi: 31. januar, min-verdi: 1. februar

Alternativ 3:

a) Måneder, eksempel:
   Gyldig min-verdi: 1, gyldig maks-verdi: 12
   Ugyldige maks-verdi: 0, ugyldig min-verdi-verdi: 13
b) Dager (avhenger av hvilken måned vi er i)
   Eksempel mai:
   Gyldig min-verdi: 1, gyldig maks-verdi: 31
   Ugyldige maks-verdi: 0, ugyldig min-verdi-verdi: 32

Part 2 – Decision table

Løsningsforslag:

BI: Bachelorgrad i informatikk
ST: Kurs i Software testing
C: Gjennomsnittskarakter over C.

A: Avslag uten begrunnelse
O: Opptak
BO: Betinget opptak
FK: Oppfordring om å forbedrekarakterene

A*: Avslag pga. manglende kurs i testing og for gjennomsnittskarakter under C.

NB! Det er ikke nødvendig å skille mellom A og A*

a) 3 poeng

	Regel 1	Regel 2	Regel 3	Regel 4	Regel 5	Regel 6	Regel 6	Regel 8
BI	true	true	true	true	false	false	false	false
ST	true	true	false	false	true	true	false	false
C	true	false	true	false	true	false	true	false
Aksjon	O	FK	BO	A*	A	A	A	A

NB! Det er ikke nødvendig å skille mellom A og A*, men det gis ikke trekk hvis man gjør det.

b) 2 poeng

R5 til R8 kan slås sammen, da søkeren vil få avslag uansett om vedkommende har høy gjennomsnittskarakter og/eller kurs i testing:

	Regel 1	Regel 2	Regel 3	Regel 4	Regel 5-8
BI	true	true	true	true	false
ST	true	true	false	false	---
C	true	false	true	false	---
Aksjon	O	FK	BO	A*	A

NB! Det er ikke nødvendig å skille mellom A og A*, men det gis ikke trekk hvis man gjør det.

c) 2 poeng

Testtilfelle 1 dekker Regel 1.
Testtilfelle 2 dekker Regel 2.
Testtilfelle 3 dekker Regel 3.

Vi mangler to tester for avslag:

Testtilfelle 4 som dekker Regel 4:
Søkeren har bachelorgrad i informatikk, men ikke det nødvendige kurset i Software testing og heller ikke gjennomsnittskarakter på C eller bedre.
Testtilfelle 5 som dekker Regel 5-8:
Søker uten bachelorgrad i informatikk.

d) 1 poeng

Antall kombinasjoner blir $2^4 = 16$.

e) 2 poeng

Beslutningstabeller er nyttige å bruke når man skal **granske (review) programmets spesifikasjoner** under den **statiske testingen** i **planleggingsfasen** av programmet, dvs. før implementasjonen har funnet sted. (På venstre side av V-en i V-modellen).

Part 2 – State transition

Antall tilstander: 7,
Antall overganger: 8.

a) 1,5 poeng
- Testtilfelle 1: 1, 2, 3, 6
- Testtilfelle 2: 1, 2, 3, 4, 6
- Testtilfelle 3: 1, 2, 3, 4, 5, 7

b) 1,5 poeng
- Testtilfelle 3 gir høyest tilstandsdekning.
- Dekningsgraden er 6/7 (0,86 = 86%)

c) 1 poeng
- Testtilfelle 3 gir høyest overgangsdekning.
- Dekningsgraden er 5/8 (0,625 = 62,5%)

d) 2 poeng
- Tilstandsdekning: Nei. Når man kommer til tilstand 5 må man velge å enten gå til tilstand 6 eller gå til tilstand 7. Man kan ikke gå til begge tilstandene i et enkelt testtilfelle.
- Overgangsdekning: Nei. F.eks. er det tre overganger som leder til tilstand 6, men ingen overgang tilbake. Følgelig får man bare testet en av dem i et enkelt testtilfelle.

e) 2 poeng
For de tre testtilfellene til sammen blir dekningsgraden
- Tilstandsdekning: 100%
- Overgangsdekning: 7/8 (0,875 = 87,5%)

f) 1 poeng
Vi må legge til testtilfellet der kunden skriver feil passord to ganger, men riktig den tredje gangen: 1, 2, 3, 4, 5, 6.

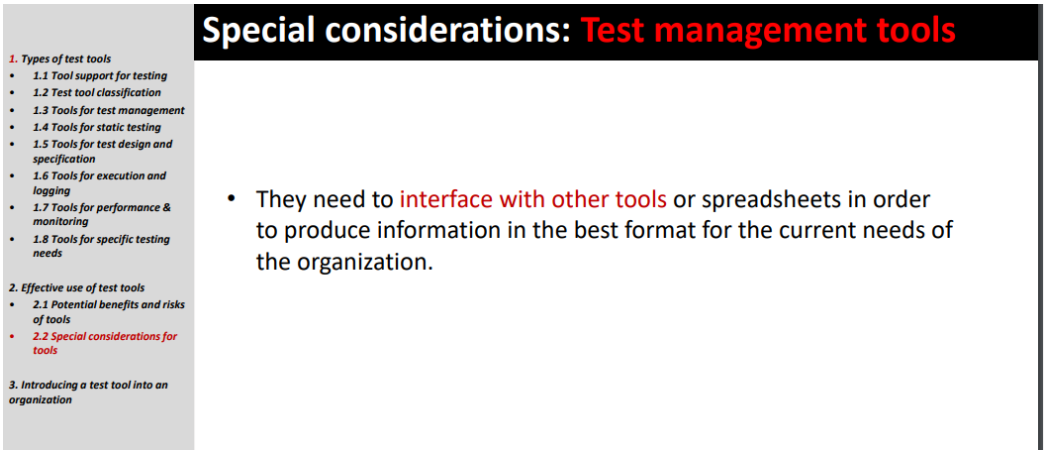
g) 1 poeng
Overgangsdekning er sterkere enn tilstandsdekning. 100% overgangsdekning garanterer 100% tilstandsdekning, men ikke omvendt.

Part 3 Short Questions

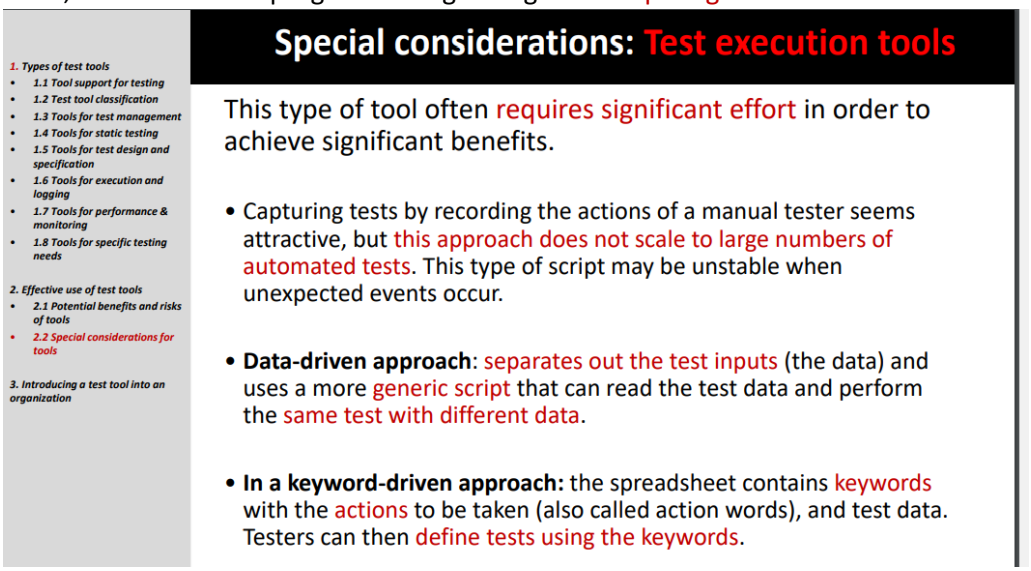
I læreboka, kapittel 6 er et av delkapitlene viet spesielle hensyn og utfordringer knyttet til bruk av testledelsesverktøy og testkjøringsverktøy.
Hva bør vi være spesielt oppmerksomme på før vi tar disse verktøytypene i bruk?

Svar:

a) Testledelsesverktøy brukes sjeldent alene, men sammen med annen type verktøystøtte. Man bør være forberedt på at det kan oppstå integrasjonsutfordringer når ulike systemer skal kommunisere og fungere sammen. 2 poeng



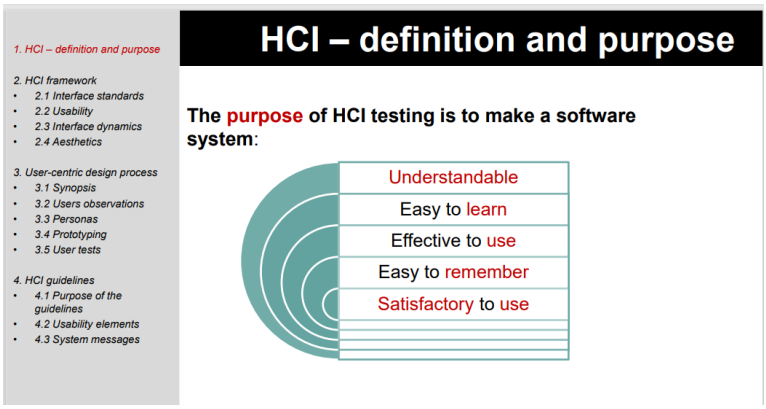
b) Man bør være oppmerksom på at testkjøringsverktøy krever programmeringsferdigheter, da man må skrive skript som beskriver de ulike stegene i testene. Skal dette gjøres kostnadseffektivt, krever det at man skrive datadrevne og/eller nøkkeldrevne (key-driven) skrift, noe som krever programmeringsferdigheter. 3 poeng



Part 3- Short question

I den andre obligatoriske oppgaven testet du en billettautomat. Denne typen automat skal kunne brukes av alle typer mennesker i alle aldre. Fagfeltet Human Computer Interaction, HCI, (menneske-maskin-interaksjon) har utviklet et rammeverk som vi kan støtte oss til for å oppnå dette. Hvilke fem punkter bør du være bevisst på i utformingen av brukergrensesnittet til billettautomaten? (Vi er her ikke interessert i nivåene i rammeverket, men kun i punktene som beskriver hensikten med HCI-testing.)

Svar: 1 poeng for hvert punkt

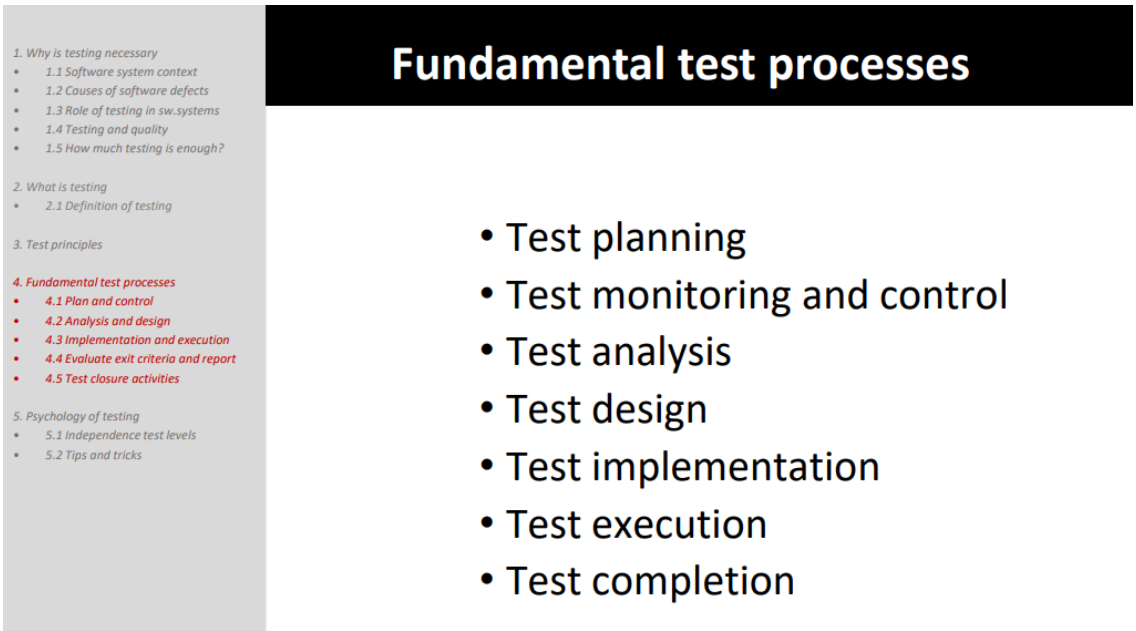


Part 3 - Short question

Navngi og beskriv kort de ulike fasene av den fundamentale testprosessen. Plasser fasene i sekvensiell rekkefølge.

Svar: 1 poeng for hver av punktene på venstre side med rødt.

I pensum til ISTQB er noen av trinnene slått sammen, slik det står på med rødt på venstre side, mens i læreboka er trinnene delt opp slik som på sliden. Hvordan de ulike trinnene er inndelt ikke viktig så lenge alle er med og de står i riktig rekkefølge.



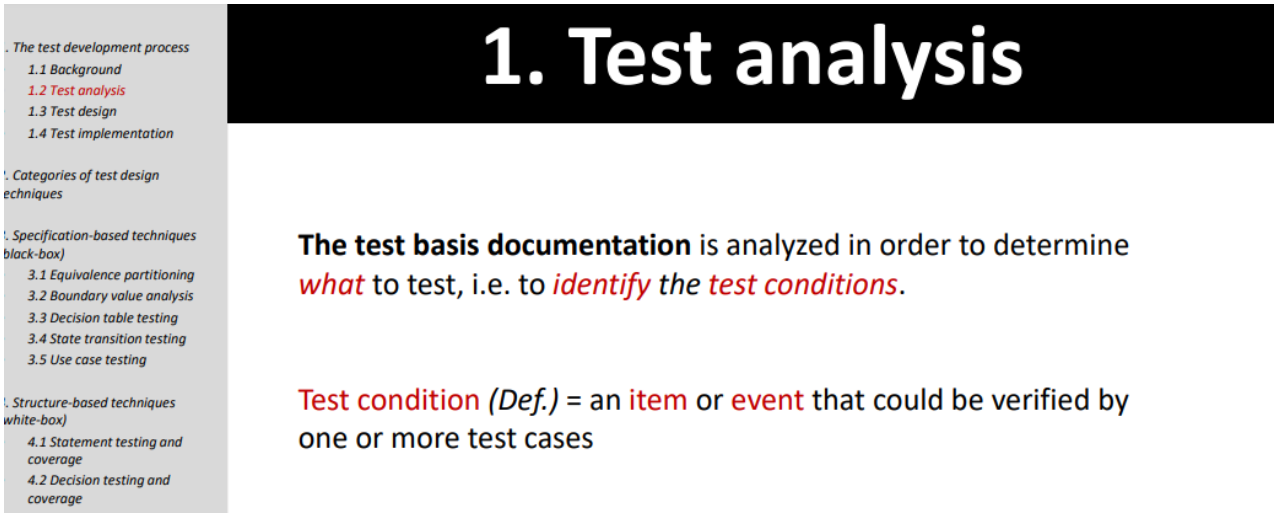
Part 3 - Short questions

1. Forklar kort forskjellen mellom et testforhold (test condition) og et testtilfelle (test case). Illustrer forskjellen med et konkret eksempel.

2. Beskriv hva som inngår i et testtilfelle.
Illustrer med et konkret eksempel.

Svar:

1. 3 poeng



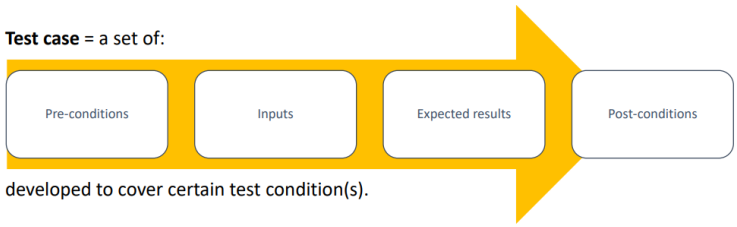
Et testforhold (test condition) er en mer overordnet beskrivelse av noe som skal testes, mens et testtilfelle er en konkret beskrivelse av en test. For å teste et testforhold, trenger vi gjerne mange konkrete testtilfeller. (Overgangen mellom testforhold og testtilfeller er imidlertid flytende.)

Eks: Et enkelt testforhold kan være testing av login-funksjonen.

Å teste login-funksjonen vil kreve mange testtilfeller:

- Gyldig/ugyldig brukernavn
- Gyldig/ugyldig passord
- Kontroll av feilmeldinger til brukeres
- Glemt passord? Tilbud om registrering av nytt
- Osv.

2. 2 poeng (4 x 0,5)



Et testtilfelle må beskrive hvilke **betingelser som gjelder før testen**, hvilke **inputverdier** testen skal ha, **hvilke betingelser som skal gjelde etter testen**, samt **forventet resultat** (kalt et orakel).

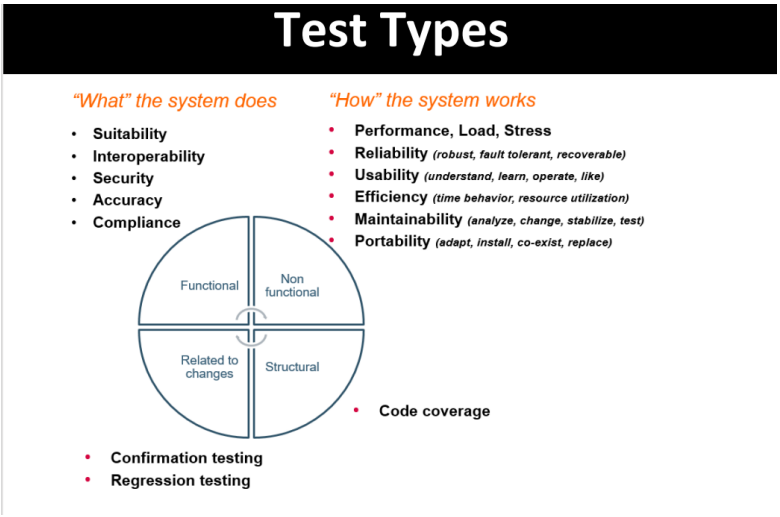
Eksempel: Innsetting av et element i en sortert datastruktur.

Part 3 - Short question

ISTQB skiller mellom ulike typer testing ut fra hva som er hensikten med testingen.

Hvilke fire testtyper har vi i henhold til ISTQB'definisjoner, og hva er hensikten med hver enkelt testtype?

Svar: 1 poeng for forklaring av type, inkludert hensikt.



Funksjonell testing tester funksjonaliteten, dvs. tester **hva** programmet gjør i forhold til dets spesifikasjoner. 1 poeng

Ikke-funksjonell testing tester **hvordan** programmet utfører oppgavene, dvs. de mer kvalitetsmessige aspektene ved utførelsen, som brukervennlighet, robusthet, responstid osv. 1 poeng

Strukturell testing vurderer grundigheten av testingen, målt i ulike typer testdekning (coverage). Krever innsyn i koden. 1 poeng

Testing relatert til forandring har til hensikt å sjekke at programmet fremdeles fungerer som det skal **etter en endring har funnet sted** (utvidelse eller feilretting).

- **Bekreftelsestesting**: Sjekker at feilen virkelig er rettet opp. 1 poeng
- **Regresjonstesting:** Sjekker at ingen nye feil er introdusert i koden som følge av en endring. 1 poeng

Part 4 - Essay

I denne oppgaven skal du skrive en tekst om de psykologiske aspektene knyttet til testing og utviklingsarbeid.

I teksten skal du besvare spørsmålene nedenfor. Du står imidlertid fritt til å ta med andre momenter som du synes er relevante for problemstillingen.

20 poeng fordeles som beskrevet under. Det fem siste poengene brukes til å evaluere essayet som helhet (hvor bra det er skrevet) på en skala fra 0 til 5.

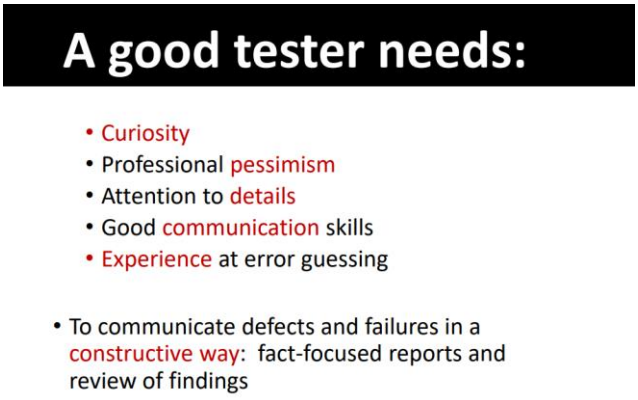
- Hva er den viktigste forskjellen mellom tankesettet til en utvikler og en tester, og hva er årsaken til denne forskjellen?

Testeren må være skeptisk til resultatet, og regne med i finne feil. Det krever et **pessimistisk** tankesett. Utvikleren er imidlertid avhengig av å tro på at hen skal mestre utfordringene for å lykkes, noe som krever et **optimistisk** tankesett. 1 poeng

- Hvorfor er det viktig for en tester å ha gode sosiale egenskaper og kommunikasjonsferdigheter?

Feilen og mangler som avdekkes må **kommuniseres til utvikleren**. Hvis ikke dette gjøres på en konstruktiv måte, risikerer man å gjøre vondt verre i forhold til målet om et sluttprodukt med best mulig kvalitet. 1 poeng

- Hvilke andre personlige egenskaper bør en tester ha? Beskriv minst tre andre egenskaper.



3 poeng

- Det hender det oppstår utfordringer i forholdet mellom utvikler og tester. Hva er årsaken?

Avdekking av feil og mangler kan oppleves som kritikk og som «kjepper i hjulene». Hvis testeren oppfører seg som en dommer i stedet for en medhjelper, og feil og mangler kommuniseres som kritikk av utviklerne, kan det skape spenninger mellom partene. Utviklere kan oppleve det som svært frustrerende hvis feil og mangler påpekes rett før ferdigstilling/levering, ikke minst med de praktiske og økonomiske konsekvensene det medfører. (Her kan det skrives mye mer …) 1 poeng

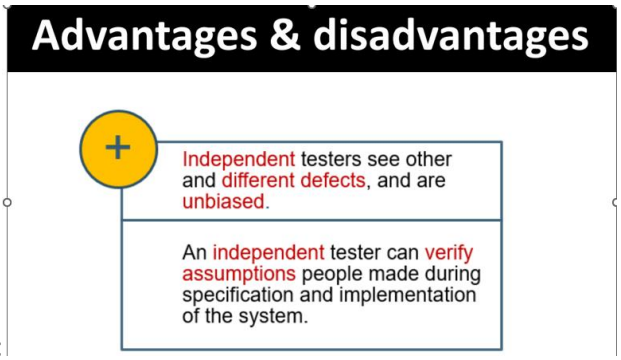
- Hvordan kan testeren kommunisere feil, mangler og annen negativ informasjon til utvikleren på best mulig måte?

For å nå et felles mål om best mulig kvalitet på sluttproduktet, er det avgjørende at relasjonen mellom utviklere og testere er best mulig. Feil og mangler må kommuniseres på en fakta-fokusert måte, og ikke som kritikk av utvikleren. Testeren må gi klart uttrykk for at målet er å bidra til et bedre produkt, og ikke en evaluering av utviklerne. Som tester kan det være lurt å ikke bare påpeke feil og mangler, men også komme med positiv kritikk av det som er bra (til tross for at det første er jobben).
2 poeng

- To communicate defects and failures in a constructive way: fact-focused reports and review of findings

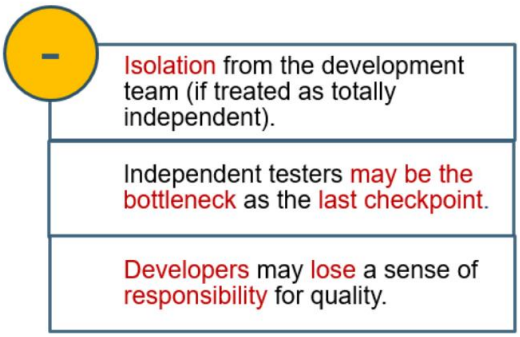
- Hva er fordelene og ulempene med uavhengig testing?

Positivt: 3 poeng



:

Negativt: 3 poeng



- Beskriv minst fire ulike nivåer av testuavhengighet.

4 poeng

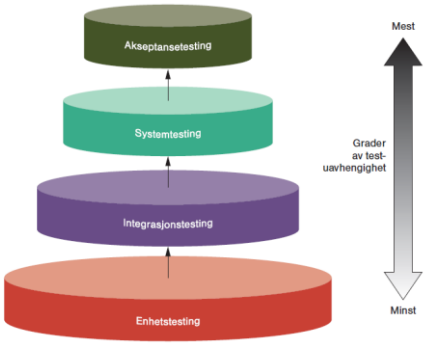
Independence test levels

- Independence levels:

- Tests designed by the same person who wrote the code
- Tests designed by another person from the same team, but same organization
- Tests designed by a person from a separate testing team , but in the same organization
- Tests designed by a person from an outside organization / company (outsourcing the testing)

- Uavhengig testing brukes i ulik grad på de fire testnivåene. Hvilke grader av testuavhengighet er best egnet på de ulike testnivåene? Ranger de fire testnivåene fra best egnet til minst egnet i forhold testing utført av et uavhengig testteam.

2 poeng



Figur 15.1: Grader av uavhengighet.