

Att skriva rapporter

Det förekommer nästan alltid delade meningar om vad en rapport bör/ska innehålla för att den ska vara bra. Vissa tycker att studenter får lära sig fel saker i början och sedan, tyvärr, håller fast vid detta.

Det problem som har funnits är att man under sin första kurs fått krav om att rapporten ska se ut på ett speciellt sätt. Detta är det egentligen svårt att ha något emot, men däremot vad man kräver kan diskuteras. Det som ställer till mest frustration bland studenter, och även handledare för den delen, är det avsnitt som kallas Systembeskrivning. Studenterna får lära sig att en systembeskrivning alltid har samma upplägg. Detta är tyvärr ett allvarligt misstag!

En rapport måste tillåtas vara lite flexibel. Att en kurs som sysslar med objekt-orienterade tekniker använder ett system för labbrapporter som speglar detta.

Nedan följer en rad guidelines som jag, som handledare och student, anser vara grunden för en bra rapport. Man måste dock tänka på att uppgiftens utformning styr rapportens exakta utformning.

En rapports syfte

- Träna skriftlig framställan
- Dokumentera sitt arbete på ett professionellt och formellt sätt
- Göra det möjligt för andra att förstå och ta del av arbetet
- Förbereda sig själv för det som väntar i arbetslivet, där dokumentation är en central del av arbetet

En rapports innehåll

- Försättsblad
- Eventuell sammanfattning (abstract)
- Innehållsförteckning
- Problemspecifikation
- Användarhandledning
- Algoritmbeskrivning
- Systembeskrivning
- Lösningens begränsningar
- Problem och reflektioner
- Testkörningar
- Eventuell litteraturförteckning
- Källkod
- Eventuella övriga bilagor

Försättsblad

- Kursens namn
- Laborationsnamn och laborationsnummer
- Ditt namn
- Din E-post adress här på LTU
- Eventuell sökväg till källkod
- Alla handledarnas namn
- Datum

- Lämna utrymme för kommentarer

Eventuell sammanfattning (abstract)

- Sällan nödvändigt, men är befogat vid grupparbeten eller annat som gör att rapporten blir omfattande, t ex vid mer uppsatsliknande uppgifter

Innehållsförteckning

- Innehåller alla avsnitt i rapporten
- Innehåller inte sig själv
- Innehåller helst alla bilagor
- Kan ofta genereras automatiskt
- Om man själv gör innehållsförteckningen, se till att den verkligen stämmer
- Se till att rubriker och underrubriker inte ser förvillande lika ut i innehållsförteckningen
- Det kan vara bra med t ex punkter mellan namnet på ett avsnitt och sidnumret

Problemspecifikation

- Ska beskriva vad uppgiften går ut på
- Ska kunna ge en bild av uppgiften utan att man ska behöva läsa hela originalspecifikationen
- Använd egna ord, d v s kopiera inte labbspecifikationen
- Sammanfatta problemet
- Hänvisa till originalspecifikationen
- Gör specifikationen att vissa antaganden måste göras? Ta upp dessa i sådana fall

Användarhandledning

- Beskriver hur man kör din lösning, alla parametrar etc
- Beskriver hur man kompilerar din lösning

Algoritmbeskrivning

- Beskrivning, med vanligt språk, icke-triviala delar av lösningen
- Lämpligt med en numrerad lista (i flera nivåer)
- Försök undvika att använda element som är direkt kopplade till koden, t ex variabelnamn och dylikt
- Syftet med detta avsnitt är att en läsare ska kunna få förståelse för hur en komplicerad del löses utan att behöva lusläsa kod och utifrån denna inse vad som händer
- **Designa dina algoritmer innan du börjar skriva kod!**

Systembeskrivning

- Den mest centrala delen och den som brukar innehålla flest påpekanden
- Ska beskriva systemets interna uppbyggnad och struktur
- Vad som anses viktigt är olika beroende på vilken typ av uppgift/språk man använder
- Ofta bra med figurer som illustrerar centrala delar samt programmets flöde, d v s hur olika delar är sammankopplade
- Bättre med små tydliga figurer än ett enda stort spindelnät

- Beskriv relationer mellan klasser, med figurer och kommentarer till dessa
- Generella riktlinjer
 - Alla egendefinerade datatyper och strukturer beskrivs
 - En beskrivning utifrån ett designperspektiv ska göras

Lösningens begränsningar

- Beskriver alla begränsningar som du kan komma att tänka på, eller har stött på under testningen
- Uppriktighet anses positivt. Alltså bör du tala om de begränsningar som strider mot specifikationen
- Nästan alla lösningar innehåller någon begränsning, tänk till lite bara
- Hur kan/kunde begränsningarna undvikas?

Problem och reflektioner

- Vilka problem har du stött på?
- Vad tyckte du om laborationen?
- Har du några åsikter du vill framföra?
- Finns det personer som varit till särskild hjälp? Tacka dessa
- Detta avsnitt är mycket fritt och kan innehålla nästan vad som helst

Testkörningar

- Försök hitta specialfall
- Kommentera testerna. Vad testas? Tolka resultatet
- Dela gärna upp testerna på ett lämpligt sätt

Eventuell litteraturförteckning

- Normalt ej ett krav
- Bör användas i mer uppsatsliknande rapporter
- Brukar oftast påpekas om det är nödvändigt
- Det finns standarder för hur referenser kan göras

Källkod

- Det är ofta en god ide att bifoga källkoden som en bilaga
- Ska skrivas ut med typsnitt där alla tecken är lika breda, t ex `Courier`
- Koden ska vara indenterad på ett konsekvent sätt
- Koden ska se bra ut även på papper
- Det finns vissa verktyg vars syfte enbart är att skriva ut källkod snyggt (t ex `atp`, `a2ps` och `enscript`)
- Koden ska vara kommenterad där det inte är klart vad du gjort
- Varje funktion/metod föregås av kommentarer som beskriver dess syfte, in-/utdata osv

Eventuella övriga bilagor

- Används vid behov
- Bör finnas med i innehållsförteckningen

Slutliga tips och råd

- Använd ett korrekt och formellt språk

- Normalt är språkvalet svenska/engelska fritt
- Sidhuvud/sidfot innehållandes namn, E-mail, kursnamn, laborationsnamn
- Sidnumrering ska användas
- Rapporten lämnas helst in som pdf. Använd rättstavning
- Om du använder MS Word, se till att den inte är inställd på engelska (i blir I) om du skriver på svenska