

## Förändring av examinationen i HI1024 – Programmering grundkurs

Tidigare har examinationen sett ut så här: man redovisar laborationer lite då och då under kursens gång. Detta följdes under tentaperioden av två tentor, en praktisk (programmeringstenta) och en teoretisk (skriftlig) tenta. Dessa former har haft varsitt problem. Vid laborationsredovisningarna har det inte funnits någon plagiatkontroll och den praktiska tentan har upplevts som många studenter som svår att klara. Även bland de studenter som (enligt mig) har inhämtat kursen på ett bra sätt har haft svårt att klara praktiska tentan.

För att komma till rätta med dessa problem införs två saker:

1. Plagiatkontroll av laborationerna. Ett verktyg införs som kontrollerar de inlämnade laborationerna mot varandra.
2. Den praktiska tentans innehåll modifieras och vi inför tre standardområden som man ska träna på och som utgör ett minimikrav för godkänt. Dessa områden är: *sortering/sökning*, *representation/omvandling* och *strukturering*. Detta konkretiseras på tentamen av att den utgörs av två delar, en A-del och en B-del. A-delen utgörs av (förlagsvis fyra) uppgifter på de tre nämnda standardområden, man måste klara A-delen för att få det lägsta godkända betyget (E). Om man klarar A-delen ger B-delen ett högre betyg (A-D). Om man inte klarar A-delen så rättas inte B-delen. A-delen rättas också noggrannare än B-delen, vi ger mer kvalitativ feedback på A-delen, tips på goda programmeringsvanor etc. I vissa fall kan tillgodoräkning ske, om en student visar att han/hon klarar en uppgift från B-delen så kan en uppgift i A-delen anses som godkänd, detta gäller dock endast vid anspråk på betyget E.

Beskrivning av de tre standardområdena

\* **Sortering/Sökning** – Alla möjliga saker kan sorteras på olika attribut. Vi vill här studera problem som formuleras i termer av *sortering* och/eller *sökning*. “Skriv ett program som sorterar en array av structar med avseende på medlemmen *si* och *så*, eller något attribut som beror på de olika medlemmarna.”

\* **Representation/Omvandling/format/protokoll** – Att *representera* data på många olika sätt, man kan tänka sig att en mängd trianglar sparas i en textfil, en triangel per rad (de tre sidorna som tre flyttal eller heltal) är ett centralt område inom programmering. En omvandling skulle kunna vara att skapa en binärfil och lagra samma trianglar med någon form av urval/klassificering, låt oss säga att endast de rätvinkliga trianglarna sparas. Det är också värdefullt att undersöka en datamängds konkreta format för att söka säkerställa ett materials integritet. Detta är även grunden till applikationsprotokoll.

\* **Strukturering** – En av kursens viktigaste teman är *funktioner* och *structar*, att skriva om ett givet program så att det använder sig av funktioner och structar tycker jag skulle kunna vara en lämplig uppgift, här kan man utgå från ett program som har många upprepade manövrar – dessa läggs i ett funktionsanrop, och arbetar med datastrukturer som tex en personuppgift lagrad i tre skilda strängar. Lämpligt här vore förstås att lägga ihop dem i en struct av typen

```
struct person {
    char fornamn[20];
    char efternamn[20];
    char telefonnummer[20];
};
```

Anmärkning: Inom området *Artificiell Intelligens* brukar man tala om att ett problem i allmänhet har sin lösning i hur man *representerar* problemet i en datastruktur och hur man sedan applicerar olika *sökfunktioner* på den. Det tycker jag talar för att sökning och representation ska vara betonade.