

Programmering og Problemløsning

Datalogisk Institut, Københavns Universitet

Uge(r)seddel 8 - gruppe opgave

Jon Sporring

4. - 25. november.

Afleveringsfrist: onsdag d. 30. november kl. 22:00

I denne periode skal I arbejde grupper. Formålet er at arbejde med et stort projekt der skal dokumenteres ved en rapport. Perioden har ingen øvelsesopgaver, og afleveringsopgaven er:

8g.0 I skal programmere spillet Mastermind.

Spillet Mastermind er et spil for 2 deltagere, en koder og en kodeløser. Koderen laver en skjult ode bestående af en kombination af 4 farvede kdestifter i ordnet rækkefølge, hvor farverne kan være rød, grøn, blå, lilla, hvid og sort. Kodeløseren skal nu forsøge at gætte koden. Dette gøres ved, at kodeløseren foreslår en kombination af farver, og til hvert forslag svarer koderen med et antal sorte og hvide Svarstifter. Antallet af sorte Svarstifter svarer til hvor mange af kdestifterne, som havde den rigtige farve og er på den rette plads, og antallet af hvide Svarstifter svarer til antallet af kdestifter, som findes i koden men på den forkerte plads. F.eks. hvis koden er (rød, sort, grøn, blå), og gættet er (grøn, sort, hvid, hvid) er svaret 1 sort og 1 hvid.

Minimumskrav til jeres aflevering er:

- Programmet skal både kunne være koder og kodegætter
- Programmet skal dokumenteres efter kdestandarden
- Programmet skal afprøves
- Opgaveløsningen skal dokumenteres som en rapport skrevet i LaTeX på maksimalt 20 oversatte sider ekskl. bilag, der som minimum indeholder
 - En forside med en titel, dato for afleveringen og jeres navne
 - En forord som kort beskriver omstændighederne ved opgaven
 - En analyse af problemet

- En beskrivelse af de valg, der er foretaget inklusiv en kort gennemgang af alternativerne
- En beskrivelse af det overordnede design, f.eks. som pseudo-kode
- En programbeskrivelse
- En brugervejledning
- En beskrivelse af afprøvningens opbygning
- En konklusion
- Afprøvningsresultatet som bilag
- Programtekst som bilag

Gode råd til opgave er:

- Det er ikke noget krav til programmeringsparadigme. I må også gerne blande, men overvej i hvert tilfælde hvorfor I vælger det ene fremfor det andet.
- Det er ikke noget krav, at opgaven benytter en grafisk brugergrænseflade, og det anbefales, at I starter med at lave en tekstbaseret version, før I overvejer evt. at lave en med grafisk brugergrænseflade.
- Programmet som koder er måske den nemmeste del, så det er måske et godt sted at starte.
- For programmet som kodeløser er det nyttigt at tænke over følgende: Det totale antal farvekombinationer, som et gæt kan være, er $6^4 = 1296$, og der er ikke noget “intelligens”-krav. Uanset jeres ambitionsniveau vil det være nyttigt først at implementere en simpel strategi såsom at trække et tilfældigt gætte, som ikke har været prøvet før. Hvis I har mod på og tid til en mere avanceret strategi, så kan I overveje, hvilke gæt af dem, som endnu ikke er prøvet, er ugyldige med allerede givne svar.
- I skal overveje detaljeringsgraden i jeres rapport, da I ikke vil have plads til alle detaljer, og I er derfor nød til at fokusere på de vigtige pointer.
- Det er godt først at lave et programdesign på papir inden I implementere en løsning. F.eks. kan papirløsningen bruges til i grove træk at lave en håndkøring af designet. Man behøver ikke at have programmeret noget for at afprøve designet for et antal konkrete situationer, såsom “hvad vil programmet gøre når brugeren er kodeløser, koden er (rød, sort, grøn, blå) og brugeren indtaster (grøn, sort, hvid, hvid)?”
- Det er ofte sundt at programmere i cirkler, altså at man starter med at implementere en hel skald, hvor alle de væsentlige dele er tilstede, og programmet kan oversætte og køre uden at alle delelementer er færdigudviklet. Derefter går man tilbage og tilføjer bedre implementationer af delelementer som legoklodser.
- Det er nyttigt at skrive på rapporten under hele forløbet i modsætning til kun at skrive på den sidste dag.
- Husk at rapporten er et produkt i sig selv: hvis vi ikke kan læse og forstå jeres rapport, så er det svært at vurdere dens indhold. Kør stavetkontrol, fordel skrive og læseopgaverne, så en anden end den der har skrevet et afsnit læser korrektur på det.

- Det er bedre at aflevere noget end intet.
- Der er afsat 1 undervisningsfri og 3/2 alm. undervisningsuger til opgaven (7/11-30/11 fraregnet 14/11-20/11, som er mellemuge, og kursusaktiveter på parallelkurser). Det svarer til ca. 40 timers arbejde. Brug dem struktureret og målrettet. Lav f.eks. en tidsplan, så I ikke taber overblikket over projektforsløbet.

Afleveringsopgaven skal afleveres som fsx-tekstfil navngivet `8g0a.fsx`, som skal kunne oversættes med fsharp, og resultatet skal kunne køres med mono. Funktioner skal dokumenteres ifølge dokumentationsstandard, og løsningen skal dokumenteres vha. en rapport i pdf-format med navn `8g0a.pdf`. Det hele skal samles i en zip fil og uploades på Absalon.

God fornøjelse.