

Laboration 1

Uppgift 3B

Strategi

- Börja i god tid för att undvika stress.
- Läs igenom uppgiften och reflektera över vad som ska göras.
- Utgå ifrån uteslutningsmetoden över vilka metoder och lösningar man kan använda sig utav.
- Skriv så mycket utav koden som möjligt utifrån eget kunnande.
- Felsök med hjälp av internet och den kurslitteratur vi haft tillgång till sedan tidigare.

Delmoment

- Läs igenom instruktionerna
- Analysera problemet
- Gör en grov skiss över hur problemet ska lösas
- Ta reda på information som saknas
- Utveckla skissen, planera mer utförligt
- Koda – Tänk på: Vilka variabler krävs att jag har med. Undvik slarvfel. DRY.
- Går koden att förenkla ytterligare?
- Eventuellt felsök.
- Skriv kommentarer
- Reflektera.

Planerad tid : 4h 0 min

Praktisk tid : 3 h 45 min

Avvikelser från planeringen :

Blev klar 15 minuter tidigare än planerat.

Reflektioner :

Då jag och matematik aldrig varit direkt goda vänner fick jag börja denna uppgiften med att påminna mig själv om vad exakt ett bråktal var och hur man räknade med sådana. Några gamla anteckningar från ett gammalt högskoleprov visade sig komma till stor användning här och efter att jag insett vad det var jag förväntades få programmet att göra kunde jag gå över till att lära mig vad de olika komponenterna i ett bråktal hette på engelska för att undvika fel i programmeringen och även kunna leta efter tips på hur jag skulle gå till väga i engelska forum etc.

Även om jag hade formeln för hur man räknade ut summan/multipliceringen av två bråktal bland mina högskoleanteckningar så lyckades jag hitta ett ännu mer förenklat sätt att skriva det på via hemsidan angiven här nedan. Annars hade jag planerat att räkna ut en del av ekvationen för sig med nya variabler angivna för varje uträkning för att till slut spara ner det färdiga resultatet i en slutgiltig variabel som skulle returnerats.

De sista 45 minuterna av min kodning gick åt till att försöka förstå varför min metod `isEqual` inte fungerade som den skulle. Trots att jag enligt angivna formler i mitt högskole-häfte så visade den att alla talen var lika. Det var först när man angav en täljare större än nämnaren som returneringsvärdet ändrades. Det var efter det jag insåg att min metod inte tog hänsyn till decimalerna i uträkningen utan bara tittade på heltalet, då de var utav typen `int`. Efter en liten ändring i koden så fungerade därefter koden fint.

Totalt sett anser jag att min planering hölls bra, jag klarade mig inom den utsatta tiden, trots en

ganska utdragen felsökning. Jag tog reda på information om vad jag skulle göra och hur uppgiften förväntades lösas innan jag satte mig och började skriva min kod vilket jag tror hade en avgörande effekt på hur väl jag lyckades med uppgiften.

Tidslogg

Tors 20/11 : 1 h 0 min. (Instruktions-analys. Grov skiss. Kodning - Felhantering.)

Fre 21/11 : 2 h 45 min (Kodsök – räkne-exempel. Kodning – uträkning. Felsök – isEqual.)

Refreferenser

<http://msdn.microsoft.com/en-us/library/s53ehcz3.aspx> (Metoderna Add & Multiply)