

Labb 1

Strategi

Tog beslutet att programmera i C# då det är vad vi arbetat mest i och där jag känner inlärningskurvan är lägst.

Jag började med att läsa igenom varje programmeringsuppgift och dela upp den i mindre delar. Sedan satte jag upp dem i "kronologisk" ordning. Vad behöver jag göra först. Vad blir nästa logiska steg osv. Efter det inventerade jag snabbt mina egna kunskaper och erfarenheter. Kunde jag lösa de olika delarna med hjälp av min egen befintliga kunskap eller skulle jag behöva läsa på, googla eller ta hjälp av handledning. Därefter uppskattade jag tidsåtgång för varje del. Skrev planeringen för hand med papper och penna.

När det var gjort kastade jag min in i kodandet och tänkte läsa på, googla eller ta hjälp av handledning allt eftersom problemen uppstod om det uppstod några.

Planering

Nedan följer min planering av varje programmeringsuppgift.

1A) Skriv ett program RaknaA

- | | |
|--|--------|
| - upplägget av programmet | 5 min |
| - skapa ett nytt projekt | 5 min |
| - läs in en textrad från tangentbordet | 5 min |
| - loopa igenom textraden | 15 min |
| - kolla efter 'A' och 'a' | 10 min |
| - testa | 5 min |
| - skriv ut resultatet | 5 min |

1B) Skriv ett program RaknaSiffror

- | | |
|---|--------|
| - upplägget av programmet | 5 min |
| - skapa ett nytt projekt | 5 min |
| - läs in ett godtyckligt tal från tangentbordet | 5 min |
| - kolla jämna, udda och 0:or | 10 min |
| - felhantering | 10 min |
| - testa | 5 min |
| - skriv ut resultatet | 5 min |

1C) Skriv ett program NastStorsta

- | | |
|------------------------------------|--------|
| - upplägget av programmet | 5 min |
| - skapa ett nytt projekt | 5 min |
| - läs in 10 heltal som en sträng | 5 min |
| - dela upp strängen i enskilda tal | 1 tim |
| - testa | 10 min |
| - jämför talen mot varandra | 30 min |
| - testa | 10 min |
| - Presentera det näststörsta | 5 min |

Genomförandet

1A)

Inga större problem med denna uppgift, men fastnade vid loopen av textraden. När jag planerade hur jag skulle lösa uppgiften hade jag klart för mig att jag skulle loopa igenom en array och det har jag gjort förut. Tänkte därför inte att det skulle vara några problem för mig att lösa det delproblemet. Där fastnade jag dock och lade 25 minuter mer på den här uppgiften än planerat.

1B)

Hade stor nytta av den lösning jag gjorde i labb 1A och klarade det därför på utsatt tid. Dock glömmar jag hela tiden att trots att jag vet hur jag ska lösa saker så kan jag det inte i huvudet och får lägga tid på att googla eller leta i gamla projekt. Med tanke på att jag hade mycket gratis från labb 1A borde denna ha gått fortare eller haft ett påslag på 5 min per uppgift för att få rätt på syntaxen.

1C)

Drog 50 minuter över tidsplaneringen. Samma fel igen. Borde ha strukturerat upp de olika momenten mer detaljerat och tagit höjd för de bitar jag har dålig koll på hur jag ska lösa. Här visste jag att utmaningen låg i att lösa uppgiften utan att använda arrayer eller andra datastrukturer. Trots det hastade jag igenom en möjlig lösning i huvudet och kastade mig sedan över kodandet med stor entusiasm. Slarvar mig igenom den ack så viktiga planeringen till förmån för det jag tycker är roligare.

Reflektioner

Nu var detta mindre programmeringsuppgifter och att de tog längre tid att genomföra än planerat blev ingen större katastrof. Om man däremot tänker sig ett mycket större projekt så blir det snabbt ganska mycket "overtid" om två utav tre uppgifter tar längre tid än vad som är tänkt de skulle ta.

Strategin att dela upp varje programmeringsuppgift i mindre delproblem tycker jag fungerar bra. Däremot borde varje del tänkas igenom mer noggrant och alternativa lösningar vägas för och emot varandra. T ex, om jag gör så här, hur kommer det påverka nästa delproblem osv. Vid ett par tillfällen löste jag en liten del av ett problem för att sedan upptäcka att nästa del inte kunde lösas på det sätt jag först tänkt. Absolut uppdelning av uppgifterna, men hela tiden med delarnas samverkan i åtanke. Att veta hur HELA uppgiften ska lösas före man kastar sig på kodandet av första delen är oerhört viktigt.

Dessutom borde jag uppskattat hur mycket min befintliga kunskap kunde hjälpa mig att lösa de olika delarna. Kan jag koda det direkt från huvudet, behöver jag lite hjälp eller har jag inte den blekaste om hur jag ska lösa det? Dessa faktorer spelar stor roll när jag ska uppskatta tiden för förberedelser. Behöver jag läsa på mycket, sätta mig in i nya delar av språket?