

Uppgift 1c, planering och reflektion

Kravspecifikation

"Skriv ett program NastStorsta som läser in tio heltal och hittar det näst största talet av dem och därefter skriver ut det. En körning kan se ut enligt följande:

Mata in 10 heltal: 67 -468 36 1345 -7778 0 34 7654 45 -666

Det näst största talet är: 1345

Försök att göra programmet så att det är lätt att ändra antalet inmatade heltal.

Rekommendation: Använd färre än 10 heltal under tiden ni utvecklar programmet.

Notera: Du får inte använda arrayer (eller andra datastrukturer) i den här uppgiften."

Planering och tidsuppskattning

Planering och förarbete: 1 h,

varav gå igenom uppgiften: 0 h 15 min,

läsa på om metoder för lösningar: 0 h 30 min,

och planera uppgiften: 0 h 15 min.

Genomförande: 1 h 45 min

varav skapa projektfiler: 0 h 15 min,

skriva källkoden: 1 h,

och testning av programmet: 0 h 30 min.

Utvärdering och reflektion: 0 h 45 min

Sammanställa samtliga filer och kontrollera dessa (leverans): 0 h 15 min

Totalt: 3 h 45 min

Utvärdering och reflektion

Den här uppgiften var svår att uppskatta tiden för då jag först var osäker hur jag skulle ta mig an problemet. Jag ville ju använda arrayer! Efter lite googlande och funderande kom jag fram till att jag skulle använda mig av nästlade if- och else if-satser för att hela tiden jämföra varje inmatat tal mot de tidigare största och näst största talen.

Jag trodde att jag hade fått ordning på programmet efter ca 45 minuter, men insåg när jag gjorde tester på lite olika sätt att det inte alls fungerade. T.ex., om jag började med att mata in ett negativt tal var det det talet som blev näst störst oavsett vilka andra tal jag matade in. Efter att ha strukturerat om if-satserna blev mitt andra inmatade tal det näst största oavsett vilka siffror jag än matade in. Det berodde på att jag glömt att ge largestNumber ett nytt värde i min absolut sista else-sats. Jag tänkte alltså inte på att ge den ett nytt värde från omgång tre och framåt - "programmet är ju slut här".

Planering och förarbete gick på halva den uppskattade tiden. Det beror främst på att jag inte behövde all avsatt tid för att hitta ett angreppssätt. Jag var för osäker på hur jag skulle lösa det innan jag började, men när jag snabbt insåg att jag hade en potentiell lösning behövdes inte all den tiden.

Under den här uppgiften har jag kunnat arbeta ostört - ingen telefon som ringer, inga mail o.s.v., vilket gjorde att jag kunde fokusera bättre och ha ett högre tempo.

Att skriva reflektionen tog 30 minuter av de 45 avsatta, och att sammanställa alla filer och leverera tog de avsatta 15 minutrarna.

Total uppskattad tid för uppgiften: 3 h 45 min

Total använd tid för uppgiften: 2 h 20 min

För mer information, se filen *Tidslog labb 1 - Iterativ Mjukvaruutveckling*, som finns i mappen *Övergripande dokument*.

Samtliga filer för denna uppgift finns på:

<https://github.com/lw222gu/1dv404-laboration1/tree/master/Uppgift%201/Uppgift%201c>