LINKÖPINGS UNIVERSITET

TDP005 Projekt: Objektorienterat system

TDP005 - Reflektionsdokument

Författare

Anton Sköld, antsk320@student.liu.se

INSTITUTIONEN FÖR DATAVETENSKAP

Höstterminen 2017
Version 1.0

2017-12-17

1 Revisionshistorik

Ver.	Revisionsbeskrivning	Datum
1.0	Första utkast	2017-12-17

Innehåll

1 Revisionshistorik **1**

2 Reflektioner **2**
2.1 Kursen . . . 2
2.2 Mitt lärande . . . 2
2.3 Svårigheter . . . 2
2.3.1 Segmentation faults . . . 2
2.3.2 Minnesläckor . . . 2
2.4 Samarbetet . . . 3

2 Reflektioner

2.1 Kursen

Denna kurs har förmodligen varit den mest lärorika och roliga kursen ännu.

Vi fick grafikbiblioteket SFML som material, och en beskrivning: Skapa ett arkadspel. Detta gav oss väldigt fria händer, och även en del ansvar.

Man har fått planera och tänka ut en kodstruktur över spelet som är bland annat lättförståelig och utökbar. Detta är olikt något annat projekt i kursen ännu, eftersom det är väldigt praktiskt.

2.2 Mitt lärande

Jag har lärt mig hur man kan utveckla ett projekt så att det matchar en kravspecifikation, och även hur man gör projektet utökningsbart för framtiden.

Jag har även lärt mig mer hur datorspel kan fungera internt, och hur de kan hantera logiken i vardera spelruta.

Detta projekt har även varit en väldigt bra praktisk övning i bl.a polymorfism, hantering av olika pekartyper, och minneshantering. Dessa koncept har varit något suddiga innan projektet, men jag har blivit upplyst över dess användbarhet och syfte.

Användingen av CMake/Make är garanterat något som jag kommer använda mig av i framtiden. Det är jättehäftigt och extremt användbart att kunna sätta upp regler för hur ett projekt hänger ihop och ska kompileras. Introduktionen av dessa metoder kan ha varit det lärorikaste i kursen.

2.3 Svårigheter

2.3.1 Segmentation faults

Den största svårigheten som vi har stött på i projektet var en samling av segmentation faults. Dessa fel började dyka upp i mitten av projektet då spellogiken var något uppsatt.

Problemen var främst logiska fel, som att ett objekt tas bort, och sedan försöker ett annat objekt kollidera med det objektet. Då kommer det andra objektet försöka komma åt en minnesplats som inte finns längre, och då blir det segmentation fault.

Segmentation faults var även väldigt svåra att felsöka, eftersom man som standard inte får någon information om vad i koden de sker. Lösningen till detta var att kompilera med -g flaggan, vilket lägger till debug-information i kompileringen. Man kunde då köra programmet med GDB, GNU Debugger, som pekade på var felen hände.

Lösningarna till dessa segmentation faults varierade, men främst var det att göra några fler kontroller i spellogiken, så att objekten man ville komma åt faktiskt existerade.

2.3.2 Minnesläckor

Några andra grejor vi behövde klura lite på var minnesläckor. När man körde spelet med Valgrind, så dök det upp många minnesläckor vid avslut av spelet. För att fixa dessa fel så lade vi till en hel del skräprensning i spelets game_over()-funktion som tömde alla listor med dynamiskt allokerade objekt.

Denna lösning fungerade bra i slutändan, men en mystisk grej existerar fortfarande. På mitt system, när spelet avslutas så ligger det fortfarande konstant 56 bytes som vi har tappat bort. Det konstiga är att denna minnesläckan verkar inte dyka upp på William's system, och vi har ingen aning alls om vad som kan orsaka den. Alla dynamiskt allokerade objekt ska rensas bort väl vid spelslut.

2.4 Samarbetet

Samarbetet har mestadels gått bra. Våra dygnsrytmer har varit väldigt osynkade, ibland till och med motsatta. Därför så arbetar vi väldigt individuellt på saker, och sedan granskar varandras kod när vi har tid.

Vi delar upp dokumentskapande väl mellan oss, men kodskapandet är något obalanserat om man kollar commit-storlek och mängd. Det obalanserade kodskapandet är förmodligen en följd av ingen konkret uppdelning av kod mellan oss, och möjligtvis en skillnad i motivation eftersom utvecklingen sker främst när vi själva vill.

I framtida projekt bör man strukturera upp kodansvar mellan personer så att det är jämlikt från första början.