

Exercise 7

Software Development 2018
Department of Computer Science
University of Copenhagen

Emil Møller Hansen <ckb257@alumni.ku.dk>,
Casper Bresdahl <Caspers mail>,
Torben Olai Milhøj <Torbens Mail>

Version 1;
Due: Friday, March 16th

1 Indledning

Spillet Galaga er et 2D spil hvor man styrer et rumskib der kan flyve til højre og venstre. Målet er at skyde alle rumvæsnerne inden de når bunden af skærmen. Implementationen vi har lavet er akkumuleret over 3 afleveringer hvor vi efter hver aflevering har fået feedback. Implementationen af spillet er lavet med en game engine kaldet DIKUArcade. På Figure 1 ses et screenshot fra vore implementation af Galaga. De tre opgaver vi har fået har gjort implementationen

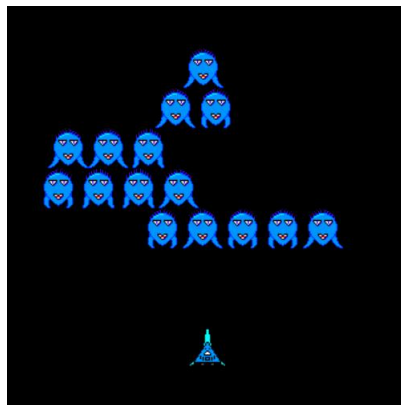


Figure 1: Den sidste implementation af Galaga

mere og mere omfattende. Efter den første opgave havde vi en implementation der lige nøjagtig var spil-bar. Det vil sige man kunne rykke og skyde, men rumvæsnerne rykkede sig endnu ikke. Efter anden opgave fik vi rumvæsnerne til at bevæge sig og de kunne starte ud i flere forskellige formationer, kaldet squadrons. Nu, efter tredje iteration har vi tilføjet en menu, man kan pause spillet og man taber eller vinder når man hhv. ikke når at skyde rumvæsnerne i tide, eller når at skyde dem alle.

2 Baggrund

Som nævnt bruger vi DIKUArcade som game engine til dette projekt. Denne har mange funktioner der hjælper os med implementationen, eksempelvis har denne en collision-detector, som vi bruger til at holde øje med om skudene rammer rumvæsnerne. Et andet element i DIKUArcade vi har brugt meget er eventhandling, som sørger for at forskellige events der sker i spillet, eksempelvis et tryk på piletasterne, bliver håndteret og at der handles korrekt på dette, for eksempelet med piletasterne vil det sige at rumskibet skal rykke sig til en af siderne.

Som implementationen blev mere og mere omfattende igennem de 3 opgaver har der været mere og mere brug for godt design af koden. Dette indebærer eksempelvis at dele programmet op i flere klasser og namespaces i flere forskellige filer og mapper.

3 Analyse

Dette projekt skal gøre brug af DIKUArcade idet der skal bruges mange grafiske elementer. Udover grafiske elementer skal vi bruge mange event-listeners og collision-detecting. Disse er også en del af DIKUArcade. Et spil skal bestå af følgende interaktioner:

1. Start-menu med 2 valg, start eller afslut
2. Der tjekkes for keyboard input, når der trykkes enter vælges den mulighed der nu er aktiv, trykkes der pil op eller ned, skifter den aktive valgmulighed.
3. Spillet starter og der tjekkes igen for input, denne gang for at rykke og skyde.
4. Når der er skudt skud af skal disse tjekkes for kollisioner med rumvæsnerne, hvis der rammes skal skuddet og det ramte rumvæsen fjernes.
5. Når man er i spillet skal man have mulighed for at sætte det på pause ved at trykke ESC, når man gør det skal der komme en menu hvor man har mulighed for at forsætte eller gå tilbage til start-menuen. Valget af dette fungerer ligesom på start-menu.
6. Når der ikke er flere rumvæsener tilbage skal der stå at man har vundet, og herfra skal man have mulighed for at gå tilbage til start-menuen. Hvis der er rumvæsener der når bunden har man tabt og herfra kan man også gå tilbage til start-menuen