

CPROG Rapport för Programmering Projektet

[Gruppnummer: 45]

[Gruppmedlemmar: Marah Zeibak 199601064000]

Skriv en kortfattad instruktion för hur programmering projektet skall byggas och testas, vilka krav som måste vara uppfyllda, sökvägar till resursfiler(bildfiler/ljudfiler/typsnitt mm), samt vad en spelare förväntas göra i spelet, hur figurernas rörelser kontrolleras, mm.

Om avsteg gjorts från kraven på Filstruktur, så måste också detta motiveras och beskrivas i rapporten.

Fyll i 'check-listan', så att du visar att du tagit hänsyn till respektive krav, skriv också en kort kommentar om på vilket sätt du/gruppen anser att kravet tillgodosätts, och/eller var i koden kravet uppfylls.

Den ifyllda Rapportmallen lämnas in tillsammans med Programmering Projektet. Spara rapporten som en PDF med namnet CPROG_RAPPORT_GRUPP_NR.pdf (där NR är gruppnumret).

1.Beskrivning

Spelmotor innehåller många klasser som representera programmet. Klass Session som innehåller run metod som ska anropa i main metod för att loopa och kolla på händelse hela tiden. Sprite klass är superklass till figurer som ska finnas i skärmen och det är abstrakt klass och bara kan vi skapa objekt av sina subklasse dynamisk. Klass Spelare representera en spelare som kan styra av användare genom att röra sig bara till höger eller vänster i piloperator i tangentbordet. Klass Food representera ett objekt som ska röra sig på skärmen hela tiden om den objekt kollidera med spelare (food ska komma åt huvud spelar det vill säga munnen)så ska två saker hända, först ska objekt ändra sitt Y värde och kommer score uppdatera (addera eller minska). I den här klass har jag en metod som ska kolla om den händer kollision mellan spelare och instans av Food eller inte.

Klass Button representera en Knapp som finns på skärmen användare kan styra den här knappen genom att trycka på den med musen. I implementation om användare trycker på knappen så sätta score till 0 i spelet.

Sprite har rörliga figurer som kan röra sig på skärmen (spelare, Food)

Och har orörliga figurer som inte kan röra sig på skärmen (Button, Label).

sökvägar till resursfiler

```
string imagePath = "../../resources/images/";  
string fontPath = "../../resources/fonts/";  
string soundPath = "../../resources/sounds/";
```

2. Instruktion för att bygga och testa

Jag har byggt spelet genom att först skapat instans av Klass Label genom att ange koordinater till var den ska ligga i skärmen en label för att visa ordet Score och en label för att visa vad är Score på den här tiden.

Sedan skapade jag en instans av Klass spelare genom att ange koordinater till den här objekt.

Och en instans av klass Button som ska ta emot en label som argument för att kunna ändra värdet på den här label.

Till sist har jag skapat flera instanser av Klass Food som ta som argument en instans av spelare för att kunna kolla på kollision mellan dem. Och en label och värde som representera hur mycket ska score ändras sitt värde.

3. Krav på den Generella Delen(Spelmotorn)

3.1. [Ja/Nej/Delvis] Programmet kodas i C++ och grafikbiblioteket SDL2 används.
Kommentar: Ja, programmet kör på C++ och jag använder grafik biblioteket SDL2.

3.2. [Ja/Nej/Delvis] Objektorienterad programmering används, dvs. programmet är uppdelat i klasser och använder av oo-tekniker som inkapsling, arv och polymorfism.
Kommentar: Ja, jag har super klass som är Sprite och flera super klasser som ärver från Sprite, jag har använt inkapsling alla medlemmar är privata, och jag har använt arv och polymorfism.

3.3. [Ja/Nej/Delvis] Tillämpning Programmeraren skyddas mot att använda värdesemantik för objekt av polymorfa klasser.
Kommentar: Ja, objekten kan bara skapa dynamiskt. Och jag har ta bort copy konstruktör och tilldelningsoperator.

3.4. [Ja/Nej/Delvis] Det finns en gemensam basklass för alla figurer(rörliga objekt), och denna basklass är förberedd för att vara en rotclass i en klasse hierarki.

Kommentar: Ja, Den heter Sprite.

3.5. [Ja/Nej/Delvis] Inkapsling: datamedlemmar är privata, om inte ange skäl.

Kommentar: Ja, alla medlemmar är privata.

3.6. [Ja/Nej/Delvis] Det finns inte något minnesläckage, dvs. jag har testat och sett till att dynamiskt allokerat minne städas bort.

Kommentar: Ja, objekten som jag har skapat de finns fortfarande i spelet när programmet pågår om det händer kollision med annat objekt jag byter bara Y koordinater och tar inte bort dem, men när vi slutar programmet jag har implementerat destruktör för att städa bort dem objekten.

3.7. [Ja/Nej/Delvis] Spelmotorn kan ta emot input (tangentbord händelser, mushändelser) och reagera på dem enligt tillämpning programmets önskemål, eller vidarebefordra dem till tillämpningens objekt.

Kommentar: Ja

3.8. [Ja/Nej/Delvis] Spelmotorn har stöd för kollisionsdetektering: dvs. det går att kolla om en Sprite har kolliderat med en annan Sprite.

Kommentar: Ja, om spelare har kolliderat med en instans av klass Food så ändra den här instans sitt Y koordinater och det finns label som innehåller score så addera eller minska score beroende på instans. Om användare trycker på knapp så kommer score sättas till 0.

3.9. [Ja/Nej/Delvis] Programmet är kompilerbart och körbart på en dator under både Mac, Linux och MS Windows (alltså inga plattformsspecifika konstruktioner) med SDL 2 och SDL2_ttf, SDL2_image och SDL2_mixer.

Kommentar: Ja

4.Krav på den Specifika Delen(Spelet som använder sig av Spelmotorn)

4.1. [Ja/Nej/Delvis] Spelet simulerar en värld som innehåller olika typer av visuella objekt. Objekten har olika beteenden och rör sig i världen och agerar på olika sätt när de möter andra objekt.

Kommentar: Ja , användare kan styra spelare genom att använda tangentbordet , och instans av klass Food röra sig i skärmen hela spelet , om spelare kollision på Food ändra Food sitt Y värde och addera score till spelare.

4.2. [Ja/Nej/Delvis] Det finns minst två olika typer av objekt, och det finns flera instanser av minst ett av dessa objekt.

Kommentar: Ja

4.3. [Ja/Nej/Delvis] Figurerna kan röra sig över skärmen.

Kommentar: Ja

4.4. [Ja/Nej/Delvis] Världen (spelplanen) är tillräckligt stor för att den som spelar skall uppleva att figurerna förflyttar sig i världen.

Kommentar: Ja

4.5. [Ja/Nej/Delvis] En spelare kan styra en figur, med tangentbordet eller med musen.

Kommentar: Ja, med tangentbordet för att styra spelare och med musen för att trycka på knappen.

[Ja/Nej/Delvis] Det händer olika saker när objekten möter varandra, de påverkar varandra på något sätt.

Kommentar: Ja,