

Dokumentasjon for pokésnake

I denne oppgaven har vi laget et snake spill med pokémon som tema. Dette skulle være en hyllest for at pokémon har vært vist på tv i [20 år](#) i år. Vi tenkte at man skulle fange ulike pokémoners med pokéballer som har “slange”-funksjonen.

Vi fikk utgitt ferdigskrevet prosjekt/kode som vi skulle jobbe ut i fra på Visual Studio. Det vi startet med var å få prosjektet til å kjøre. Første feilmelding var SDL2.lib, som fort ble fikset takket være forelesning 8 (slide 22). Vi installerte SDL2.lib på den manuelle måten, og ikke via “Manage NuGet packages”.

Etter at vi fikk prosjektet til å kjøre, gikk vi gjennom alle *cpp* og *h* filene i prosjektet. Dette ga oss en oversikt over hele prosjektet. Deretter gikk vi nøye gjennom GameManager.cpp, da de fleste endringer skulle bli gjort her. Vårt første mål var å få bildet(smileyboksen) til å bevege seg med en konstant fart, slik at man slipper å holde inne taster for å bevege på den. Da vi fikk slangen til å bevege seg som den skulle, kunne vi starte å implementere alle de andre funksjonene vi skulle ha med som f.eks skatt, hindringer, lengde på slangen osv.

For å få bevegelsen til *slangen* (magnitude + acceleration) spurte vi spillprogrammerere om hjelp. Dette på grunn av manglende kunnskaper om vektorer og bevegelse på intelligente systemer-linjen.

Vår måte å jobbe sammen på var ved å sitte ved siden av hverandre og parprogrammere. Det synes vi har fungert utmerket. Om en av oss satt fast med en funksjon, kunne den andre se etter løsninger. Vi delte opp prosjektet i mindre oppgaver, slik at det ble lettere og mer oversiktlig. En kunne for eksempel jobbe med å få *skatten* til å dukke opp forskjellige steder, mens den andre kunne jobbe med *slangens* logikk.

For å dele kode med hverandre, hadde vi tenkt å bruke versjonskontroll-programmet github. Dette ble en vanskelig sak for oss, da den ene brukte mac med virtuell windows, og den andre med windows. Det fungerte greit i starten, helt til github ikke ville pushe eller pulle på mac'en. Vi prøvde både github, gitkraken og terminal uten hell. Dette problemet oppstod tidligere under PJ3100-uken for mac'en. Løsningen vår for dette var å bruke google docs for å dele koden med hverandre i sanntid. Dette fungerte overraskende bra.

Funksjoner

GameManager.cpp

I play() funksjonen som ligger inne i GameManager ligger hele gameloopen vår. Her har vi lagt til logikken for hva som skal skje når man trykker på piltastene og escape. Det er også her pikachu og diglett blir lagt til i vinduet. Gamemanager lager vinduet og det er her man velger hvilke bitmaps som skal brukes.

GameManager.h

I header filen til GameManager.cpp har vi lagt til nye funksjoner for logikken til spillet.

Snake

Vi har lagd en struct Snake som holder på ulike variabler og funksjoner for *slangen*.

removeObstacle()

Fjerner digletts fra vinduet

obstacle()

Tegner digletts i vinduet ved hjelp av Rand()

pickFood()

Logikken for hva som skjer når man fanger en pikachu.

move()

Logikken for bevegelse

init()

Initialiserer slangen og hvilken retning den skal bevege seg mot

Vi har lagt til disse nye funksjonene for å få til spillogikken og få en mer oversiktlig kode.

Funksjoner/klasser vi ikke har brukt

Vi har ikke brukt andre klasser enn GameManager.cpp / h.

I inputmanager har vi ikke brukt mus-funksjonene. Men vi har valgt å la det være for fremtidig utvikling. For eksempel når man skal navigere rundt på menyen.

Spillet består av:

- Slangen - Pokéball
- Skatt - Pikachu
- Hindring - Diglett
- Lyder
 - Bakgrunnsmusikk
 - Pikachu
 - Diglett

Dette trenger du å vite før du starter spillet:

- Hvis du kjører VS2015 må du høyreklikke på *oppgaven* i solution explorer, velg:
 - Properties
 - Platform
 - Toolset
 - VS2015 v140
- For å kjøre programmet må du navigere deg frem til:
 - Pokesnake
 - Project
 - Oppgaven
 - SDL_Standardproject
 - Oppgaven.exe
- Du er en pokéball (snake)
- Du skal fange pikachu (gul ekorn)
- Jo flere du fanger, jo fortere går pokéballene dine
- Treffer du på en diglett (jordmarken) eller noen av veggene, starter spillet på nytt

Begrunnelse for våre valg

Vi har valgt at diglett skal dukke opp tilfeldig fra starten av, istedenfor en viss lengde. Vi har også valgt at slangen skal gå fort ut i fra hvor mange ganger du har spist pikachu. Grunnen er at vi ønsket å ha en høyere vanskelighetsgrad fra starten av. Hadde vi hatt tid, ville vi nok hatt en startmeny hvor du kan velge nivå (lett, middels, vanskelig) som påvirker når hindringer dukker opp og hastighet på slangen.

Dette kan vi forbedre

- Få bedre flyt, istedenfor å flytte 50 pixler om gangen
- Lagre high score over de beste spillerne
- Lage en meny hvor man kan starte, avslutte, og gå til highscore
- Score counter som vises når man spiller, viser antall pikachu man har fanget
- Power-ups;
 - for eksempel ekstra liv
 - få *slangen* til å gå saktere
- Diglets dukker opp etter en viss lengde på *slangen*

Kilder:

http://lazyfoo.net/SDL_tutorials/lesson08/index.php (Key press)
http://lazyfoo.net/SDL_tutorials/lesson17/index.php (Collision detection)
http://lazyfoo.net/tutorials/SDL/21_sound_effects_and_music/index.php (Sound)
https://www.youtube.com/watch?v=I-HkG0_LY68
https://www.youtube.com/watch?v=E_-lMZDi7Uw
<https://www.youtube.com/watch?v=6dob7qTa7QI&t=296s>
<http://jeanpetric.github.io/2016/03/30/simple-game-development-using-c-and-sdl.html>

Pokémon-relatert

<http://www.imdb.com/title/tt0176385/> (Pokemon fra 1997)
<https://pixabay.com/no/pokemon-pokeball-3d-pokeball-1635610/> (Pokeball)
<http://www.pngmart.com/image/9489> (Pikachu)
<http://nintendo.wikia.com/wiki/File:Diglett.png> (Diglett)
http://www.soundboard.com/sb/Pikachu_Sound (Pikachu sound)
<http://kismusica.info/mp3/dugtrio-voice-sound.html> (Diglett sound)