# **Dokumentasjon av programmet**

Øverst i programmet importeres alle nødvendige biblioteker. Alle nødvendige filer som blir brukt i programmet ligger i mappene: bilder, filer og lydfiler. Programmet er delt inn i 5 klasser og 3 underklasser: Info, Gjenstand (med underklassene Sky, Busk og Platform), Spiller, Skurk og Spill.

Info-klassen er en klasse som brukes for å lage objekter som viser info om det pågående spillet på toppen av skjermen. Klassen brukes for å lage objekter som viser score, liv og tid. Unikt for klassen er derfor at den har attributtene type og verdi, som forteller hvilken type informasjon som vises og verdien av denne informasjonen.

Gjenstandklassen er en klasse som brukes for å lage ulike stasjonære gjenstander i miljøet. I konstruktøren blir det laget et boks-attributt som blir viktige senere når det gjelder kollisjonsdeteksjon. I tillegg er spillobjekt-attributtet viktig for at vi skal få tilgang til de diverse andre objektene i spillet. Klassen har en metode for å bevege objektene. Metoden sjekker om brukeren bruker tastene (og spilleren ikke er på venstre side av skjermen, for da vil heller spilleren bevege seg) og beveger gjenstanden riktig vei. Slik får vi en illusjon av at spilleren beveger seg til tross for at det er gjenstandene som beveger seg på skjermen. De ulike gjenstandene skiller seg litt i konstruktøren og i metoden for å vise objektet og derfor har klassen tre underklasser: Sky, Busk og Platform. Skyunderklassen er en underklasse som brukes for å lage skyer i bakgrunnen. Da velges et tilfeldig bilde og bilde tegnes i vis-metoden. Buskunderklassen ligner på Skyunderklassen, men her velges det naturligvis et tilfeldig busk-bilde. I Plattformunderklassen tegnes heller et rektangel med en tilfeldig størrelse. I tillegg har denne underklassen en metode for å tilfeldig opprette skurker og busker på plattformen.

Spillerklassen er en klasse for å lage spilleren. Spilleren har en beveg-metode som kan bevege spilleren i x- og y-retning. Venstre pil/A vil naturligvis bevege spilleren mot venstre og høyre pil/D vil bevege spilleren mot høyre gitt at spilleren er på venstre side av skjermen, dette henger sammen med logikken for å bevege gjenstand-objektene. SPACE vil naturligvis bevege spilleren i y-retning. Her vil farten i y-retning reduseres med en satt tyngdeakselerasjon for å simulere tyngdekraft. Metoden sjekker også om spilleren har kollidert med en plattform eller en skurk ved å bruke boks-attributtene. Kollisjonsdeteksjon med plattformer passer på at spilleren ikke går gjennom plattformen, men heller holder seg på toppen. Kollisjon med skurk vil drepe spilleren med mindre spilleren lander på toppen av skurken, da vil skurken dø. Vis-metoden stopper spillet og respawner hvis spilleren blir drept og ellers tegner den spilleren som normalt. Animasjonsmetoden brukes i beveg-metoden og den går gjennom ulike bilder av spilleren for å animere bevegelsen. Respawn-metoden avslutter og restarter spillet hvis spilleren har null liv, ellers vil den respawne spilleren med ett mindre liv.

Skurkklassen er en klasse for å lage skurker. Beveg-metoden beveger skurken i forhold til spilleren. Den animerer skurken på lik måte som spilleren. Farten snur retning på enden av plattformen slik at skurken beveger seg frem og tilbake. I vis-metoden forsvinner skurken hvis den blir drept og scoren øker, hvis ikke blir den tegnet normalt.

Spillklassen er en klasse for å lage selve spillet. I konstruktøren startes pygame og pygame.mixer opp. I tillegg lages blant annet infoboksene, spilleren, listen med plattformer, listen med skyer og listen med busker. Plattformens metode for å opprette elementer kjøres for å generere skurker og busker på plattformen. Metoden visStartvindu henter og formaterer dataen fra CSV-filen som lagrer score og tid. Trykker brukeren SPACE endres spill-attributtet til True, trykker brukeren L plottes det en graf over de siste spillene og trykker brukeren I endres info-attributtet til True. Deretter blir ulike tekstbokser tegnet på skjermen. Metoden visInfo tegner opp tekstbokser som viser hvordan å spille. Metoden opprettElementer oppretter tilfeldig genererte skyer, plattformer, busker og skurker når spilleren beveger seg. Metoden kjorRunde oppdaterer spilleren og miljøet ved å kjøre beveg-metoden for hvert av objektene på skjermen. Deretter tegnes hvert av objektene ved å kjøre vis-metoden deres. Metoden spillLokke er selve spilløkka. Da sjekkes det om spilleren har endret spill-attributtet eller info-attributtet til True som vil kjøre henholdsvis kjorRunde- eller visInfo-metoden. Hvis ikke vil visStartvindu-metoden kjøres.

Helt nederst i programmet blir et spillobjekt opprettet og spillLokke-metoden kalles for å starte spillet.