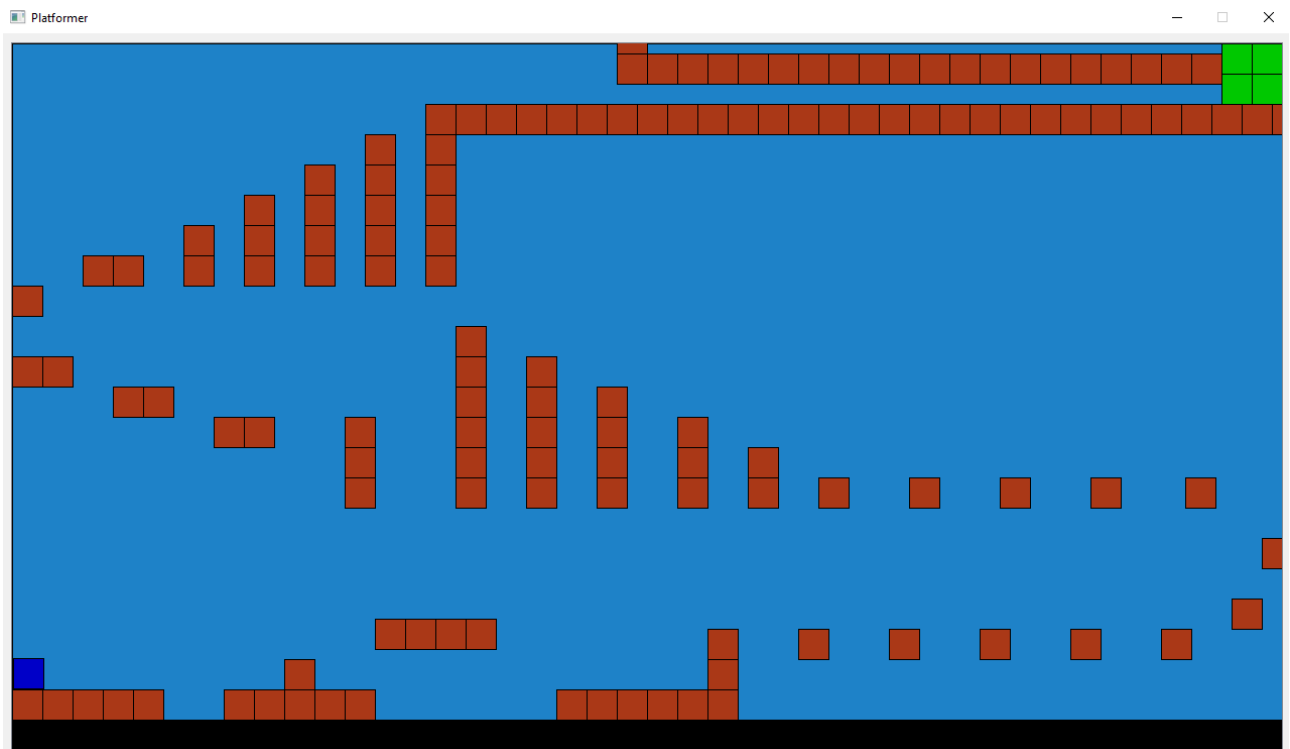


1. Henkilötiedot

Tasohyppelypeli, Sani Letchu, 715036, CHEM, 25.3.2021

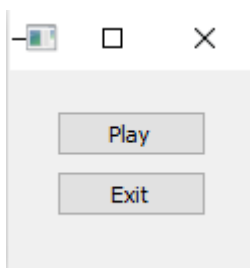
2. Yleiskuvaus

Luotu tasohyppelypeli neliöistä. Tasohyppelypelissä on tarkoitus päästä maaliin, eikä saa tippua alas. Hahmoa liikutellaan AQWED näppäimillä. Pelissä on myös pieni supervoima, jonka avulla hahmo pääsee ahtaisiin paikkoihin. Peli on vaikeustasoa keskivaikea. Supervoima oli uusilisäys suunnitelmasta ja näppäimet Q JA E myös.



3. Käyttöohje

Ohjelma avataan ajamalla tiedosto MainIkkuna.py. Tämä avaa menun, jossa on valintoina Play ja Exit. Exit sulkee ohjelman ja Play avaa isomman ikkunan, johon peli käynnistyy.



4. Ulkoiset kirjastot

Projektissa on käytetty PyQt5 ja unittest kirjastoja.

5. Ohjelman rakenne

Ohjelma on jaoteltu tiedostoihin MainIkkuna, Esine, Pelaaja ja Kartta. MainIkkuna sisältää kaikki käyttöliittymään liittyvät asiat, sekä näppäinten havaitsemiset ja painovoiman ja törmäys havainnoinnin. MainIkkuna myös alustaa kaikki tarvittavat asiat kuten kartan ja pelaajan. MainIkkunassa tarkistetaan myös voitto ja häviö ehdot sekä mahdollisen supervoiman käyttöehdot.

Pelaaja tiedostossa luodaan pelaajahahmo ja se sisältää myös funktion, joka päivittää hahmon sijaintia x ja y akselilla pyydettäessä. Esine tiedostossa on kaikki käytettävät neliö esineet: maali, kuolema, palikka ja tausta. Kartta tiedostossa käytetään Esine tiedoston esineitä, jotka alustetaan omiin x ja y kordinaatteihin. Kartta tiedostossa on pari funktiota, jotka osaavat muodostaa tasoja ja seiniä tietyn määrän ylös tai oikealle päin.

6. Algoritmit

Projektissa käytetään kahta erilaista algoritmia collision detection ja gravitaatio. Collision detection toimii aina kun hahmo yrittää liikkua. Collision detection käy läpi kaikki kartan palikat ja tarkistaa ettei hahmo ja palikat mene päällekkäin. Tämä on hyvin raskas menetelmä, mutta toteutus oli helppo ja peli näyttää pyörivän hyvin. Gravitaatio tarkastaa onko hahmo maassa vai ei, jos hahmo ei ole maassa alkaa hahmo kiihtymään alaspäin loputtomasti, kunnes hahmo osuu maahan. Gravitaatio funktiota käytetään hyväksi myös hyppäämisessä, luomalla aluksi negatiivinen kiihtyminen alaspäin.

7. Tietorakenteet

Ohjelmassa käytetty pythonin omia listoja. Ohjelmassa on käytetty PyQt5 scene tietorakennetta, johon on tallennettu kaikki näytöllä näkyvät esineet ja hahmot.

8. Tiedostot

Ohjelma käyttää apunaan kartta1.txt tiedostoa, jonka avulla voidaan muokata karttaa kirjoittamalla tekstitiedoston riveille taso tai seinä, haluttu määrä, xtaso, ytaso.

9. Testaus

Koodia tehdessä testaus tehtiin manuaalisesti käyttöliittymän avulla Esim. katsomalla toimiiko gravitaatio oikein tai toimiiko törmäys funktio. Tein pari yksinkertaista yksikkötestiä, joissa testataan, että kartta luo oikean määrän palikoita, sekä testin, joka testaa, että hahmon liike ja sijainti toimii oikein.

10. Ohjelman tunnetut puutteet ja viat

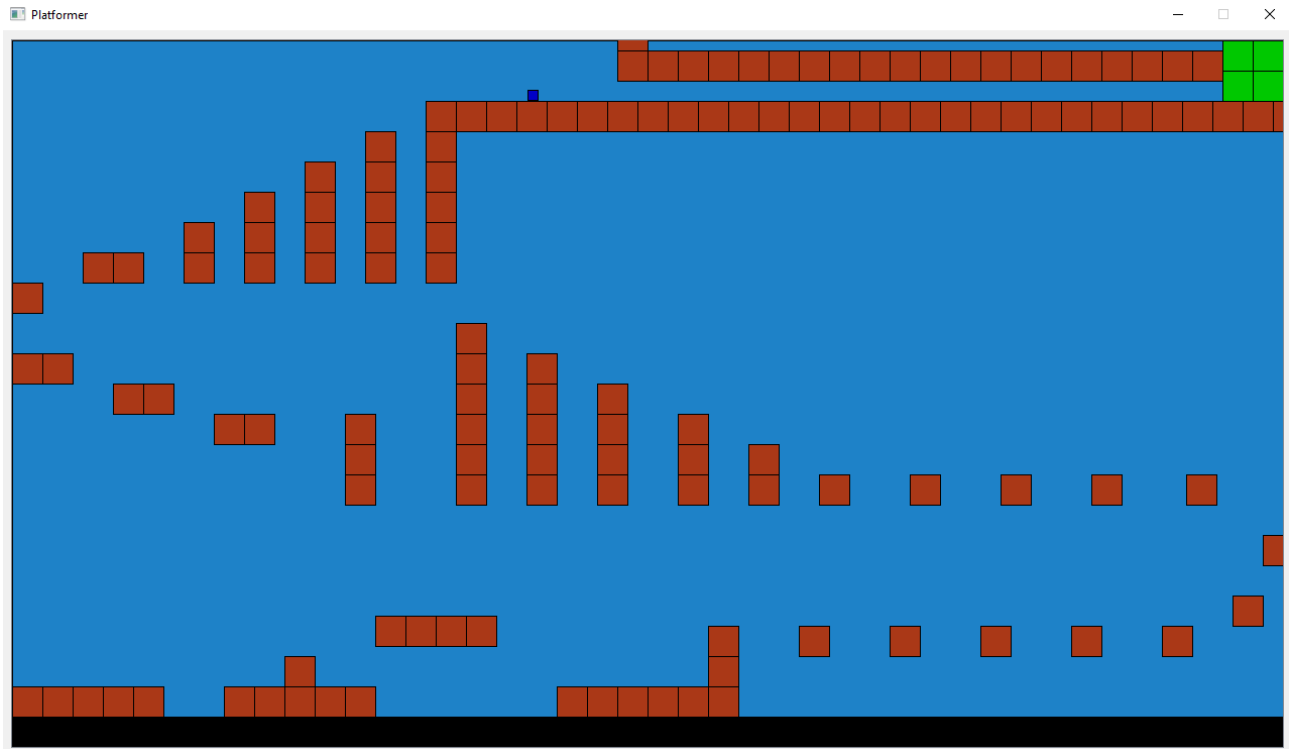
Yhdessä kohtaa karttaa on mahdollista tehdä tuplahyppy, vaikka sitä ei ollut erikseen tarkoitettu. Mahdollisesti vika on gravitaatio funktiossa.

11. 3 parasta ja 3 heikointa kohtaa

Mielestäni gravitaatio on toteutettu hyvin ja näyttää hyvältä. Huono: Supervoima toimii vain yhteen suuntaan ja on käytettävissä ylhäällä, Collision detection on hyvin raskas suorittaa, koska funktio varmistaa, että hahmo ei mene päällekkäin minkään palikan kanssa, joka kerta kun hahmo liikkuu.

12. Poikkeamat suunnitelmasta

Lisäsin supervoiman ja lisäsin liikkumisnäppäimet Q ja E. Hahmon muutos supervoiman jälkeen.



13. Toteutunut työjärjestys ja aikataulu

Suunniteltu työjärjestys ja aikataulu eivät vastanneet toisiaan. Suunnitelman mukaan piti tehdä noin 20 riviä koodia per viikko. Toteutus kuitenkin tapahtui noin viikossa, jolloin koodi oli jo lähes valmis.

14. Arvio lopputuloksesta

Ohjelma on mielestäni hyvä. Liikkuvuus voisi olla hieman sujuvampaa. Puutteet ja mahdolliset parannukset mainittu jo hieman kappaleessa 11.

15. Viitteet

Kurssin ohjelmaesimerkit, joissa käytetään PyQt5, sekä doc.qt.io -sivusto.

16. Liitteet

Ei käytettyjä liitteitä