

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
SVEUČILIŠTE U ZAGREBU

Računalna animacija

3. laboratorijska vježba

Jakov Kruljac

0036501190

19. siječanj 2022

1. Uvod

Tema posljednje laboratorijske vježbe bila je prikazati naučeno na kolegiju Računalna animacija tako da implementiramo neku problematiku vezano uz gradivo. U ovom radu obrađene je tema detekcije sudara te izračunavanje nove putanje objekata nakon sudara. Za demonstraciju rješenja kreirana je jednostavna računalna igra „Chicken invaders“ . Igra je implementirana u programskom jeziku Python, a za prikaz animacija koristio se paket `pyglet`.

2. Pokretanje igre

Igra je izrađena u verziji Pythona 3.8, a dodatno se koristi i paket `pyglet` verzije 1.5.21. Za pokretanje igre potrebno je otvoriti naredbeni redak ako se koristi operacijski sustav Windows ili terminal ako se koristi operacijski sustav Linux, pozicionirati se u direktoriji projekta te napisati naredbu „`python game.py`“.

3. Opis igre

3.1 Cilj igre

Cilj igre je da igrač koji upravlja svemirskim brodom pogodi sve mete prije nego što ga neka od meta pogodi svojim metkom.

3.2 Igrač

Sve pojedinosti o igraču nalaze se u klasi *SpaceShip*. Igrač na početku ima definiran broj života i početne koordinate na ekranu. Klasa kao atribut također ima teksturu slike koja prikazuje igrača. Igrač u trenutku može imati najviše 10 metaka koji nisu pogodili cilj ili granicu polja za igru. Igrač izgubi život kada ga pogodi metak. Funkcija klase *SpaceShip is_shoot(EnemyBullet)* u prilikom svakog osvježavanja ekrana provjerava jeli pozicija metka unutar kvadrata kojim je definiran igrač. Kvadrat je definiran donjom lijevom točkom teksture te njegovom duljinom stranice(širina teksture)



Slika 1. Prikaz igrača i njegovog metka

3.3 Meta

Meta je definirana klasom *Enemy*. Klasa u sebi sadrži attribute kao što je početna pozicija, kut pod kojim se meta kreće, vrijeme između dva metka kao i vrijeme proteklo od zadnjeg ispucanog metka. Mete se u igrici prikazuju u slojevima.

Svaki sloj zadužen je za praćenje: broja meta u sloju i sve metke ispaljene od strane meta tog sloja. Igra podržava tri vrste sloja: *LinearLayer*, *CircleLayer* i *BossLayer*. Svaki sloj ima neke svoje karakteristike, kao što su veličina mete ili kretanje mete.

U sloju *LinearLayer* sve mete se kreću pod istim kutom pravocrtno. U tom slučaju potrebna je detekcija sudara mete s granicom sloja. U slučaju sudara mete s nekom od granica svim metama u sloju kut se promijeni za 180 stupnjeva. U sloju *CircleLayer* mete se kreću u krug te tu nisu potrebne nikakve provjere. U sloju *BossLayer* mete su duplo veće, imaju 5 života te se svaka meta kreće pod različitim kutom. U ovom sloju potrebno je detektirati sudare između meta kao i sudar mete s granicom sloja. Kako je u igri bitnije da se dvije mete ne preklope nego da se metama omogućiti da dođu što bliže jedna drugoj, radi jednostavnosti i bržeg računanja detekcija sudara između dvije mete se vrši tako da se oko svake mete opiše kružnica i onda se ispituje odnos dvije kružnice, u slučaju da kružnice imaju jednu ili dvije zajedničke točke svakoj meti kut se promijeni za 180 stupnjeva. U svakom sloju vrši se i detekcija je li meta pogođena od strane svemirskog broda. Ova detekcija ista je kao i detekcija metka i svemirskog broda.



Slika 2. Prikaz mete i njezinog metka



Slika 3. Prikaz igre s dva sloja, *LinearLayer* i *BossLayer*

4. Upute za igru

Pokretanjem programa automatski počinje i igra. Igrač se kreće s tipkama W, A, S, D dok metke ispaljuje s tipkom SPACE. Kad igra završi bilo pobjedom igrača ili meta program se automatski ugasi.