

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tomaž Rak |  | Miha  Kirbiš |  | Marko  Črnko |

Poročilo SIGMA

PROGRAMSKI VZORCI - Vaja 2

Načrtovanje

Razmislek

Ugotovili smo, da za ta program, potrebovali za premikanje v prostoru 3 dimenzionalna polja, ki bodo beležila našo pozicijo, in naše trajektorje.

1. Korak

V orodju Unity smo izdelali preprost model drona, z imenom »DroneParent«. Ta je vseboval telo in 4 propelerje. Nato smo zanj še ustvarili c# skripto »DroneMovementScript«.

2. Korak

V omenjeni skripti smo definirali »ourDrone«(Rigidbody telo drona), »trenuten«(int zap. št. Trajektorja v seznamu) in »hitrost«(float hitrost premikanja drona) ter List »trajektorji«(Vector3 cilji drona), ki so ključniga pomena za simulacijo.

V funkciji »Update« povemo da se naj pozicija drona spremeni v koordinate, ki so bljižje naslednjemu trajektorju. In ko prispe, se naj začne približevati naslednjemu trajektorju, ki je v našem primeru za lažjo načrtovanje bil »Empty Game Object« z pozicijo.

3. Korak

Nato smo se odločili, da trajektorjev nebomo dodajali, kot objektov v Unityu, ampak kot »position«, ki je spremenljivka objekta za vodenje drona.  
Dronu smo telo odstranili in mu dodali visoko detajlno telo, ki smo ga najdli na »Asset Store«  
Dodali 4 objekte z teksturo, ki predstavljajo oddajnike.

4. Korak   
Izdelali smo »GameController«, ki je namenjen za branje iz datoteke in osnovno povezovanje.  
Tukaj smo izdelali funkcijo »DodajDrona«, ki dinamično dodaja drone.  
V funkciji »ReadString« smo dodali funkcionalnost, da beremo z datotek, ki nam povedo položaj, rotacijo, hitrost, in pot elementov.

5. Korak(zadnji)  
Sceni smo dodali še gumb, ki dinamično dodaja drone po želji.  
Dodali smo še funkcionalnost »Instance«, ki je primeren za prihoden razvoj.  
Za zaključek smo še kameri dodali ozadje.

Končni rezultat  
Drone ki kroži med lokalnimi cilji.  
Opomba: cilji niso vidni in oddajniki so pobarvani rumeno.



Razredni diagram



