

# Tema GP

## Roșu Mihai Cosmin

### Teren:

Pentru realizarea terenului am folosit fractal noise, cu functia preluata din laboratorul 4, pe care l-am aplicat pe un obiect de tip Terrain.

Am folosit urmatoarele valori pentru parametrii fractal noise-ului:

- Scale = 20;
- Octaves = 5;
- Amplitude = 1;
- Frequency = 1;
- Lacunarity = 3;
- Persistence = 0.35;

### Plante:

Pentru realizarea plantelor am folosit conceptul din laborator, pe care l-am adaptat pentru 3D folosind urmatorul [tutorial](#).

Pentru a desena liniile, in loc de turtle din python, am folosit obiecte cu LineRenderer.

Am folosit urmatoarele reguli:

- F = forward (muta un obiect empty de la pozitia curenta mai in fata si apoi configureaza un LineRenderer intre cele doua puncte)
- + = Rotate right Oz
- - = Rotate left Oz
- \* = Rotate left Oy
- / = Rotate right Oy
- [ = Push position in stack
- ] = Pop position from stack

Am folosit urmatoarele siruri de caractere pentru plante:

- [\*+FX]X[+FX][/+F-FX]
- [F[-X+F[+FX]][\*-X+F[+FX]][/-X+F[+FX]-X]]
- [F[+FX][\*+FX][/+FX]]

### NPC si Items:

Pentru NPC si Items am folosit implementarea din laboratorul 6.

Fiecare NPC are un nume, o clasa, o personalitate, viata si damage. Pentru nume, clasa si personalitate am creat o lista de 5 variante din care este aleasă una aleator. Pentru viata si damage este ales un numar dintr-un interval ales de mine.

Fiecare item are un tip(nume), o raritate, damage/defence, durabilitate si o abilitate pentru raritatile mari. Pentru tip si abilitate am creat o lista de 5 variante din care este aleasă una aleator. Pentru raritate este ales un numar intre 0 si 4, de la 2 in sus fiind folosita si abilitatea. Pentru damage/defence si durabilitate este folosit un interval ales de mine.

### **Dungeon:**

Pentru crearea dungeon-ului este folosit algoritmul BSP din urmatorul [tutorial](#).

In urma BSP-ului obtin o lista de camera. Acestea sunt conectate folosind urmatorul algoritm:

1. Se alege o camera din lista;
2. Se cauta cea mai apropiata alta camera;
3. Sunt conectate prin crearea unei linii care unește centrele camerelor;
4. Cea de a doua camera este scoasa din lista si devine lista curenta;
5. Se repeta pasii 2-5 pana nu mai sunt camera in lista.

In final, pentru fiecare punct dintr-o camera sau dintr-un corridor este plasat un tile. In jurul tile-urilor sunt plasati pereti.

Personajul este plasat intr-o camera, un portal in alta camera, si pentru fiecare camera ramasa este ales cu sanse egale sa fie plasat un trap, un cufar cu aur, un cufar cu item sau un inamic.

Trap-ul este pur si simplu un prop, cufarul cu aur are o cantitate dintr-un interval ales, cufarul cu item are un item generat la fel ca mai sus, iar inamicii sunt generati la fel ca un NPC.

### **Conecțare:**

Toate conceptele de mai sus sunt conectate astfel:

Intai este creat terenul, apoi sunt plasate aleator pe teren un numar de plante, apoi sunt plasati un numar de NPC, apoi sunt plasate un numar de iteme si apoi este plasat un portal si player-ul. Daca player-ul intra in portal se va incarca scena cu dungeon-ul. In dungeon, elementele sunt plasate dupa cum e mentionat mai sus, daca jucatorul intra in portalul din dungeon se intoarce in scena principala.