Dungeon Room Graph Generator (GUI), Florian Ionescu, CMPM 147

Overview

RGG2 is a procedural dungeon layout generator that creates abstract room graphs using ASCII visualization. Rooms are represented by characters and connected by corridor lines. The program includes a GUI for adjusting generation parameters and exporting batch outputs to a .txt file.

How To Run

1. Make sure both files are in the same folder:
    - RGG2_GUI.exe
    - SDL2.dll

2. Double-click RGG2_GUI.exe.

Features

- Adjustable total room count
- Adjustable main path length
- Branch probability and branch length control
- Loop/shortcut probability
- Multi-size room footprints (e.g., 3x3 block rooms)
- Weighted dungeon room types:

    - S = Start
    - G = Goal
    - H = Health Room
    - M = Monster Room
    - T = Treasure Room
    - K = Key Room
    - B = Boss Room
    - P = Puzzle Room
    - R = Rest Room

- Batch export to .txt
- Live ASCII preview in GUI
- To generate multiple layouts:
- Set the number of samples.
- Enter an output filename.

Notes

- Different seeds produce different layouts.
- Changing branching and loop parameters significantly alters navigability.
- Larger room sizes create stronger visual landmarks.

Requirements

- Windows (x64)
- SDL2.dll must be in the same folder as the executable.

Core Structure Parameters

rooms

Total number of rooms generated (including Start and Goal).
Higher values create larger, more complex layouts.

mainLen

Length of the critical path from Start (S) to Goal (G).
Increasing this makes the dungeon more linear and directional.
Lower values make the goal closer and branches more dominant.

branchChance

Probability that a room will generate a side branch.
Higher values produce more exploratory layouts.
Lower values create cleaner, more linear dungeons.

Range: 0.0 – 1.0

maxBranchLen

Maximum number of rooms that can appear in a branch chain.
Higher values create deeper side paths.

loopChance

Probability of adding shortcut connections between nearby rooms.
Higher values create more interconnected, maze-like structures.
Lower values keep layouts tree-like and easier to read.

Range: 0.0 – 1.0

Canvas Settings

W (Width)

Horizontal size of the ASCII canvas.

H (Height)

Vertical size of the ASCII canvas.

Smaller canvases restrict layout growth and compress structure.

Room Size Controls (Multi-Room Blocks)

multiRoomChance

Probability that a room will occupy multiple grid cells (e.g., 3x3 block).
Higher values create more landmark-style rooms.

Range: 0.0 – 1.0

minRoomW / maxRoomW

Minimum and maximum room width (in room cells).

minRoomH / maxRoomH

Minimum and maximum room height (in room cells).

Increasing these allows larger rectangular or square room blocks.

Room Type Weights (Dungeon Roles)

These control how often each room type appears.

Higher weight = more likely to appear.

- H – Health Room
- M – Monster Room
- T – Treasure Room
- K – Key Room
- B – Boss Room
- P – Puzzle Room
- R – Rest Room

Weights influence distribution but do not guarantee exact counts.

seed

Controls randomness of generation.
Using the same seed produces the same dungeon layout.
Changing the seed creates a different layout.

Test Output:
https://github.com/Silverxbullet/Dungeon-Layout-Generator/blob/main/batch_output.txt

Use Evidence:

Tester: Cameron Coleman

- Tester really enjoyed the GUI interface.
- Noted a glitch where the dungeon becomes just goal ASCII cells at a certain Multiroom chance threshold (~ 0.68)
- Be sure to ensure that parameters are well explained in README