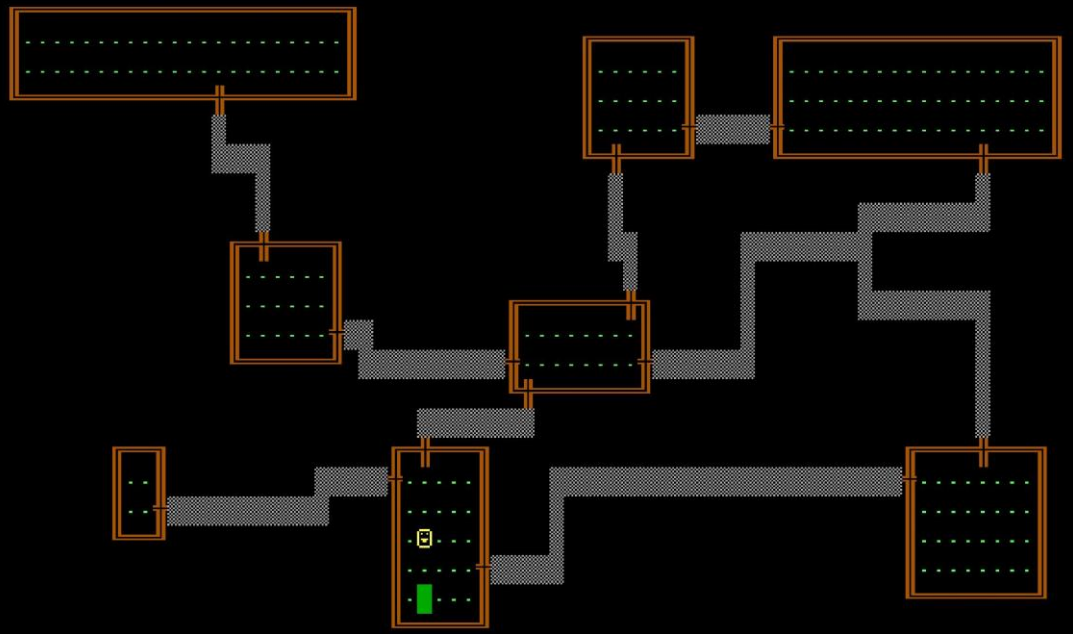
Procedural Generation

What is Procedural Generation

Used for:

- Level & Terrain Generation
- Character Dialogue & Animation
- Object Instantiation
- Loot Systems
- Many More!

How It Started



1980s Rogue – Procedurally Generated Level

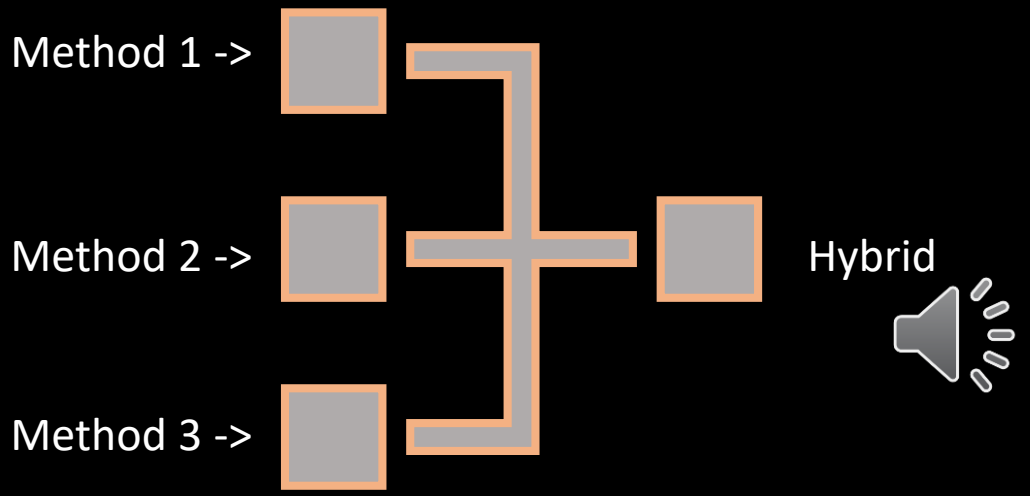
- Every Level is procedurally generated.
- Different every time the game is played.
- Started a new genre: Rogue-Like

How It’s Done

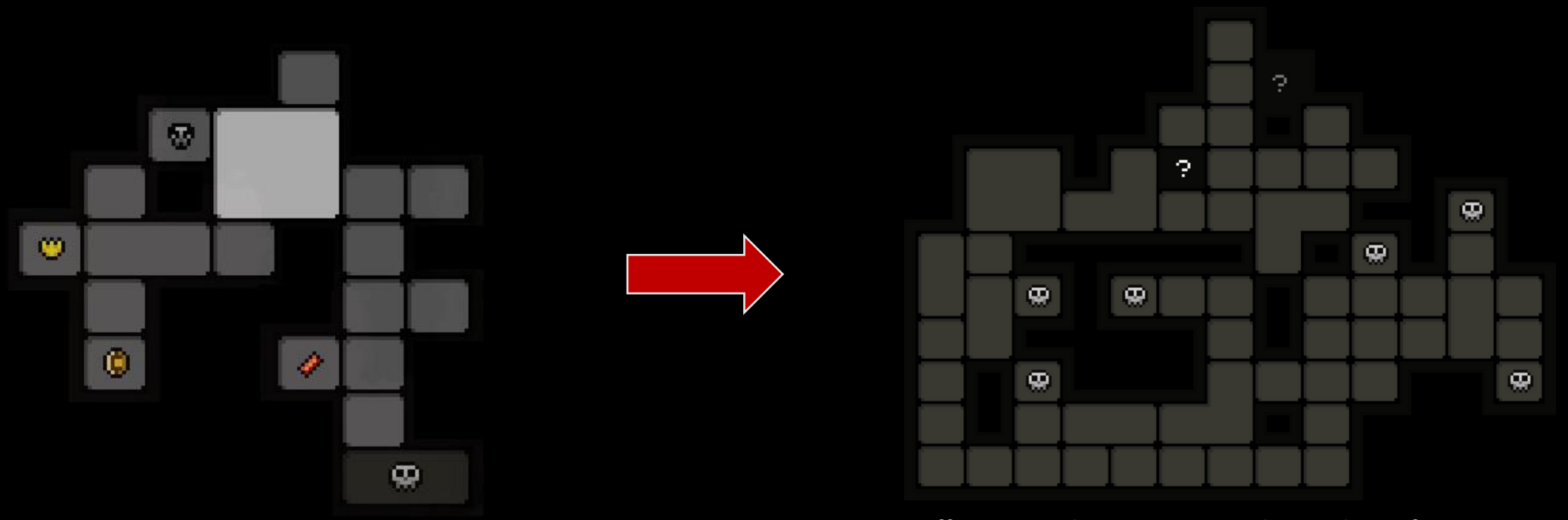
Bunch of known methods:

- Grammars
- Wave Function Collapse
- Markov Chains
- HEAPS MORE!

Sometimes Combined!



Focus Topic = Level Generation



Different Level Generation in The Binding Of Isaac

- Different Every Time = Replay Value++

Proc Gen In Rogue-Likes

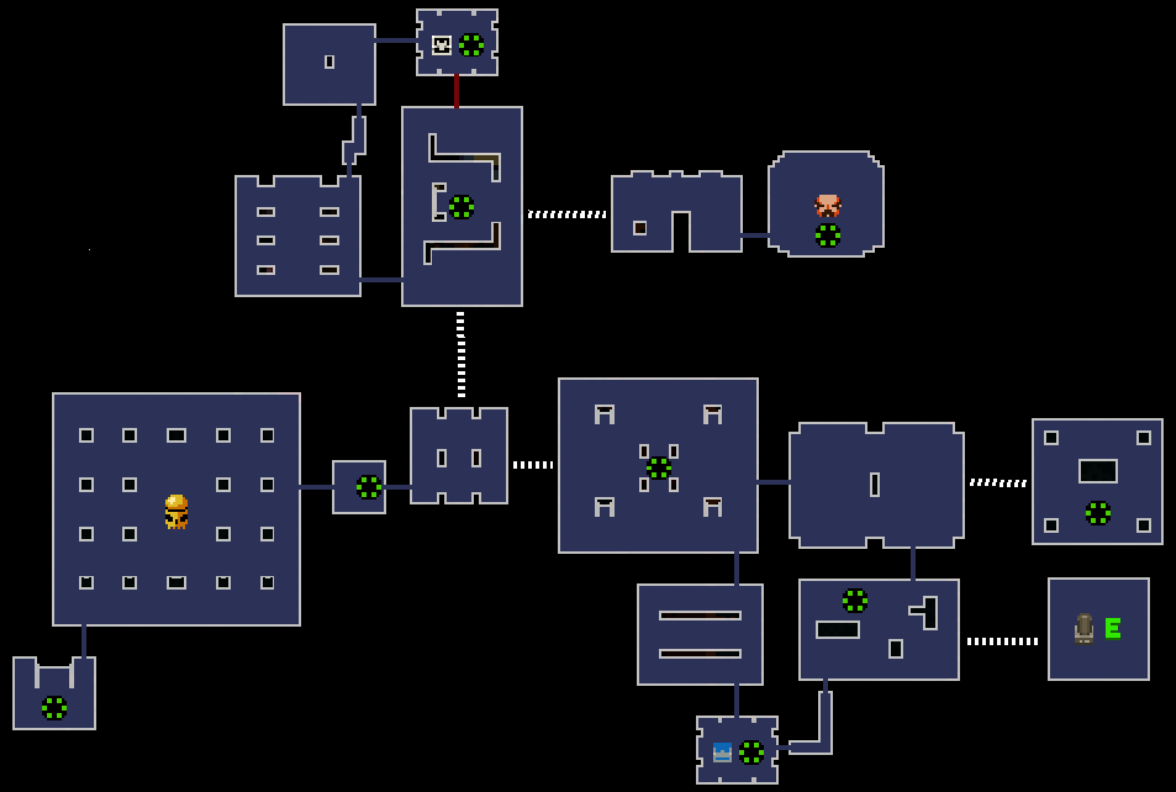
Shared Features In Rogue-Likes:

- Proc Gen for:
  - Levels
  - Items/Rewards
- Different every time
- Perma Death

Proc Gen Level Types

Random Levels With Basic Rules

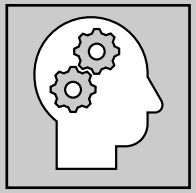
VS



Strict Structure With Slight Variations

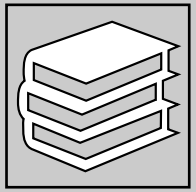
Project Outline

Learning Goals:



Learn & Understand Proc Gen

Learn multiple techniques



Learn to combine them

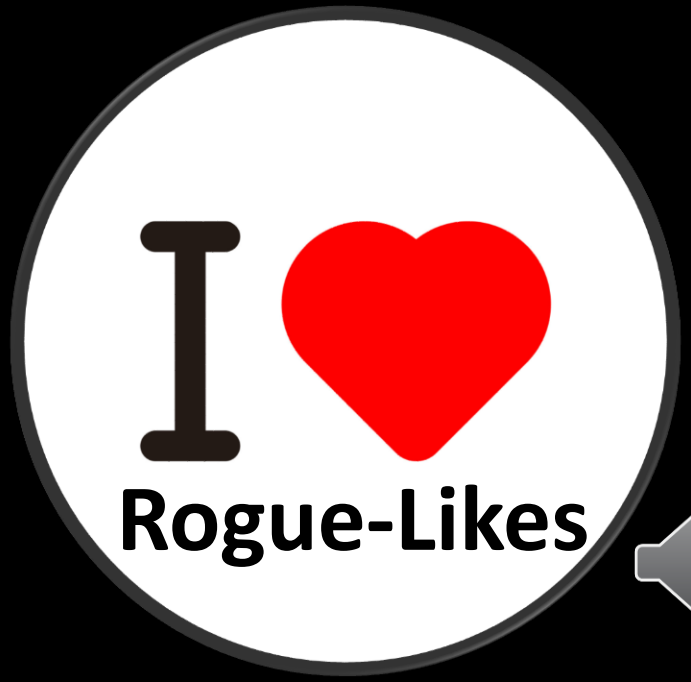
Why Learn This?

I want:

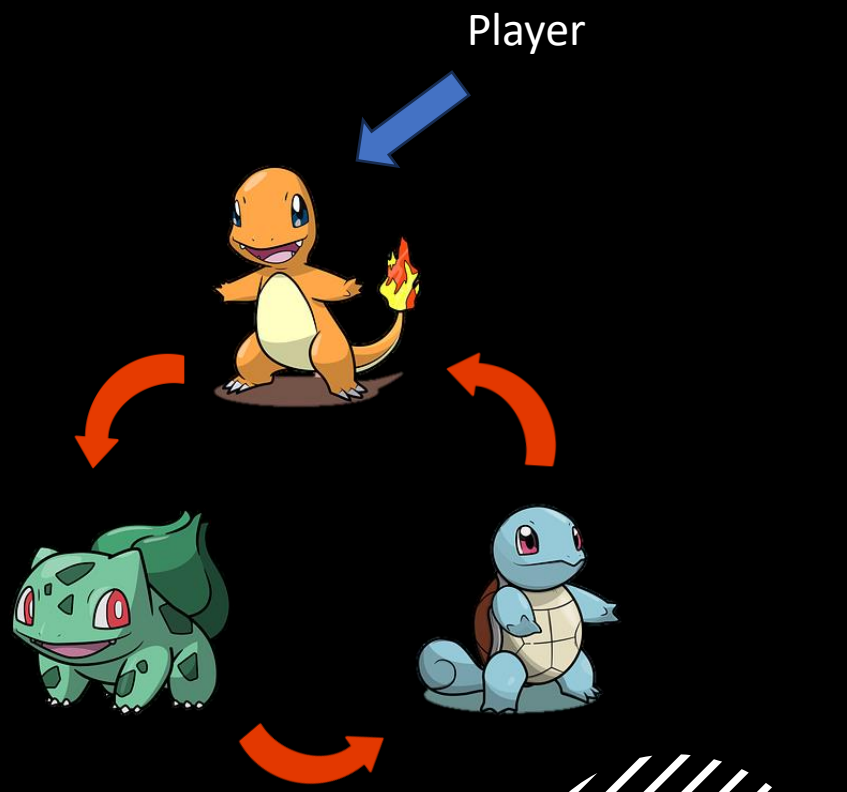
- Make my own Rogue-Like

+

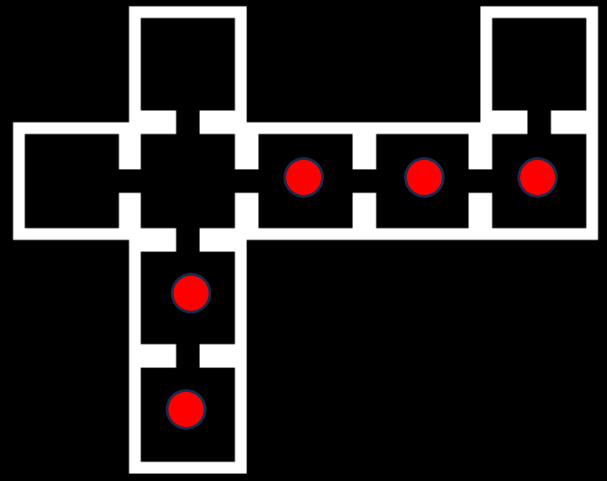
- Reactive Levels



What Does This Mean?



Difficulty = 0

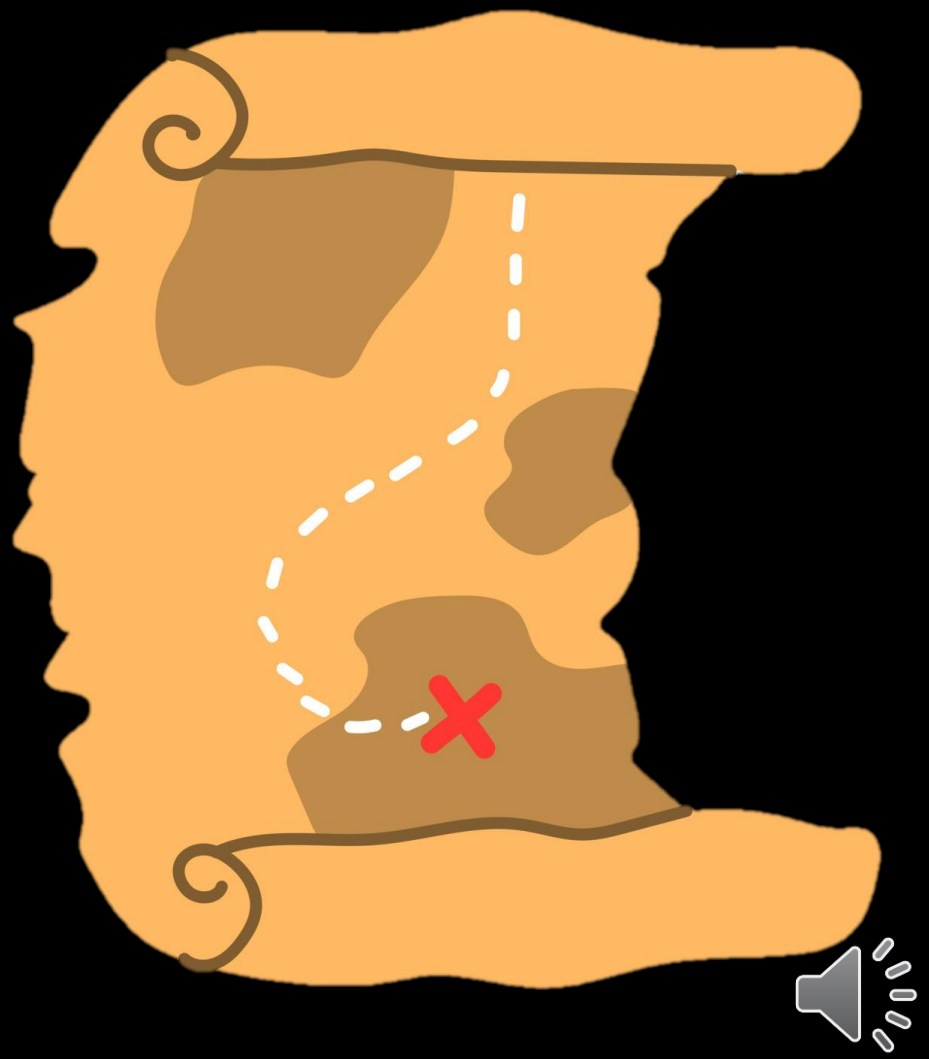


= Fire Enemy

= Water Enemy

Deliverables

Stage 1:	• Few algorithm prototypes
Stage 2:	• Hybrid version that fits me
Stage 3:	• Populate the level



Resources

https://dribbble.com/lwdavisdesigns

https://www.rockpapershotgun.com/how-do-roguelikes-generate-levels

https://www.youtube.com/watch?v=1-HIA6-LBJc

https://dl.acm.org/doi/10.1145/3402942.3402945

https://kentpawson123.medium.com/procedural-generation-an-overview-1b054a0f8d41

https://tuliomarks.medium.com/how-i-created-roguelike-map-with-procedural-generation-630043f9a93f

https://www.boristhebrave.com/2020/09/12/dungeon-generation-in-binding-of-isaac/