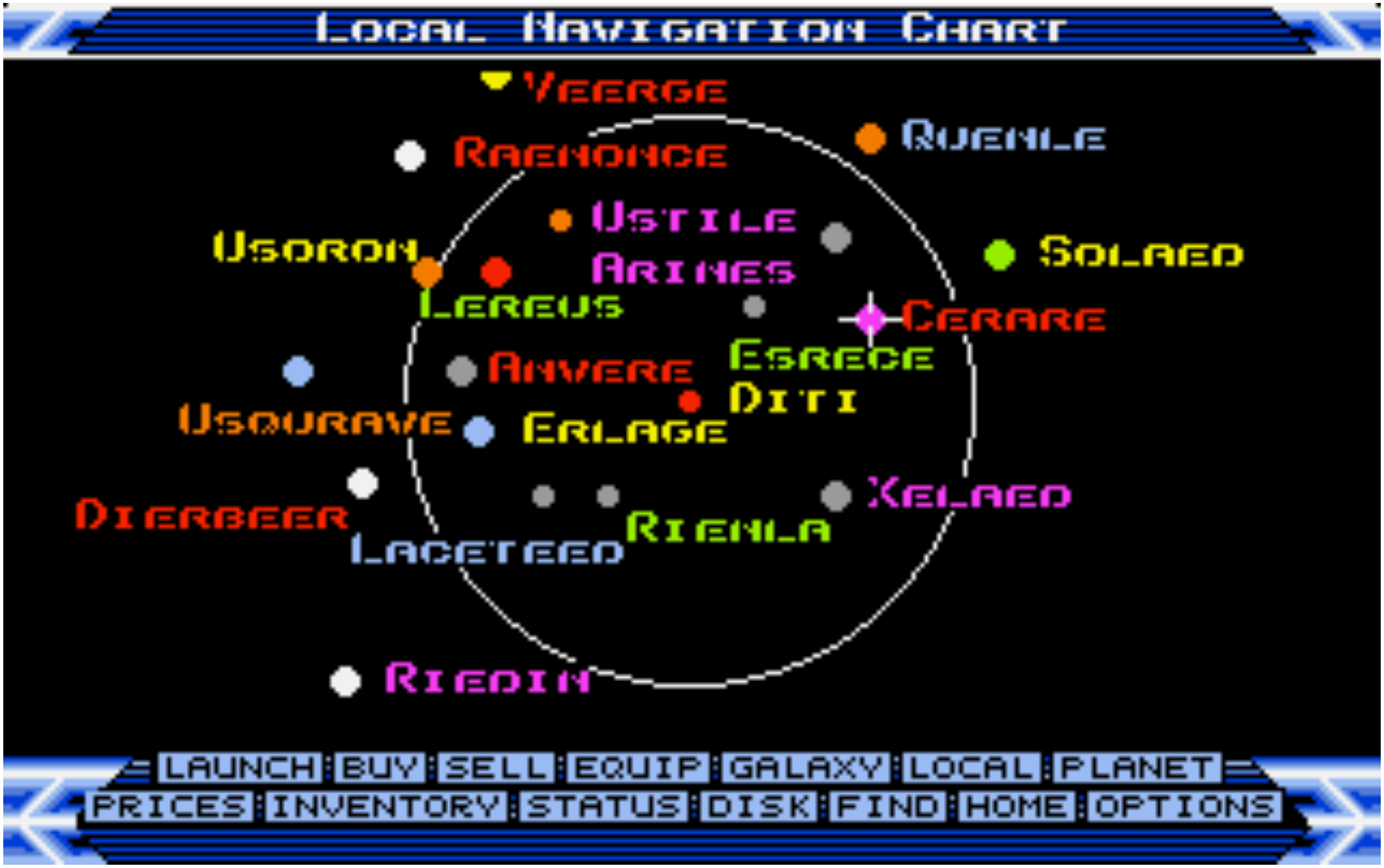
Lecture 4:
Procedural Content Generation

Game Design
Fall 2023

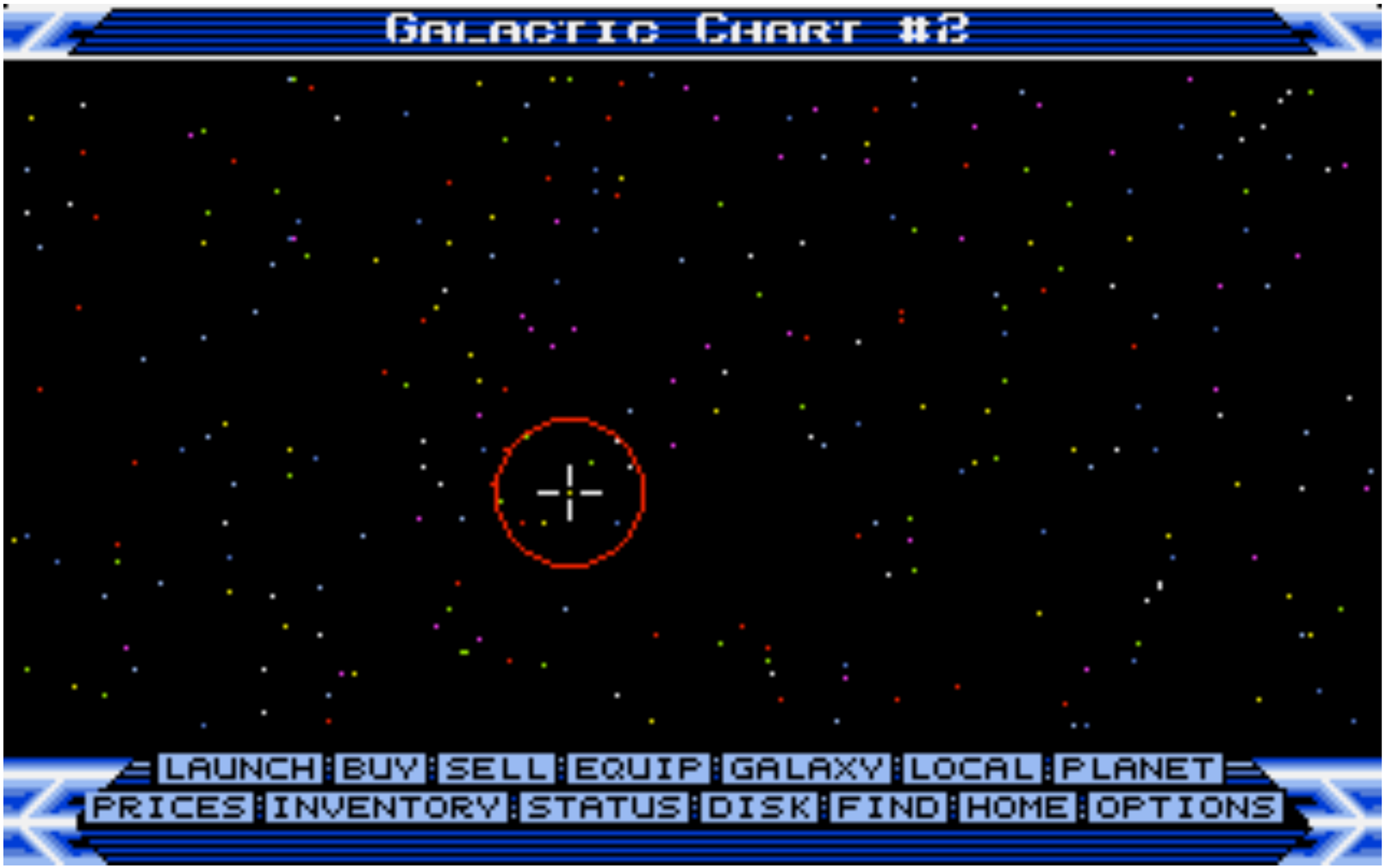
Julian Togelius

Elite

Elite


Local Navigation Chart
Veerge
Raenonce
Quenle
Ustile
Usoron
Aries
Solaed
Lereus
Cerare
Esrece
Anvere
Diti
Usourave
Erlage
Xelaed
Dierbeer
Rienla
Lacteed
Riedin
Launch Buy Sell Equip Galaxy Local Planet
Prices Inventory Status Disk Find Home Options

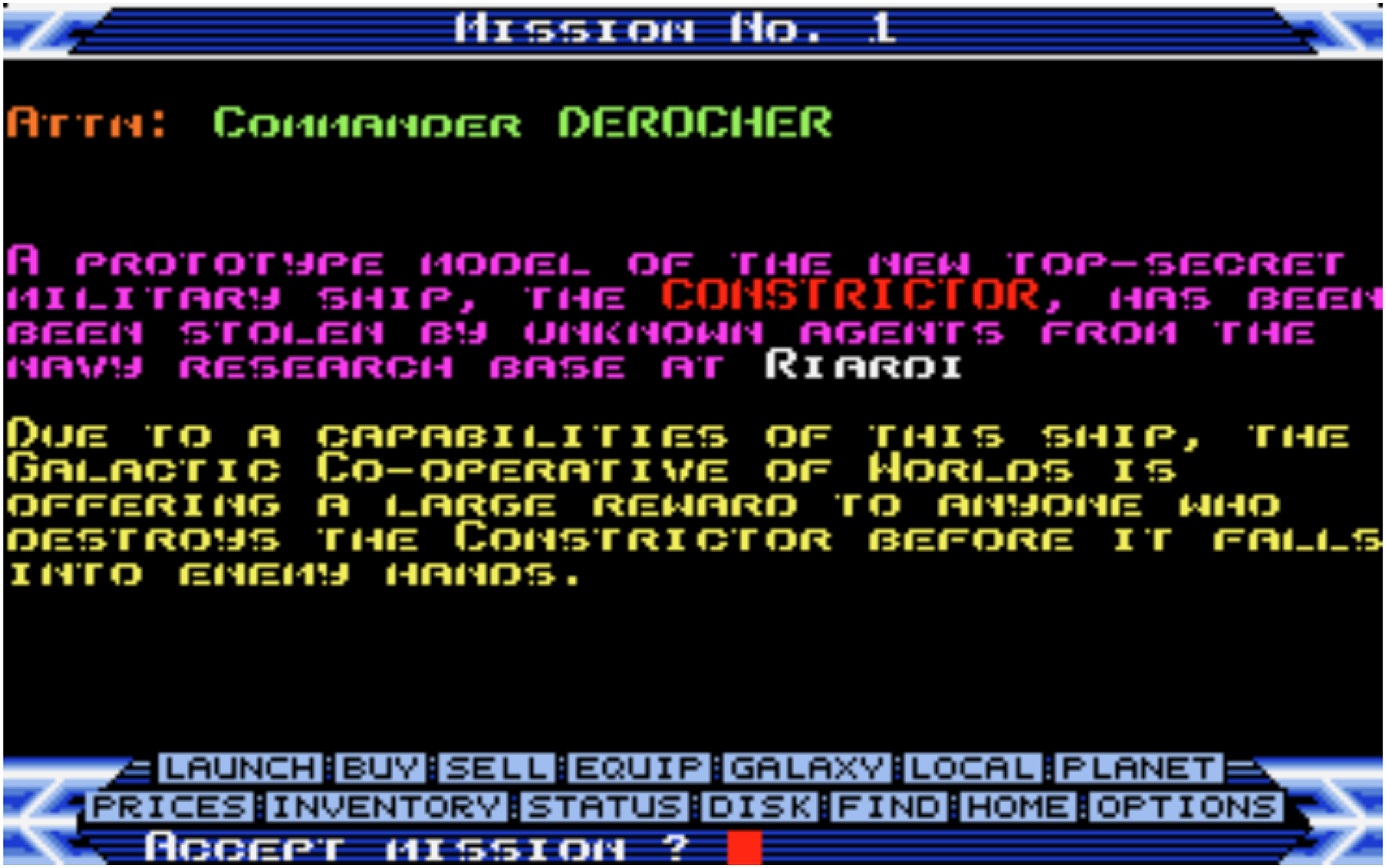
Elite


GALACTIC CHART #3
LAUNCH
BUY
SELL
EQUIP
GALAXY
LOCAL
PLANET
PRICES
INVENTORY
STATUS
DISK
FIND
HOME
OPTIONS

Elite


Buy Cargo
5.2
7.6
23.2
14.0
31.2
88.8
157.6
79.2
54.8
36.4
78.8
63.6
9.2
38.4
70.0
20.8
40.0
97.2
Cash: 29630.5 Cr, Hold: 35.0t spare
LAUNCH BUY SELL EQUIP GALAXY LOCAL PLANET
PRICES INVENTORY STATUS DISK FIND HOME OPTIONS
Slaves - None available

Elite

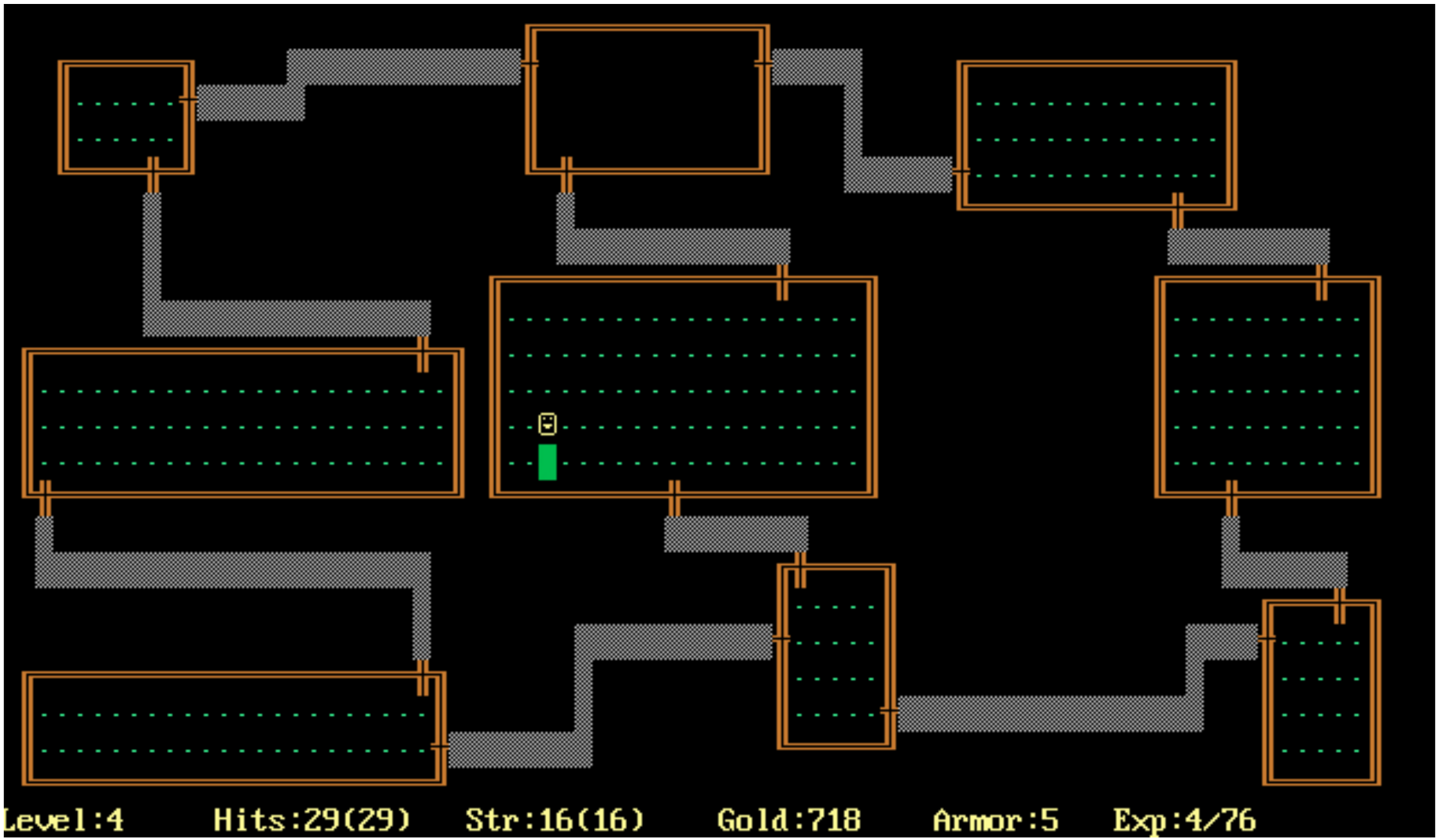

Mission No. 1
ATTN: COMMANDER DEROCHER
A PROTOTYPE MODEL OF THE NEW TOP-SECRET MILITARY SHIP, THE CONSTRICTOR, HAS BEEN BEEN STOLEN BY UNKNOWN AGENTS FROM THE NAVY RESEARCH BASE AT Riardi
DUE TO A CAPABILITIES OF THIS SHIP, THE GALACTIC CO-OPERATIVE OF WORLDS IS OFFERING A LARGE REWARD TO ANYONE WHO DESTROYS THE CONSTRICTOR BEFORE IT FALLS INTO ENEMY HANDS.
LAUNCH BUY SELL EQUIP GALAXY LOCAL PLANET
PRICES INVENTORY STATUS DISK FIND HOME OPTIONS
Accept Mission ?

Elite

Fits in memory on a Commodore 64!

No Man’s Sky

Rogue



Diablo III

The Binding of Isaac

Hades

Dwarf Fortress

Spelunky

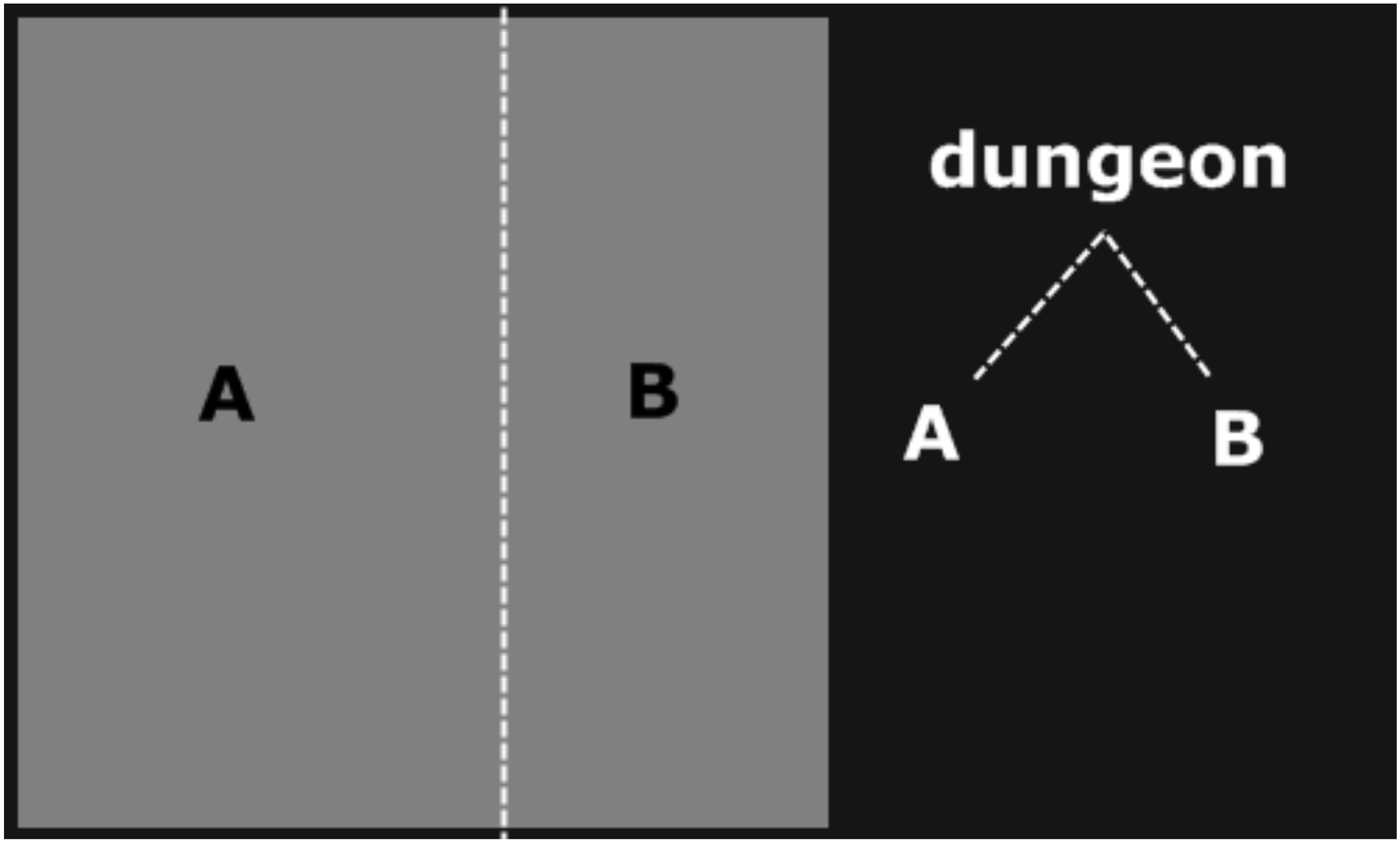
Civilization IV

How do you do it?

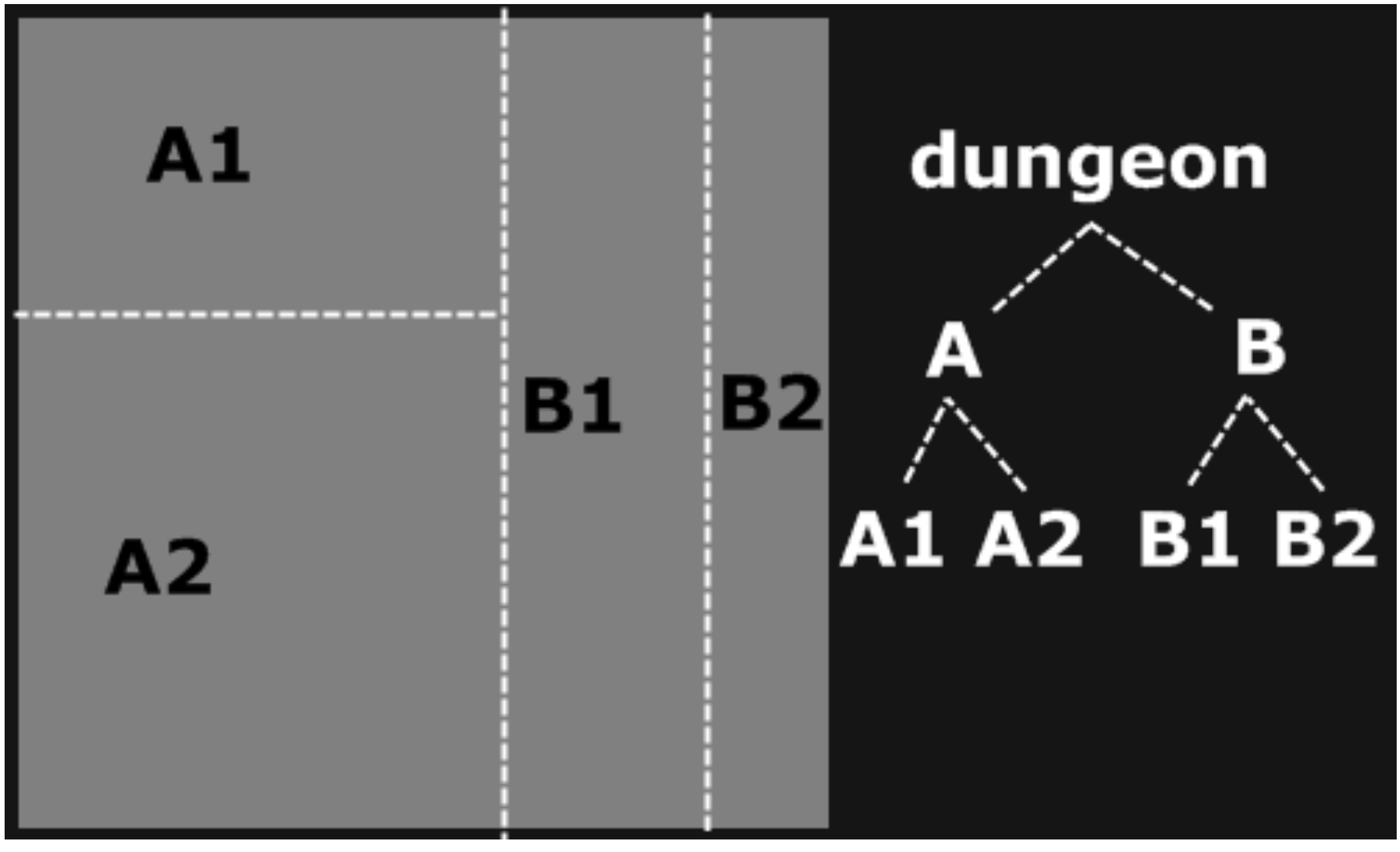
Binary space partitioning

- choose a random direction : horizontal or vertical splitting
- choose a random position (x for vertical, y for horizontal)
- split the dungeon into two sub-dungeons
- call the same procedure for each sub-dungeon until finished
- Finally, create a room in each leaf and connect siblings

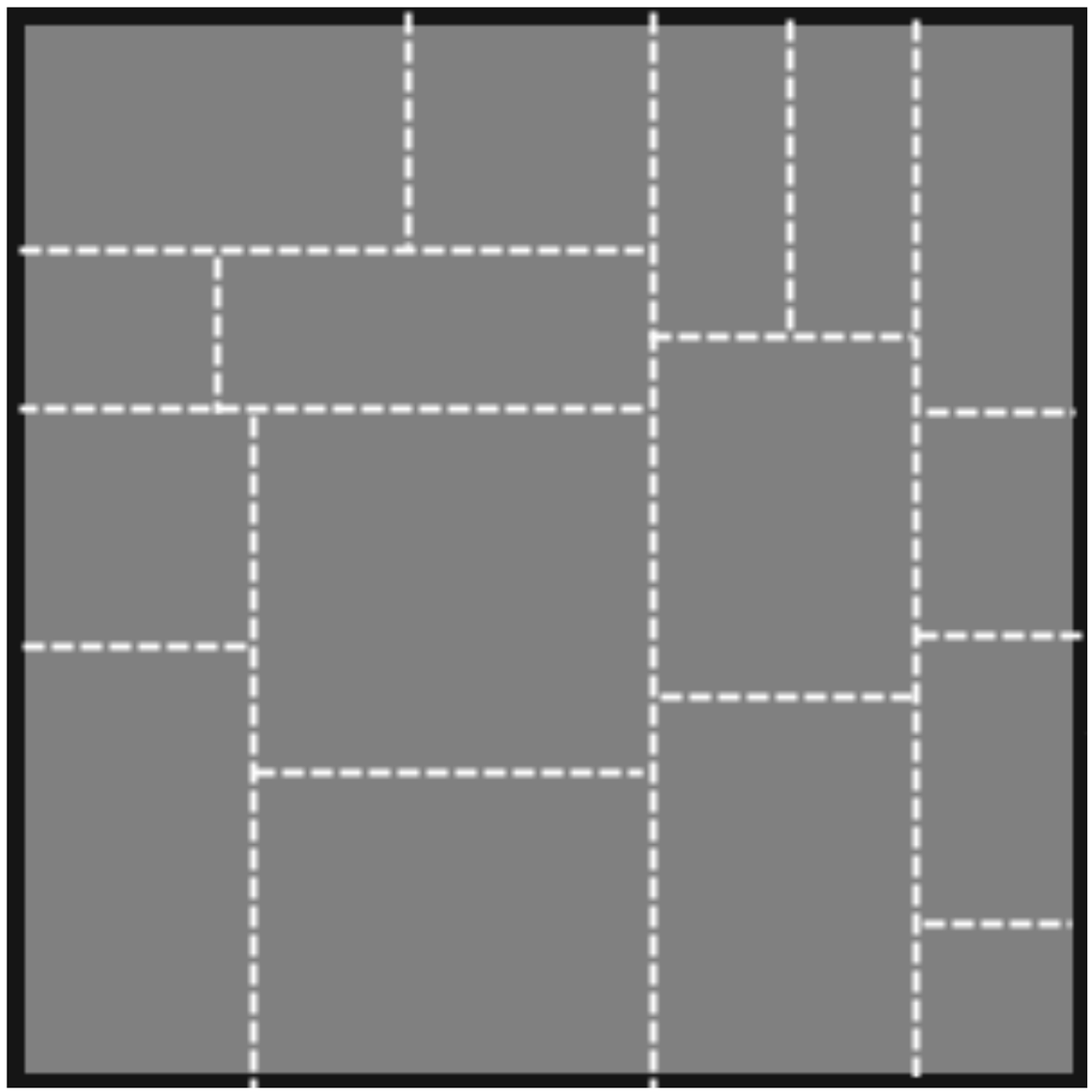
Binary space partitioning



Binary space partitioning

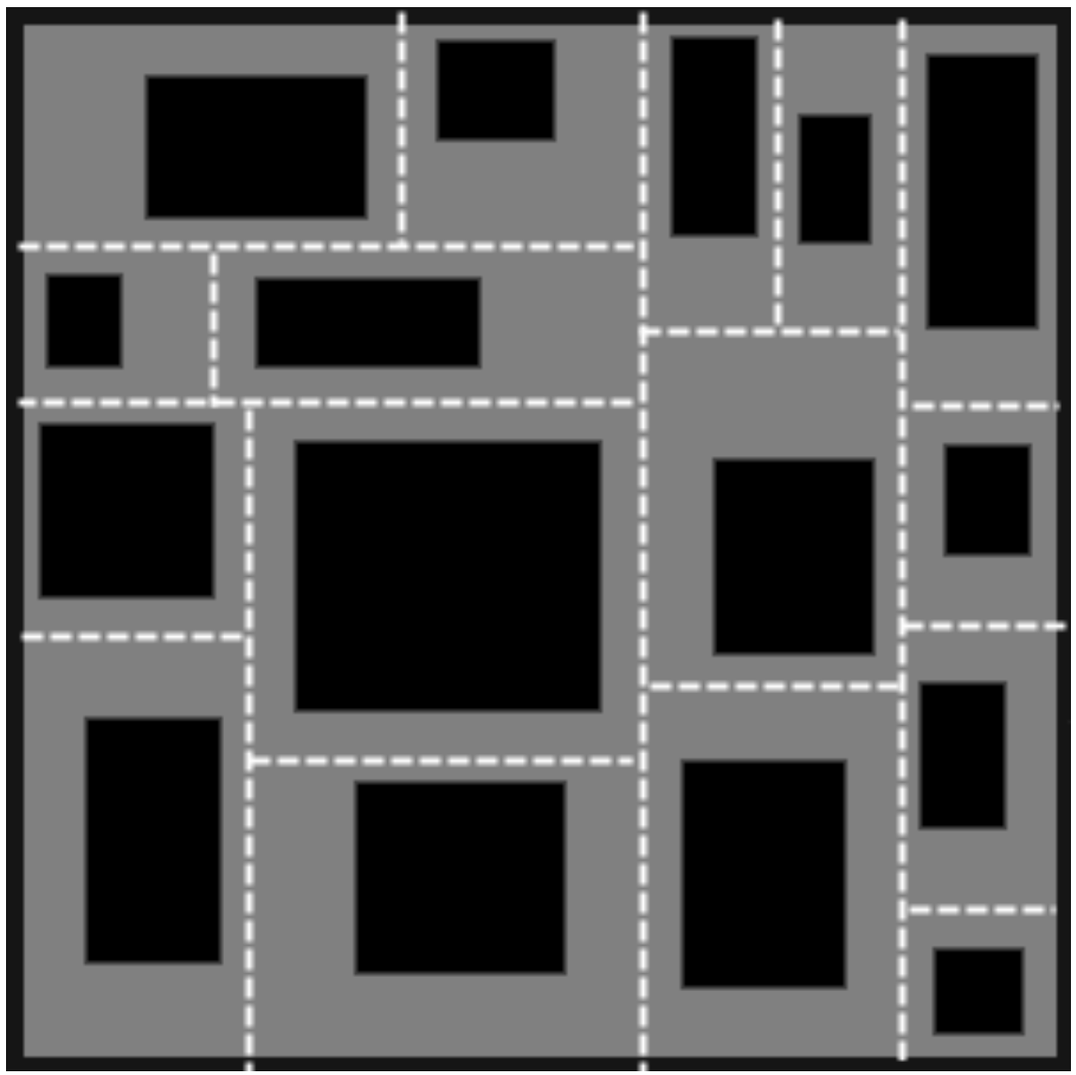


Binary space partitioning



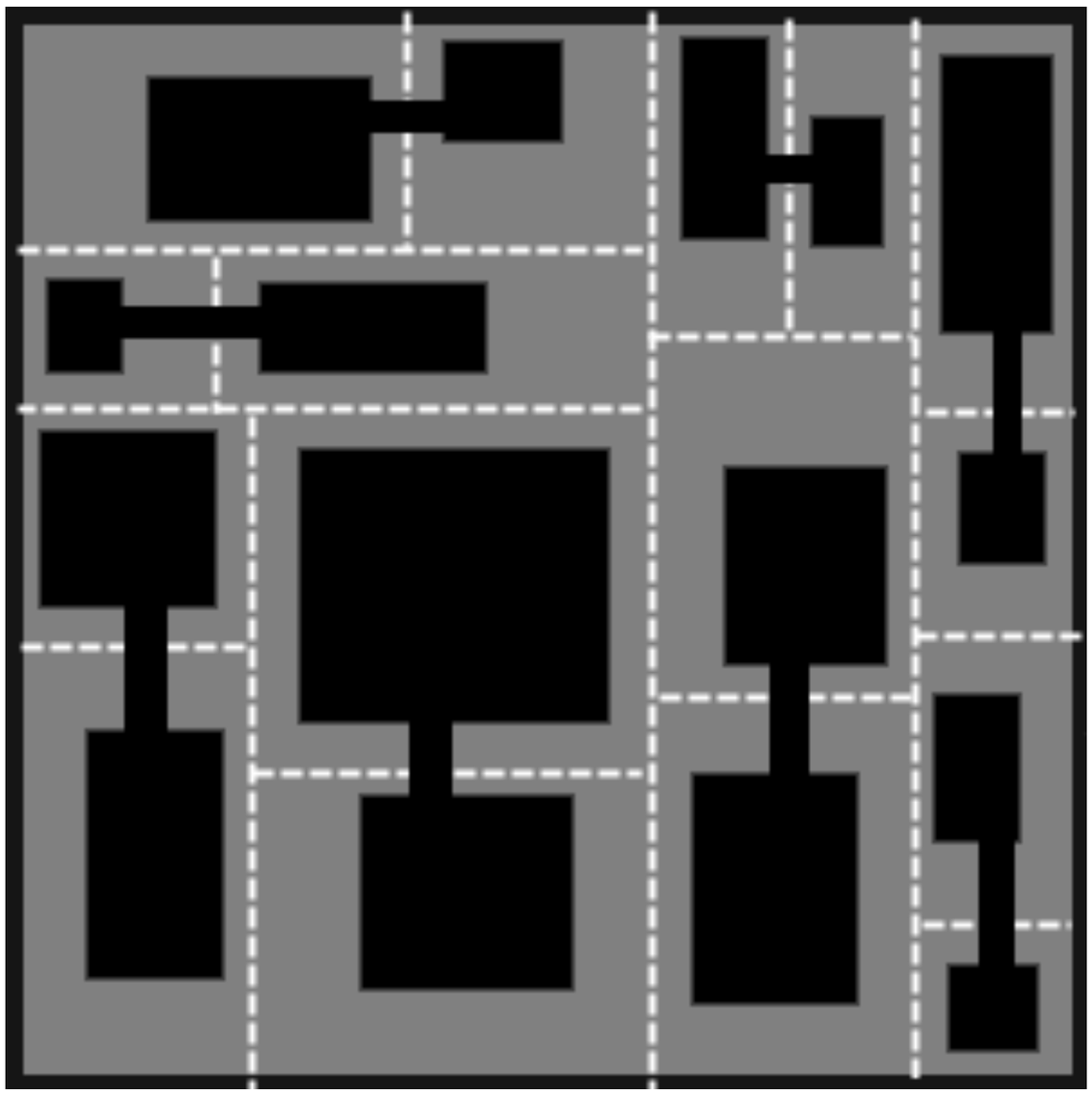
After 4 splitting

Binary space partitioning



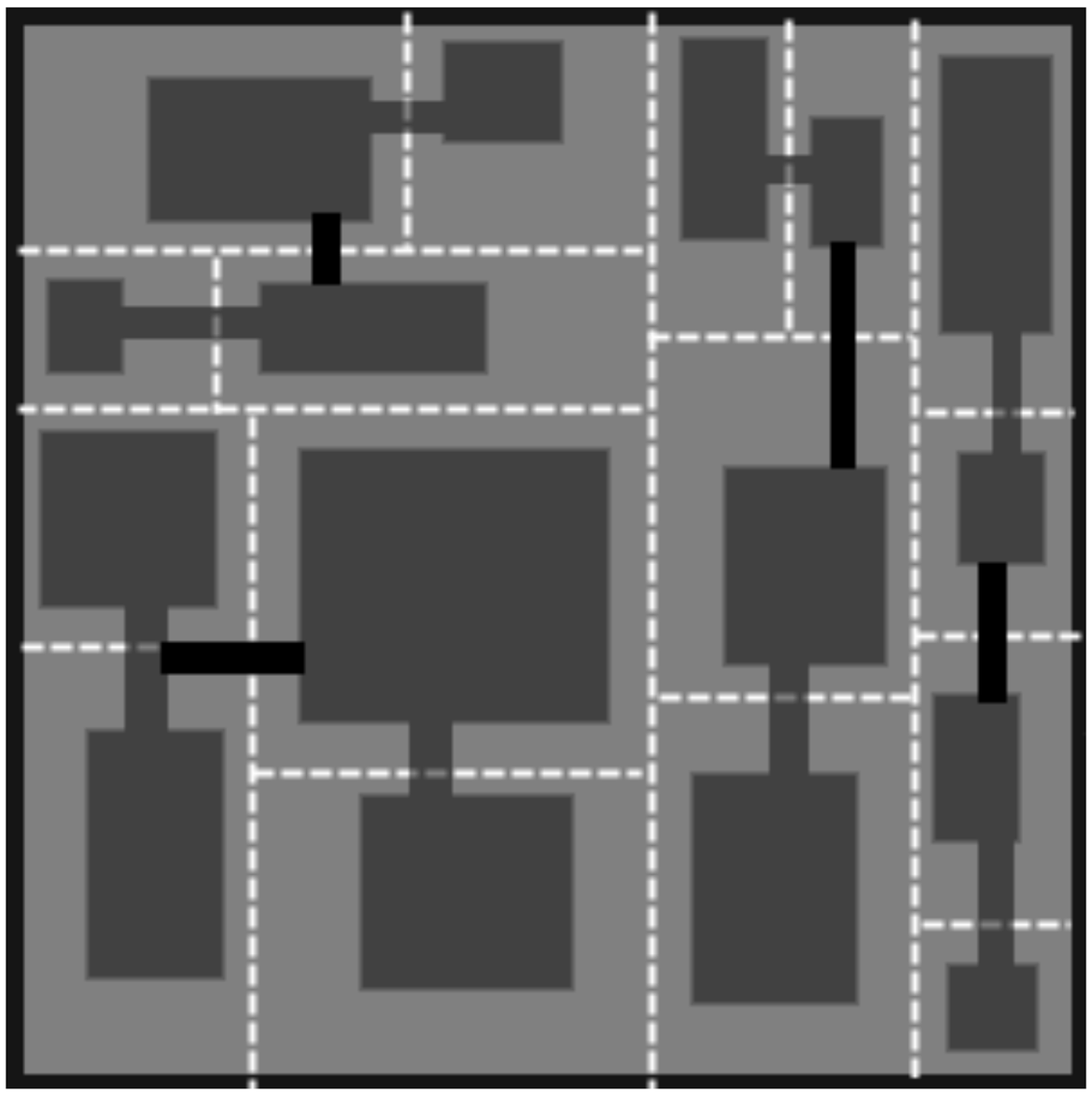
Creating the rooms

Binary space partitioning



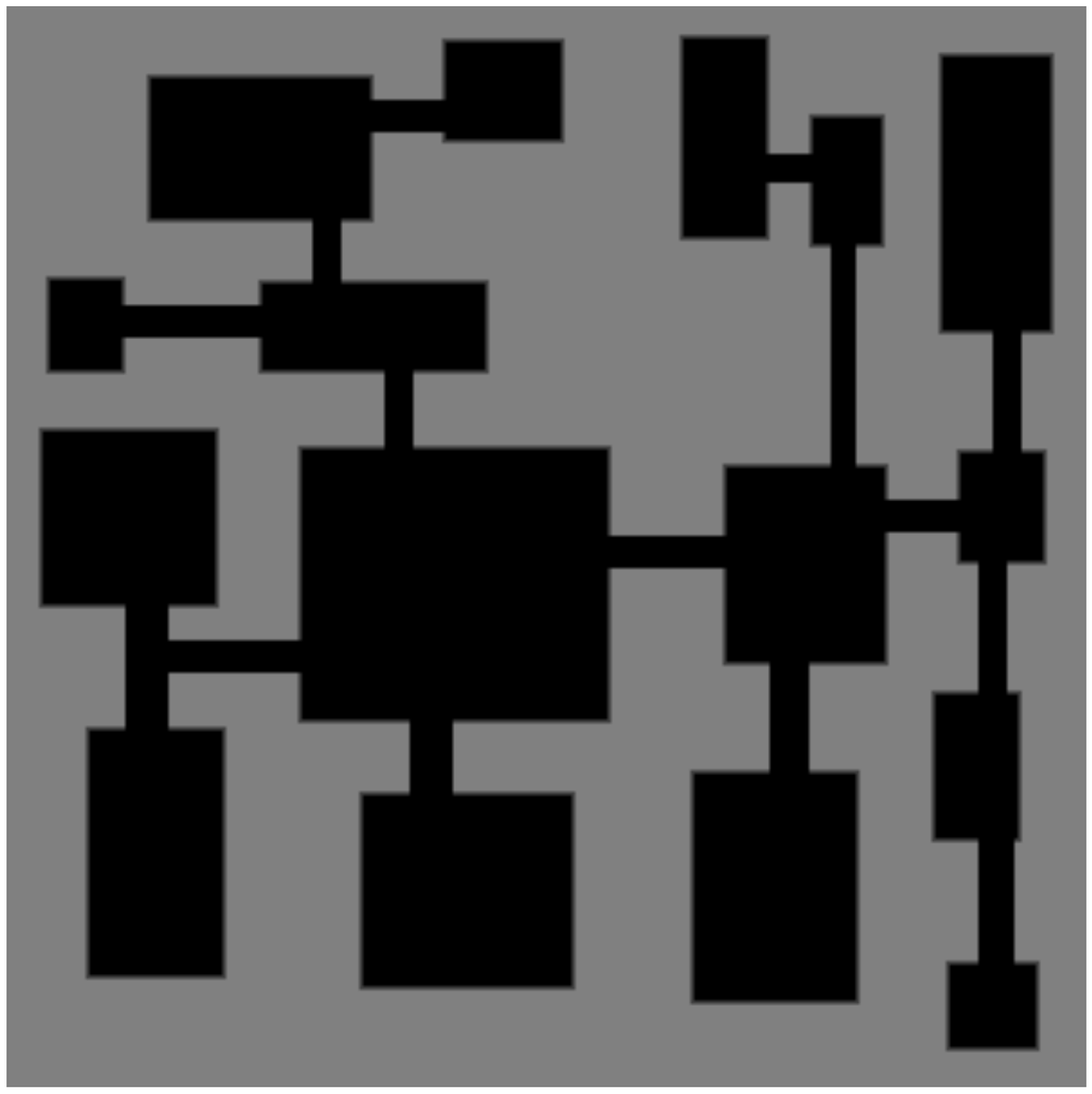
Adding level-1 corridors

Binary space partitioning



Level-2 corridors

Binary space partitioning



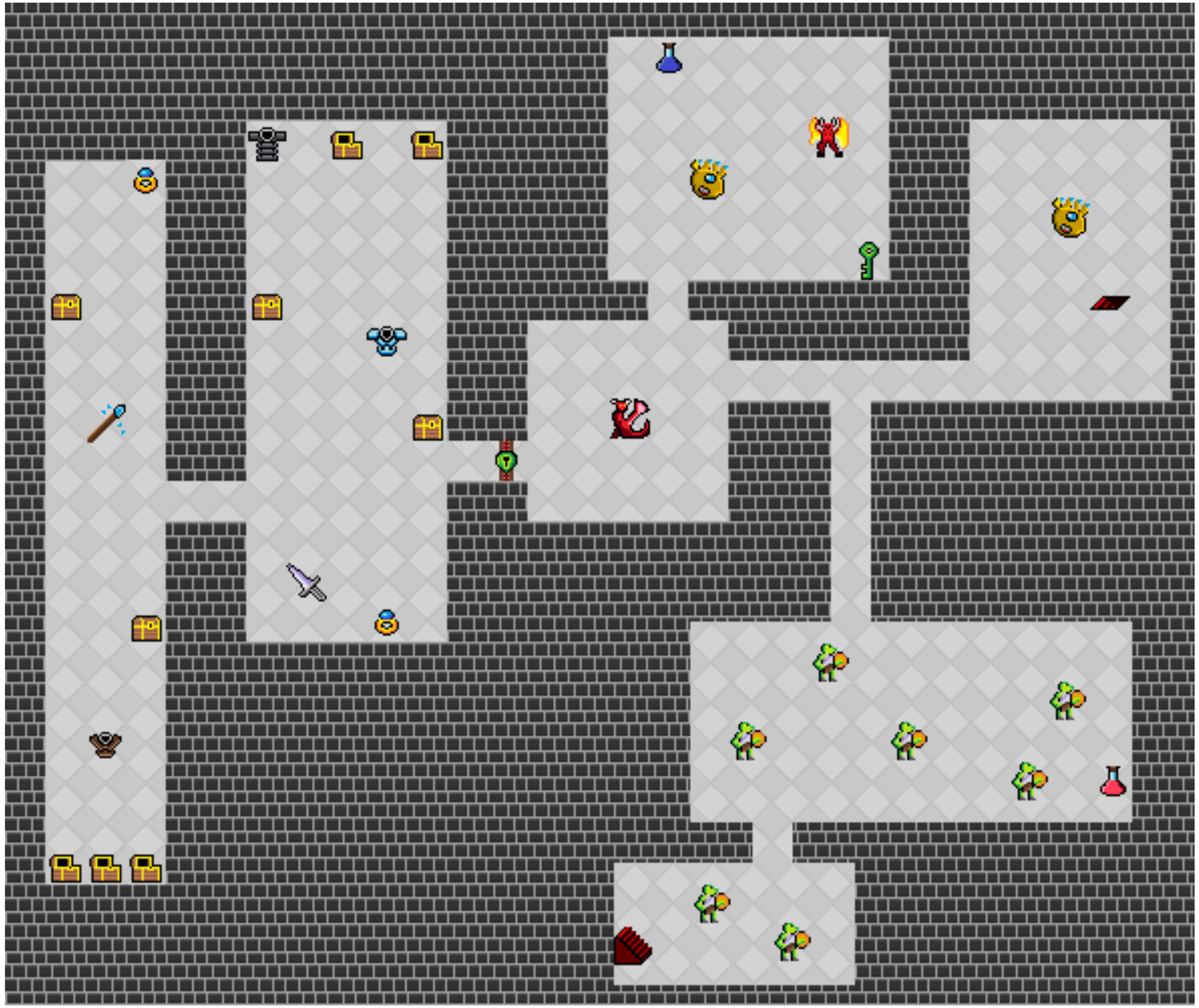
The complete dungeon

Procedure

```
1: start with the entire dungeon area (root node of the BSP tree)
2: divide the area along a horizontal or vertical line
3: select one of the two new partitions
4: if this partition is above than the minimal acceptable size:
5:   go to step 2 (using this partition as the area to be divided)
6: select the other partition, and go to step 4
7: for every partition:
8:   create a room within the partition by randomly
     choosing two points (top left and bottom right)
     within the partition's boundaries
9: starting from the lowest layers, draw corridors to connect
   rooms in the nodes of the BSP tree with children of the same
   parent
10:repeat 9 until the children of the root node are connected
```

Bros vs cons

- Bros:
  - easy to implement
  - no overlapping rooms or corridors
  - easy to create group of rooms
- Cons
  - very neat

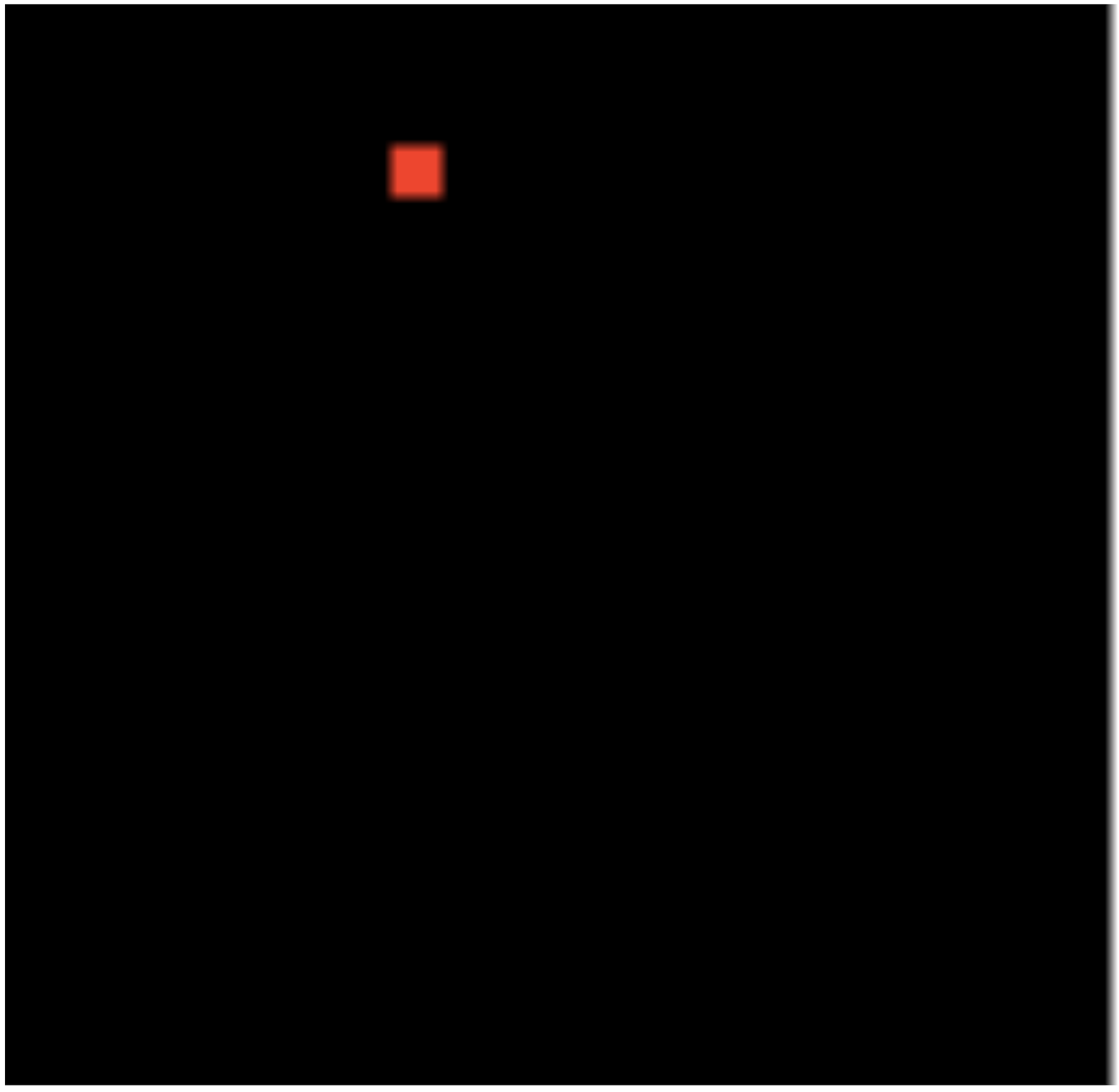


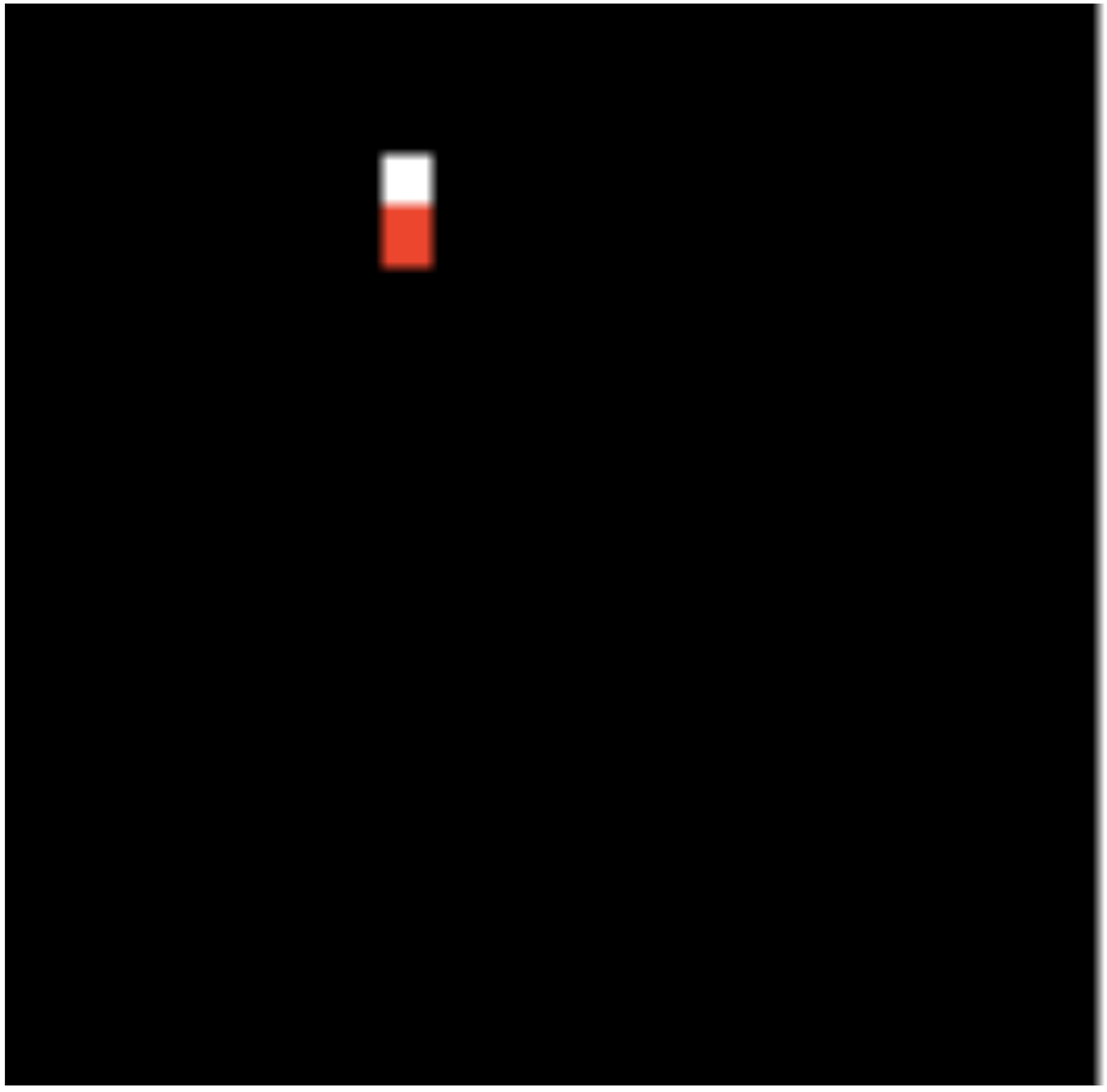
Agent-based methods

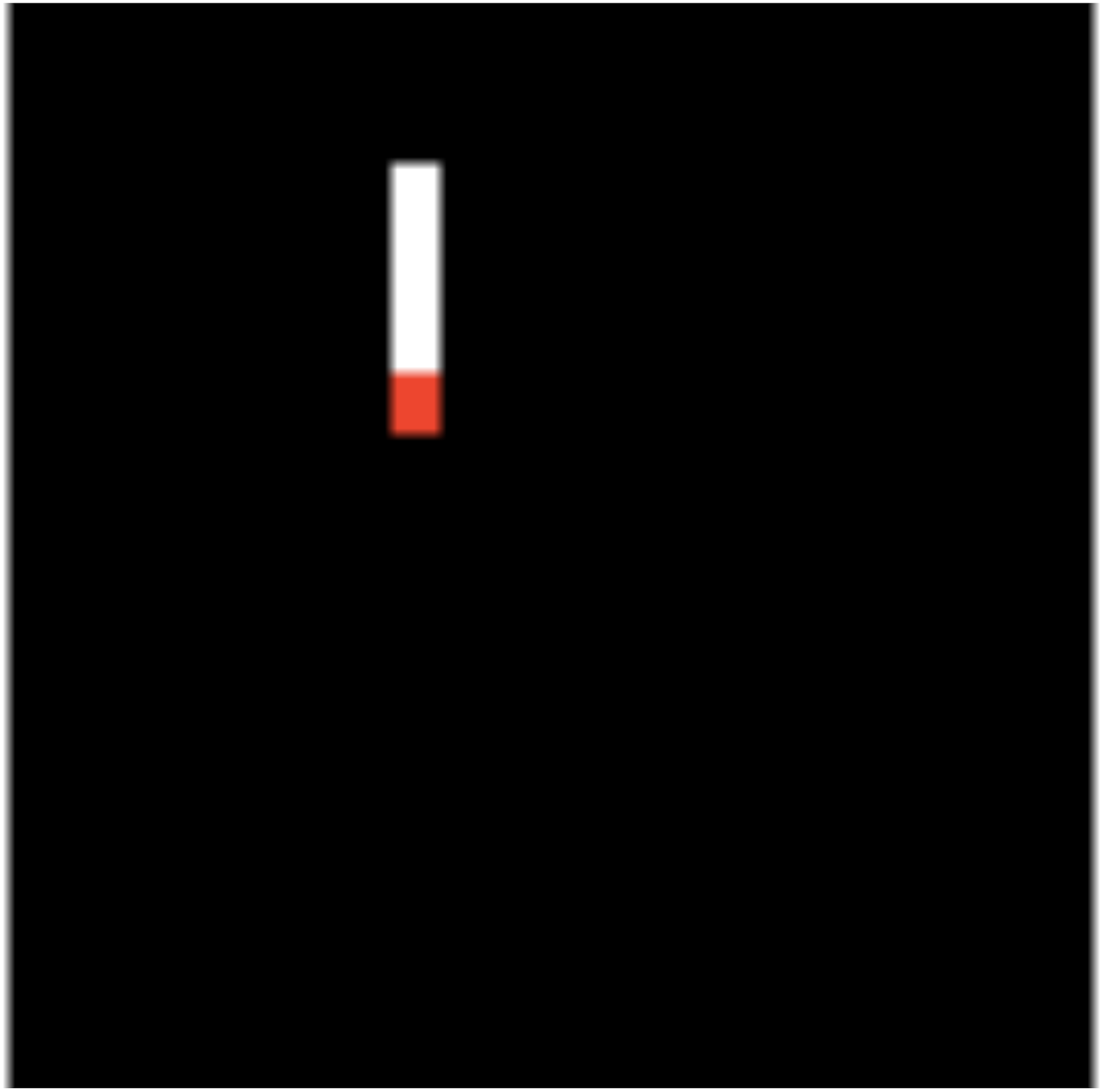
- Less predictable
- Less organized

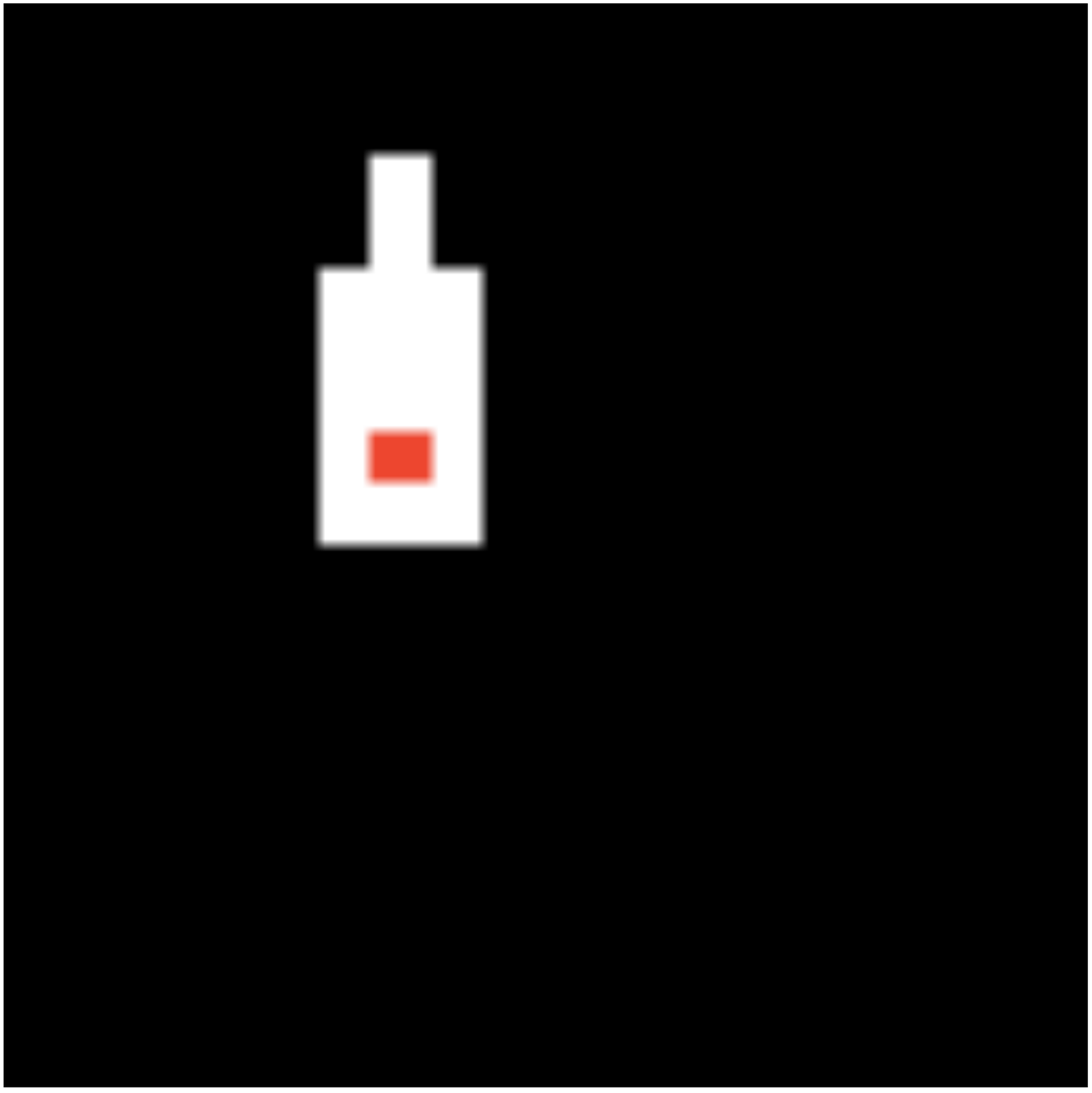
A highly stochastic method

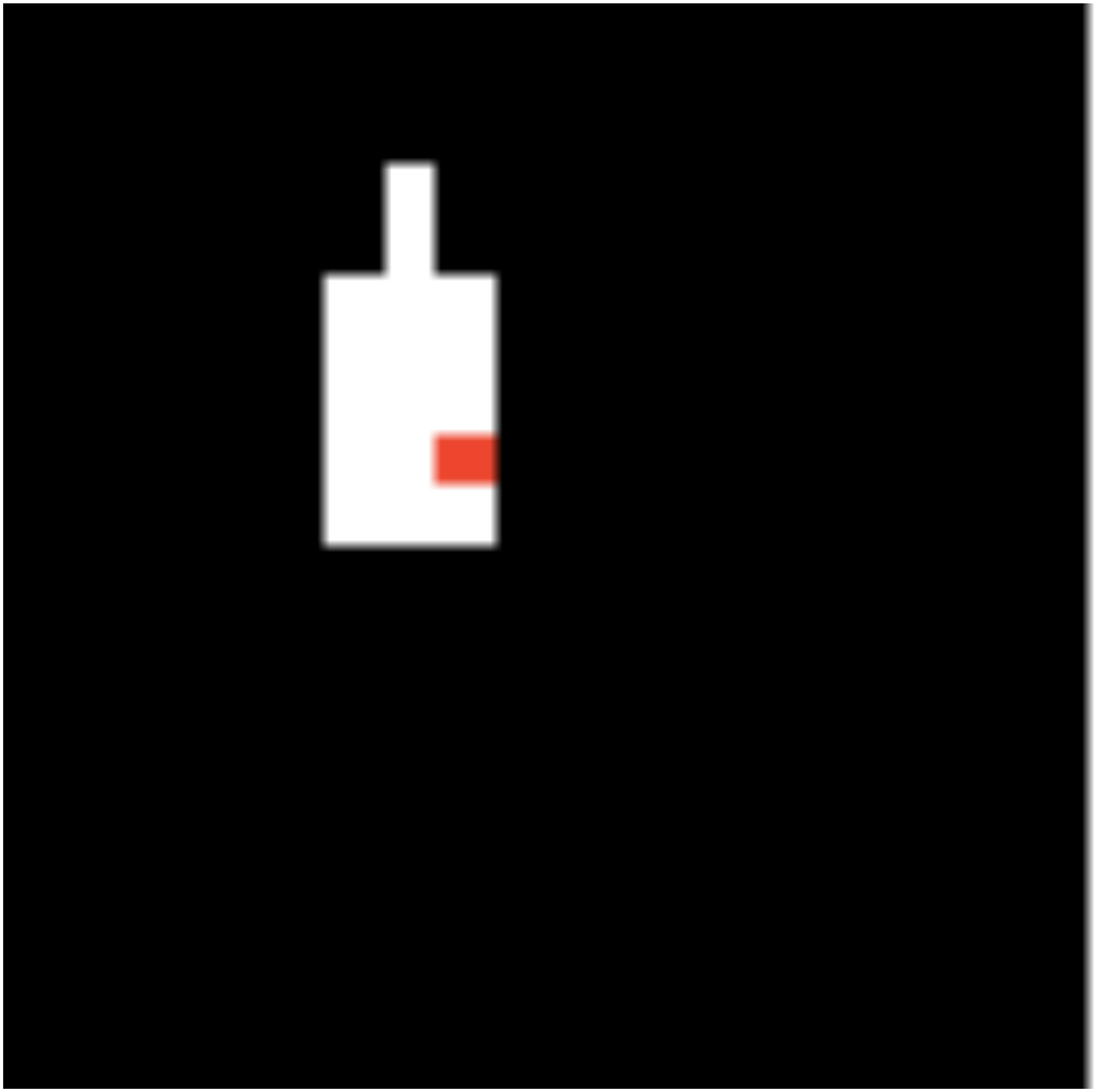
```
1: initialize chance of changing direction Pc=5
2: initialize chance of adding room Pr=5
3: place the digger at a dungeon tile and randomize its direction
4: dig along that direction
5: roll a random number Nc between 0 and 100
6: if Nc below Pc:
7:   randomize the agent's direction
8:   set Pc=0
9: else:
10:  set Pc=Pc+5
11:roll a random number Nr between 0 and 100
12:if Nr below Pr:
13:  randomize room width and room height between 3 and 7
14:  place room around current agent position
14:  set Pr=0
15:else:
16:  set Pr=Pr+5
17:if the dungeon is not large enough:
18:  go to step 4
```











Less stochastic method

- Use look ahead to avoid overlaps

- Make few changes in the direction

Procedure

```
1: place the digger at a dungeon tile
2: set helper variables Fr=0 and Fc=0
3: for all possible room sizes:
3:   if a potential room will not intersect existing rooms:
4:     place the room
5:     Fr=1
6:     break from for loop
7: for all possible corridors of any direction and length 3 to 7:
8:   if a potential corridor will not intersect existing rooms:
9:     place the corridor
10:    Fc=1
11:    break from for loop
12:if Fr=1 or Fc=1:
13:  go to 2
```

Less stochastic method

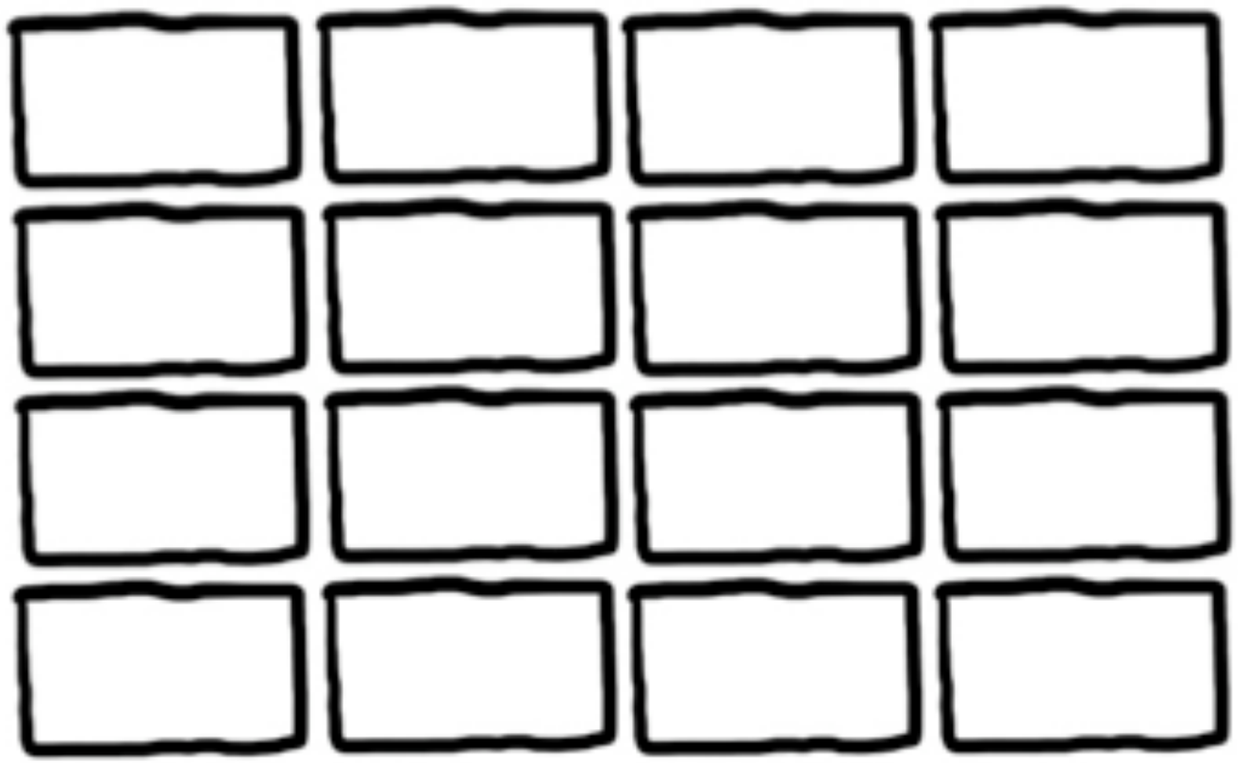
Less stochastic method

Spelunky

- Roguelike-like platformer

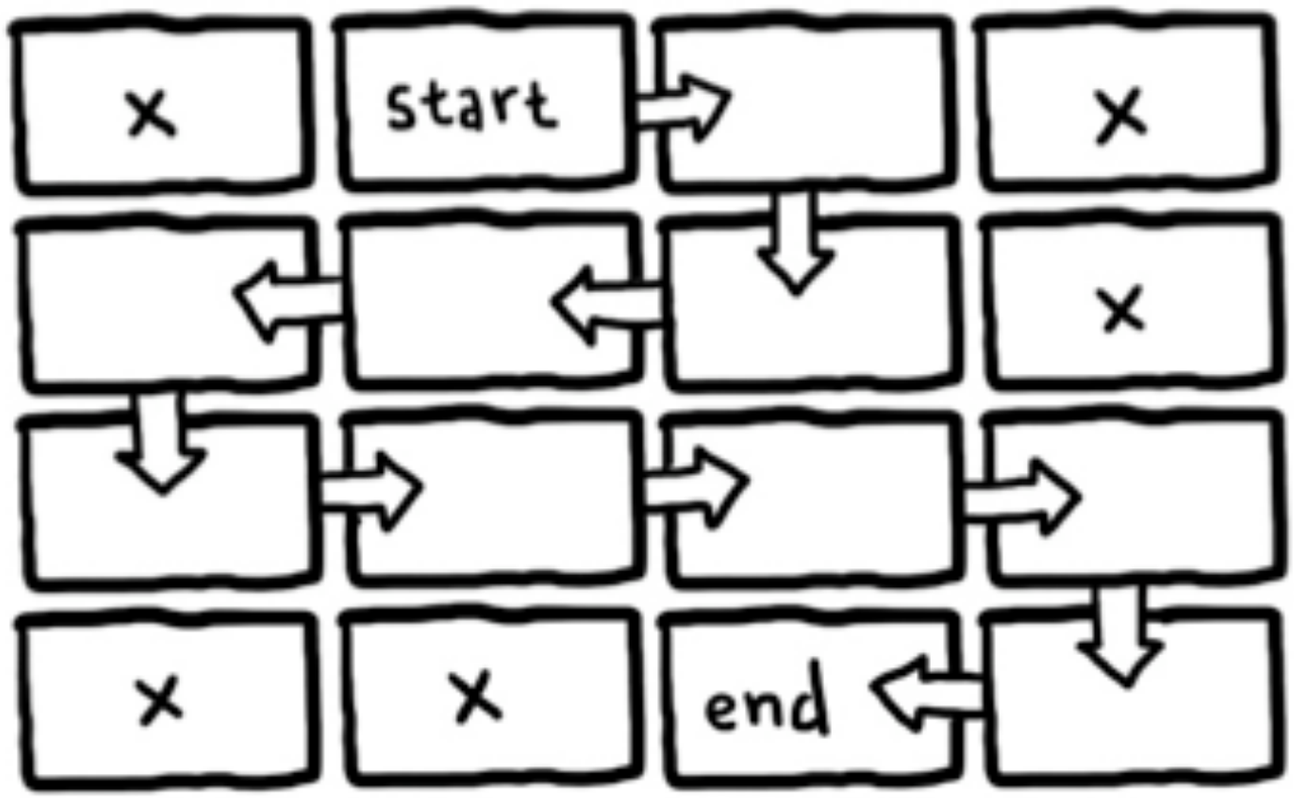
- Combines the fast pace of the platformer with the replayability and unpredictability of the roguelike

Spelunky level generation



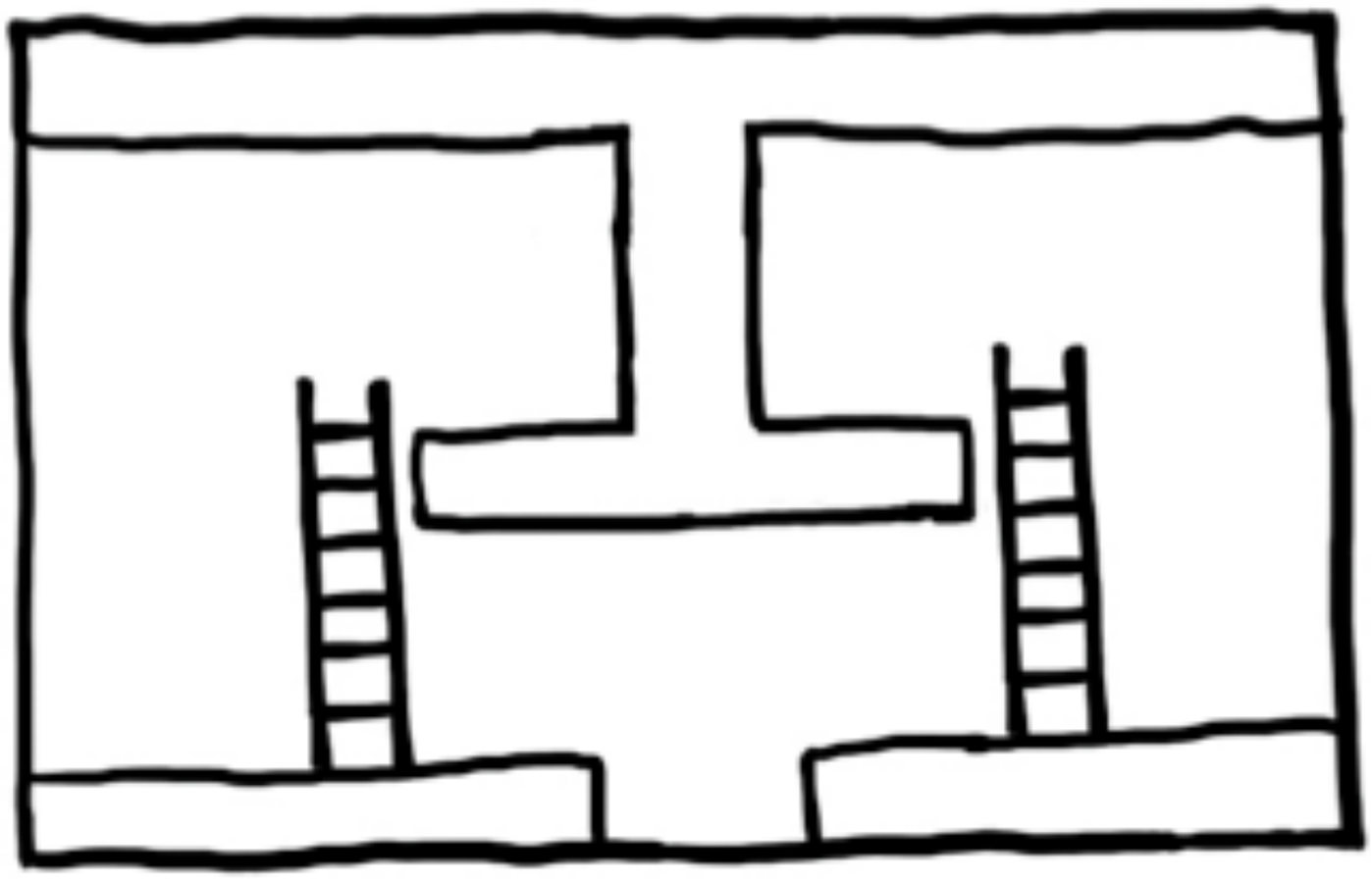
Each level is divided into a grid of 16 rooms.

Spelunky level generation



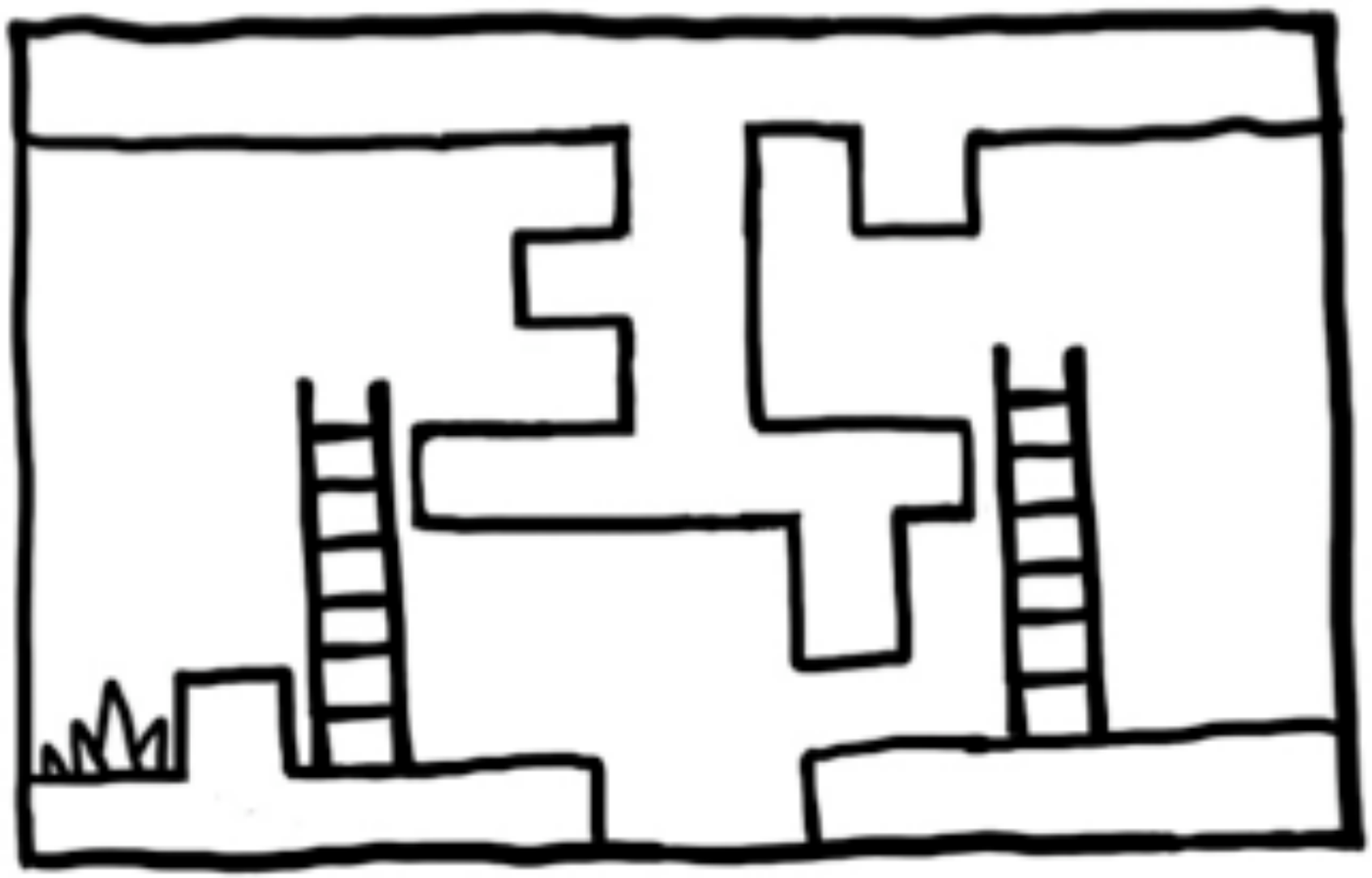
A path is drawn from entrance at the top to exit at the bottom.

Spelunky level generation



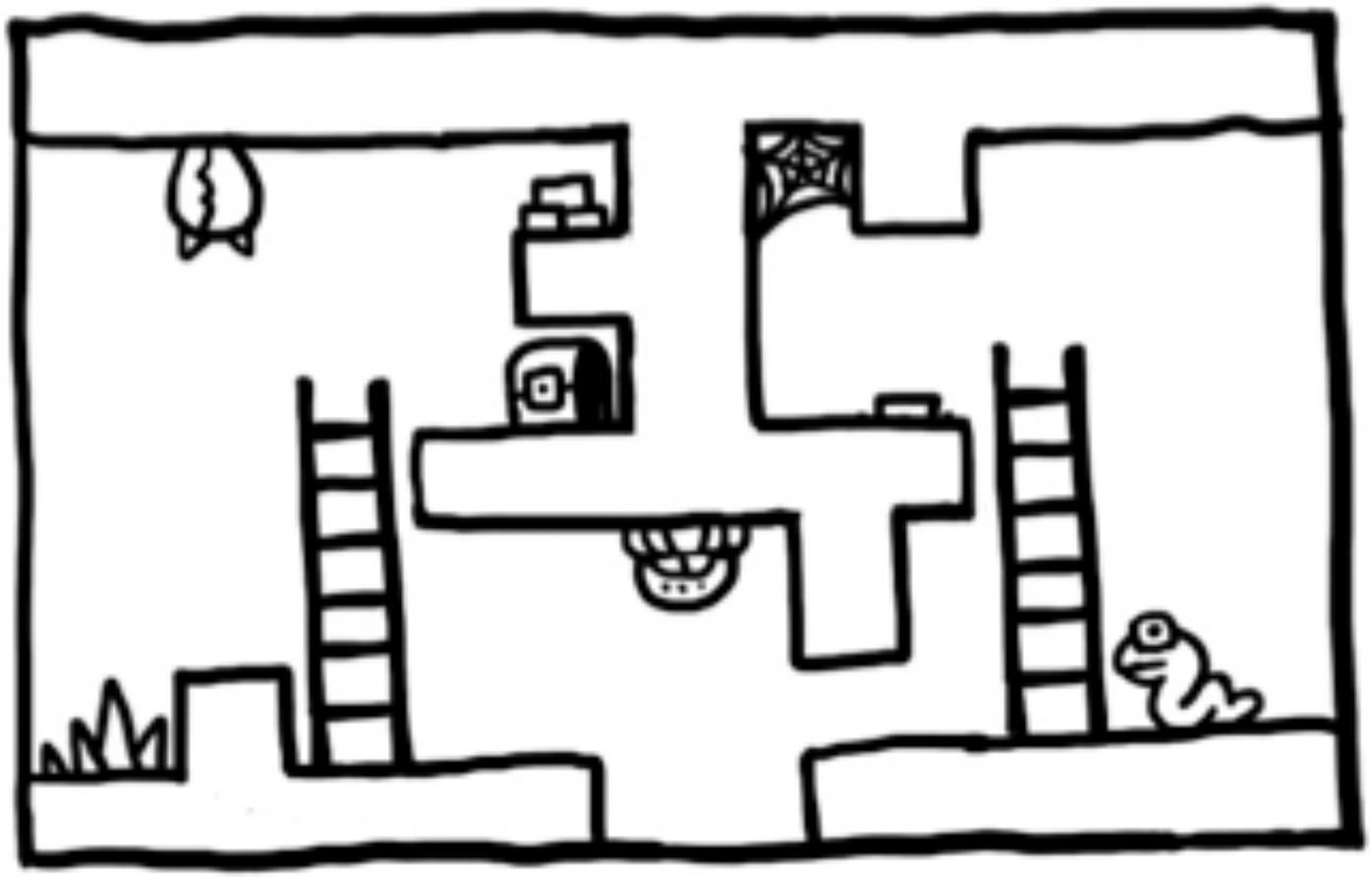
Each room is selected from a set of templates,
so as to fit in the path drawn in the previous step
(and also with the position in the level).

Spelunky level generation



Randomized chunks in each room add variation.

Spelunky level generation



Finally, critters, traps, treasures etc are added.

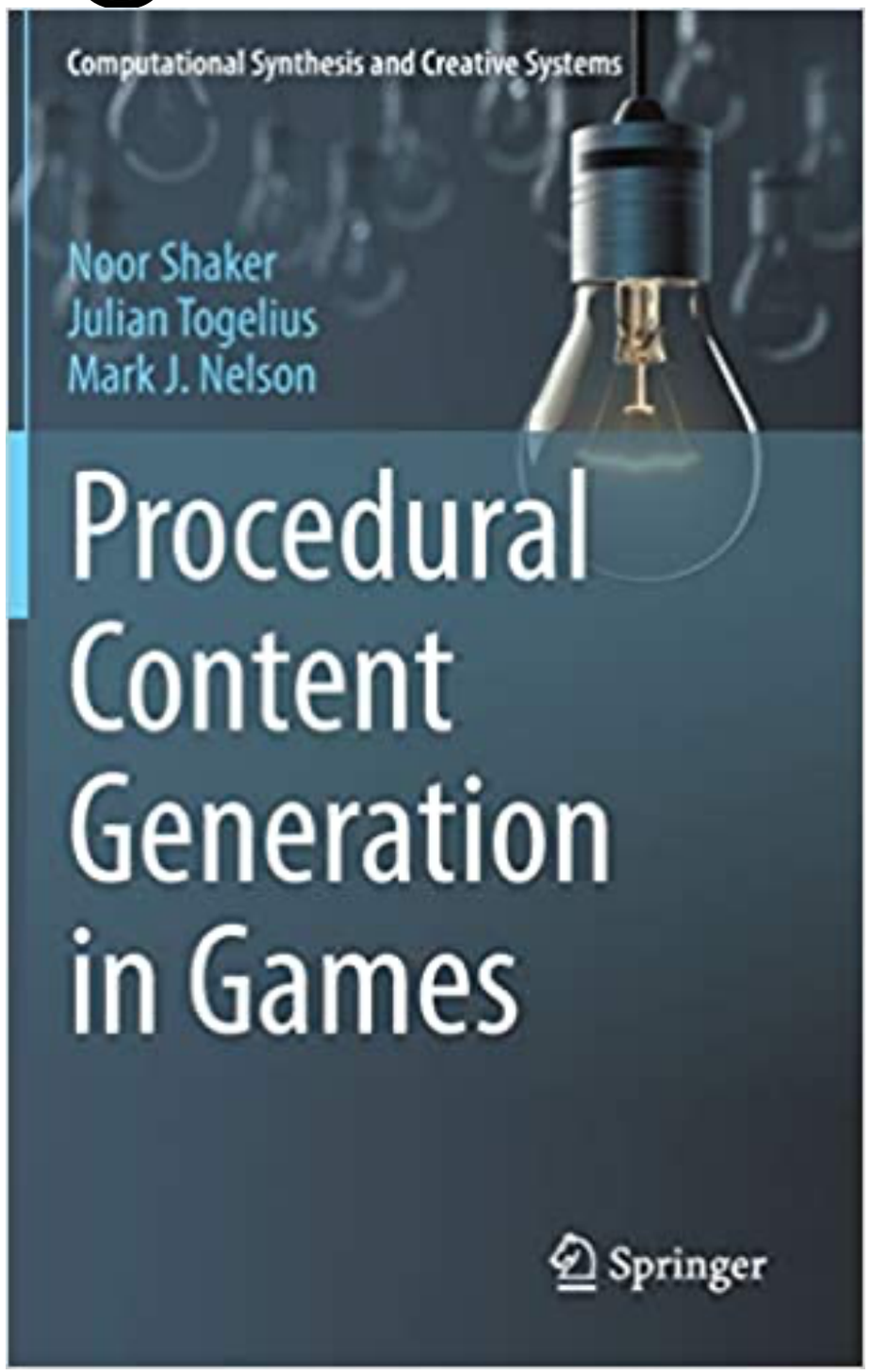
Spelunky level generation

```
00000000011
00600000L11
00000000L11
00000000L11
00000000011
00000000011
11111111111
```

Many other methods available

- Search-based PCG (evolution)
- Supervised machine learning
- Reinforcement learning
- Grammars
- Cellular automata
- Constraint satisfaction
- Noise, fractals

pcgbook.com


Computational Synthesis and Creative Systems
Noor Shaker
Julian Togelius
Mark J. Nelson
Procedural Content Generation in Games
Springer

Other resources

- Rot.js (ROguelike Toolkit for JavaScript)
- Kate Compton's writings (including Tracery, PCG for Everyone) http://www.galaxykate.com

Your mission

- Build a game prototype that incorporates PCG
  - Generate the levels… or something else!
- Explain how your PCG works
- Explain how PCG adds to your game