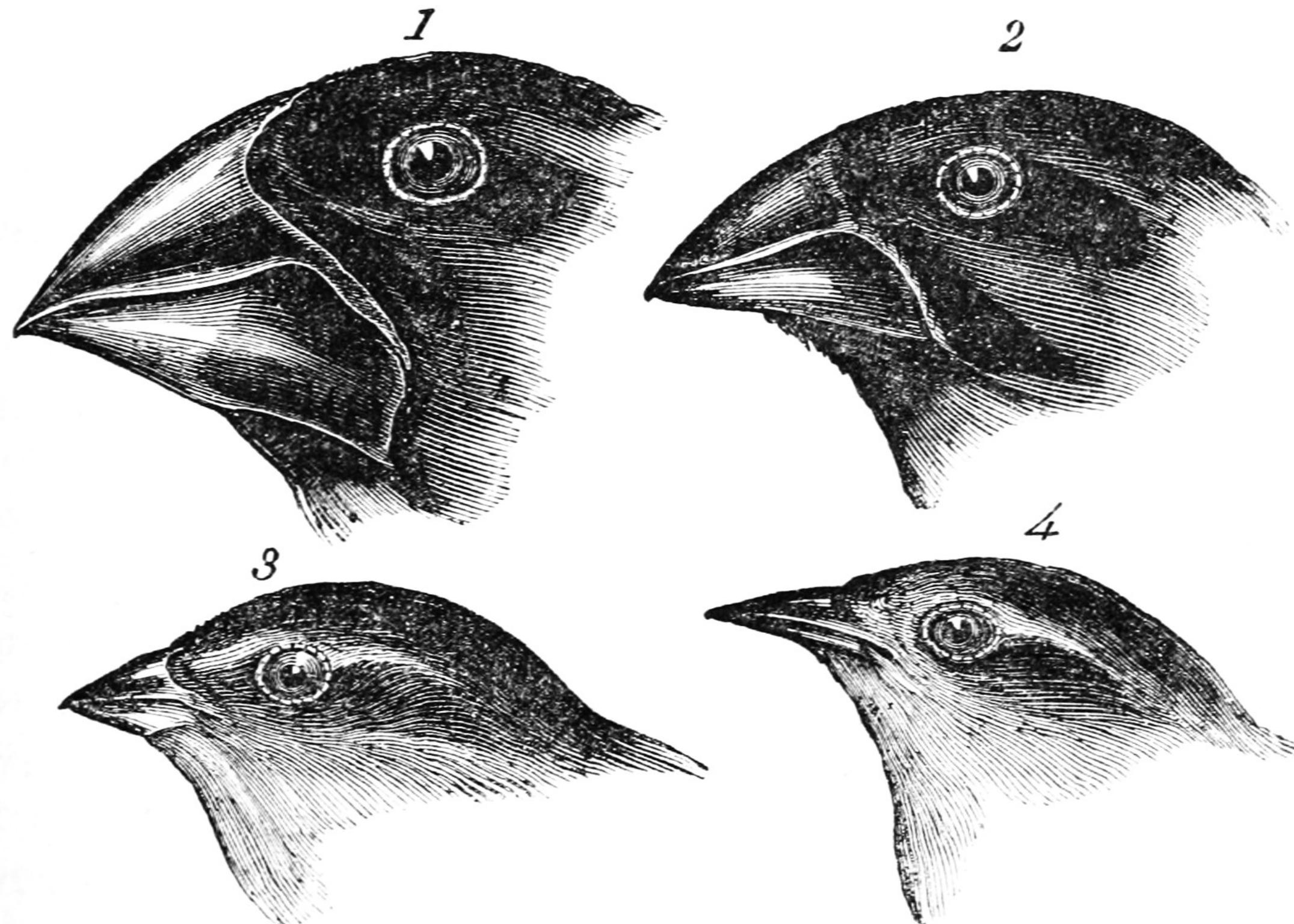
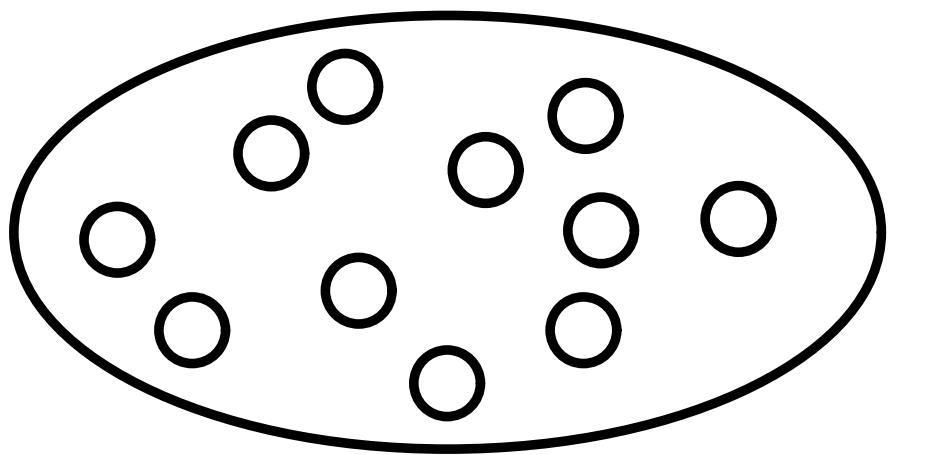


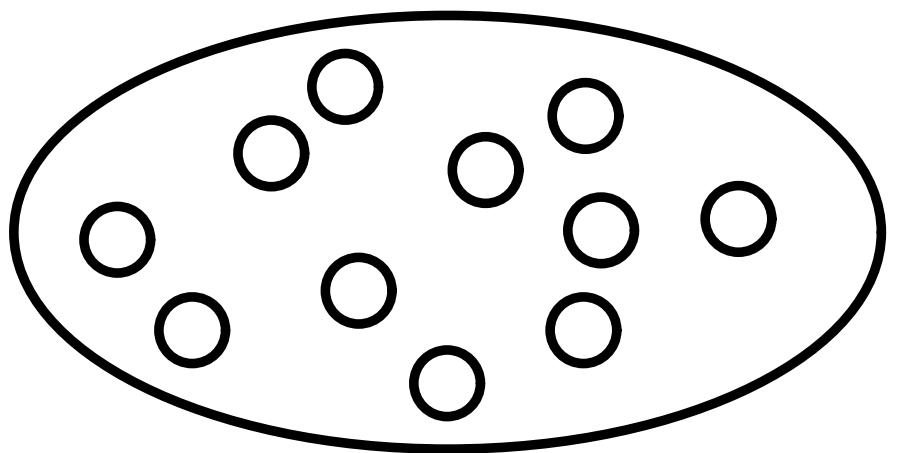
Genetic programming



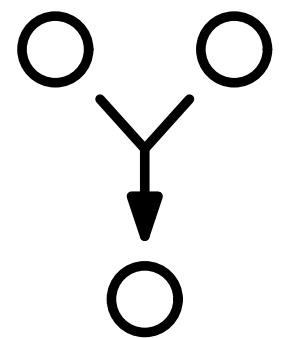
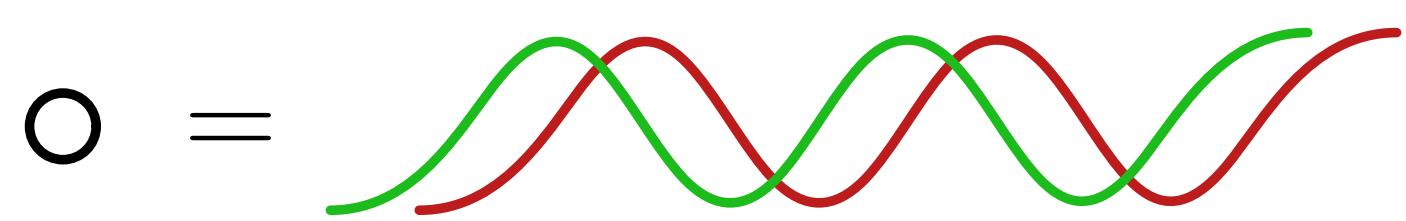
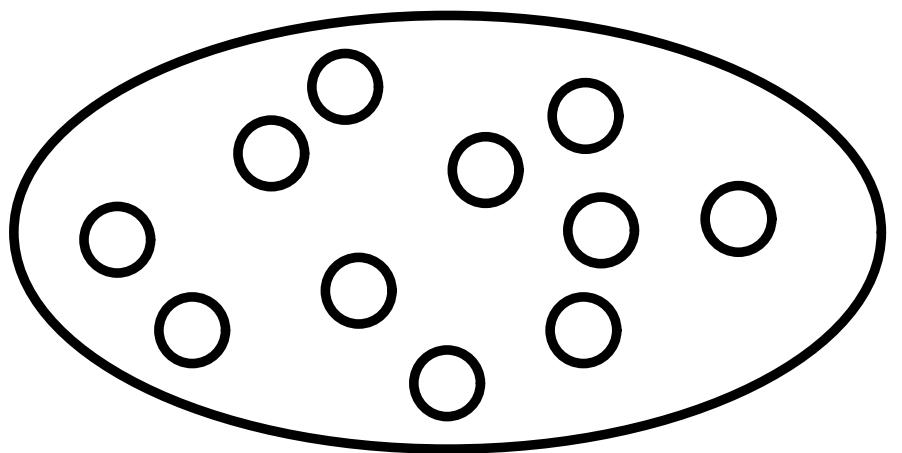
1. *Geospiza magnirostris.*
3. *Geospiza parvula.*

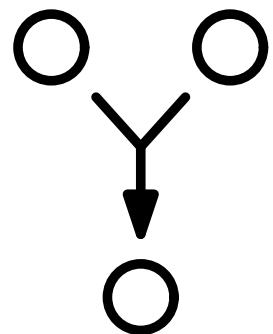
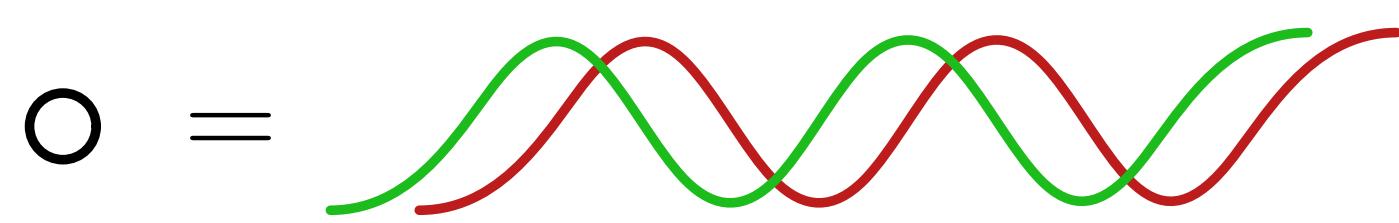
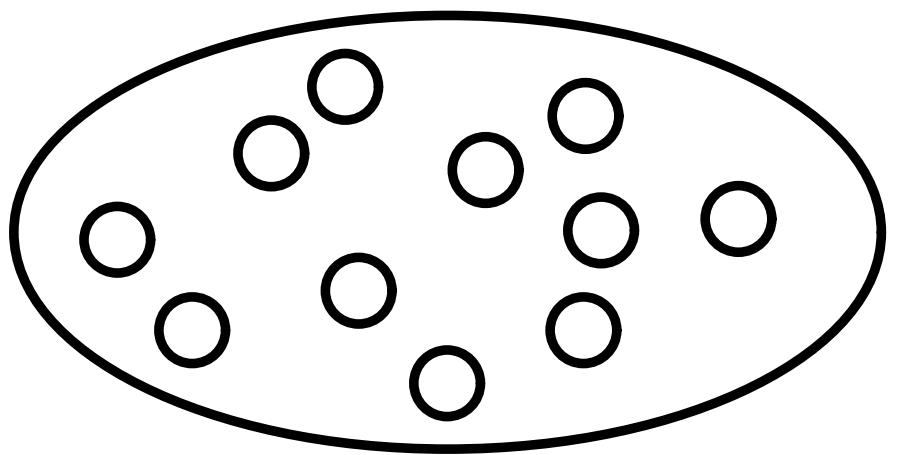
2. *Geospiza fortis.*
4. *Certhidea olivacea.*

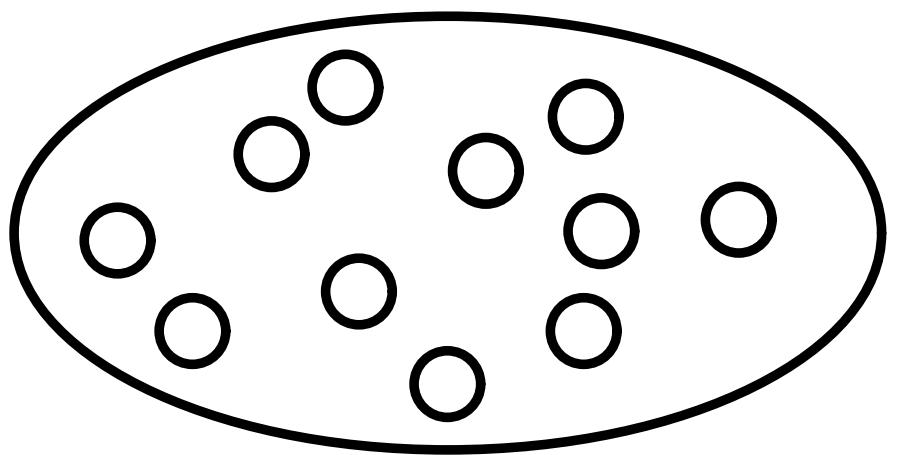


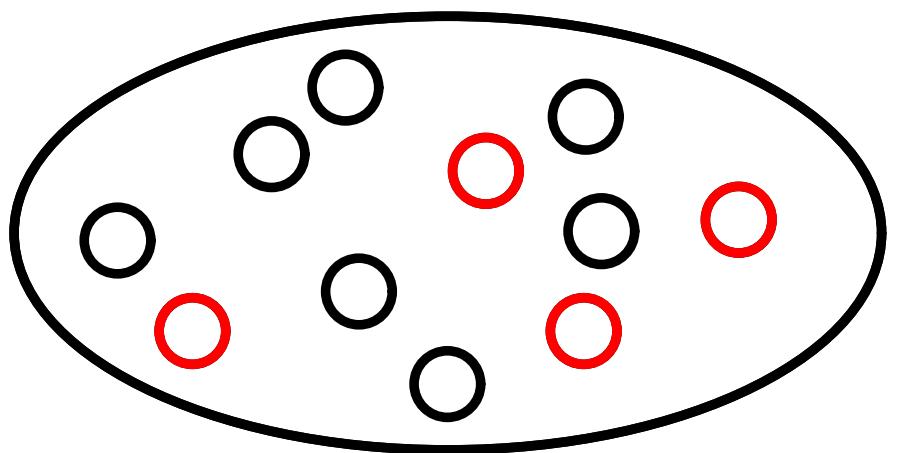


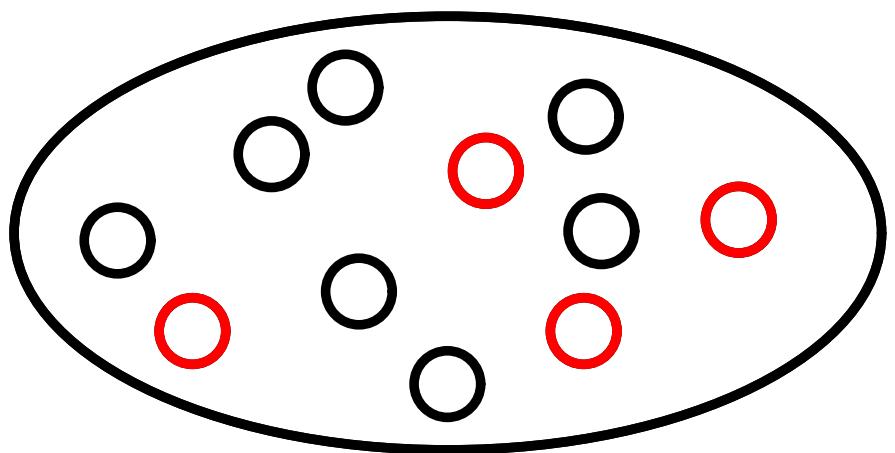
$\circ =$ A mathematical expression where a black circle symbol is followed by an equals sign and a plot of two overlapping bell-shaped curves. The curves are red and green, showing a double-peaked distribution. They are positioned at approximately [79, 420, 490, 535].

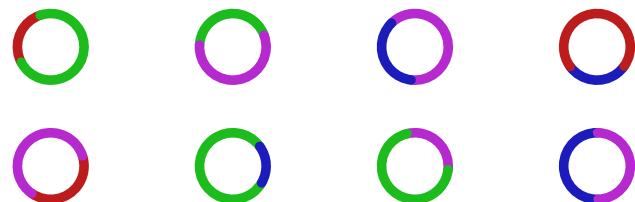
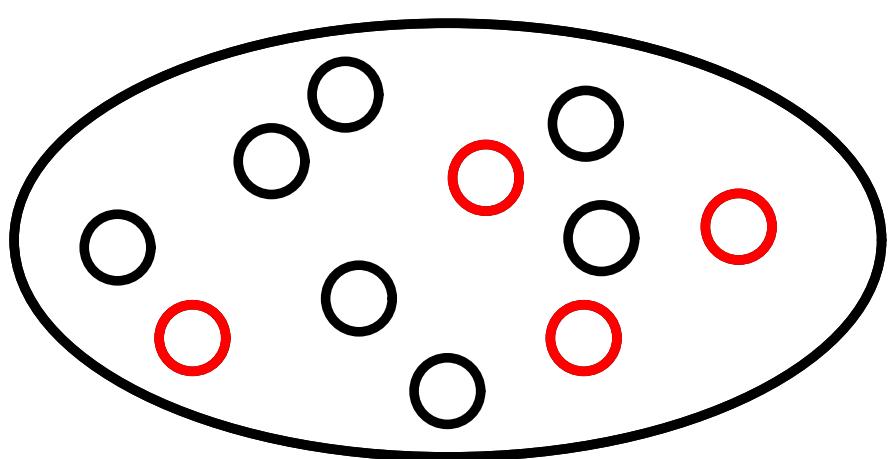












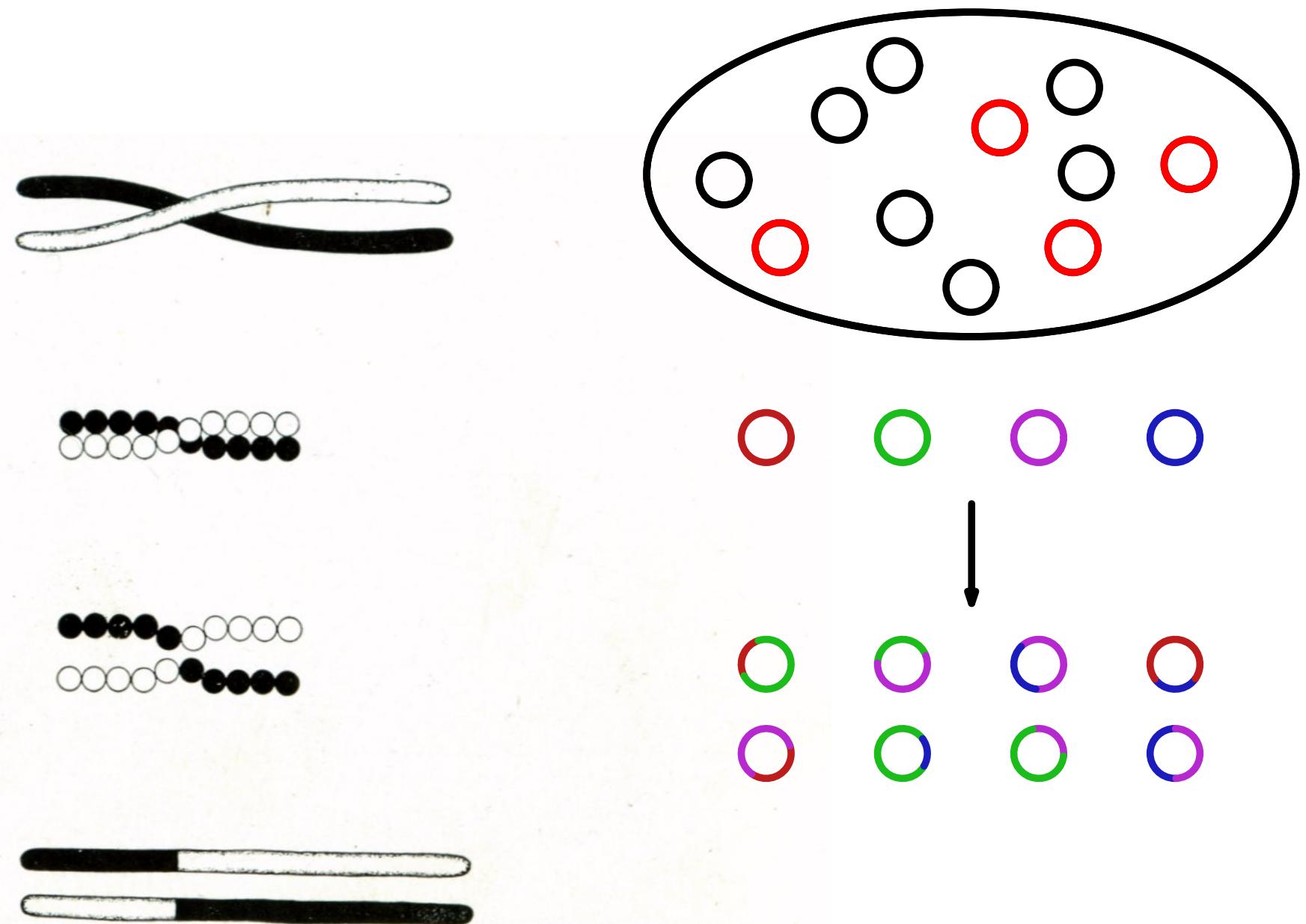
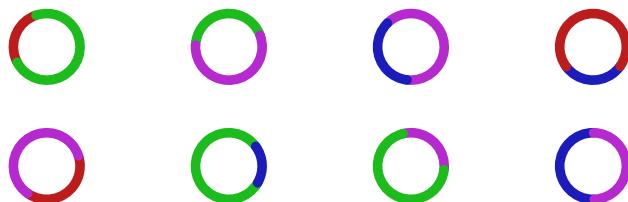
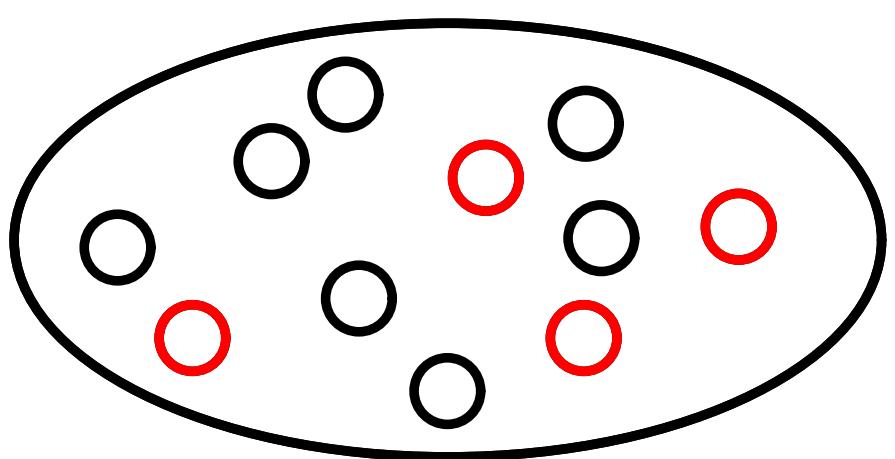
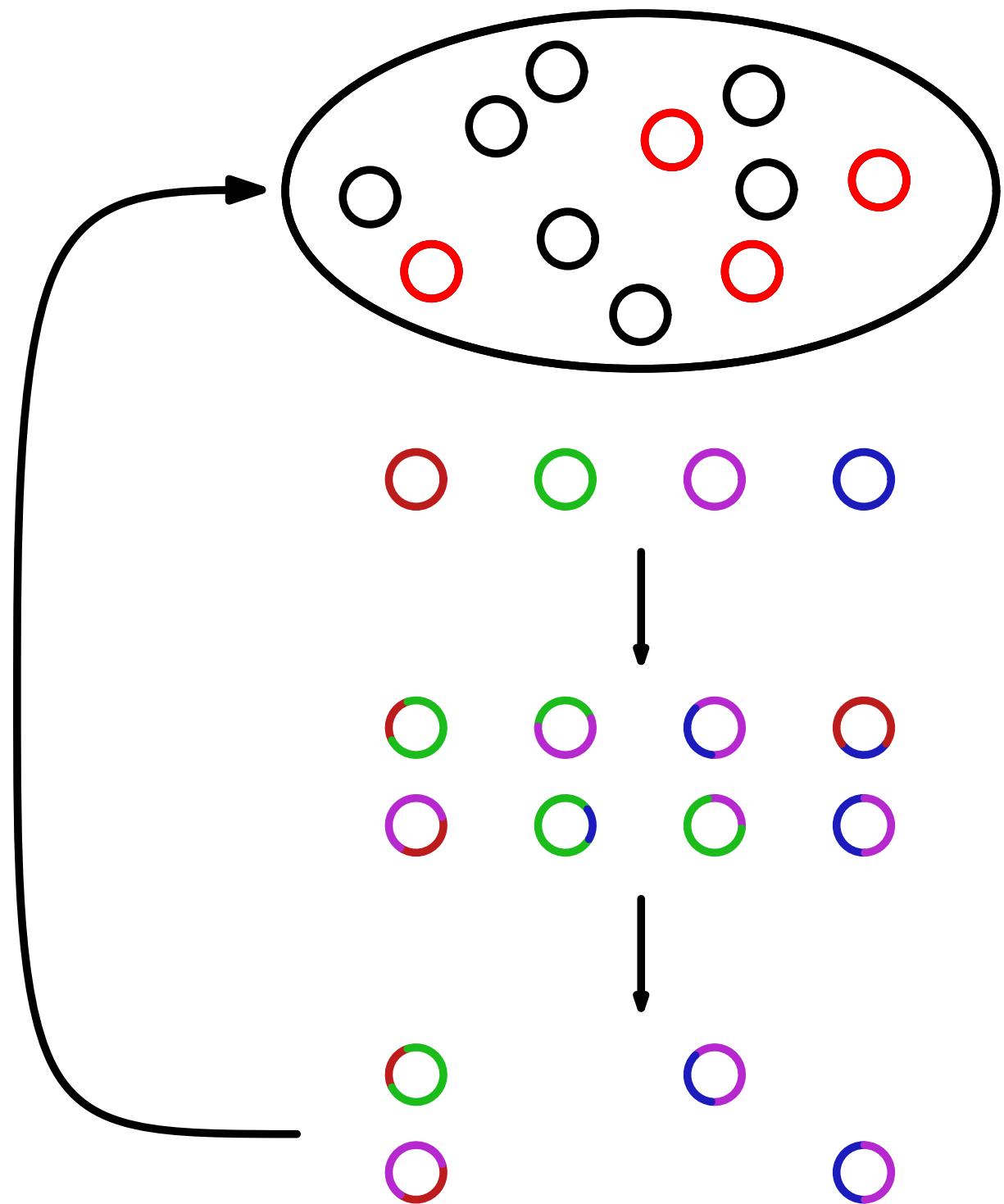
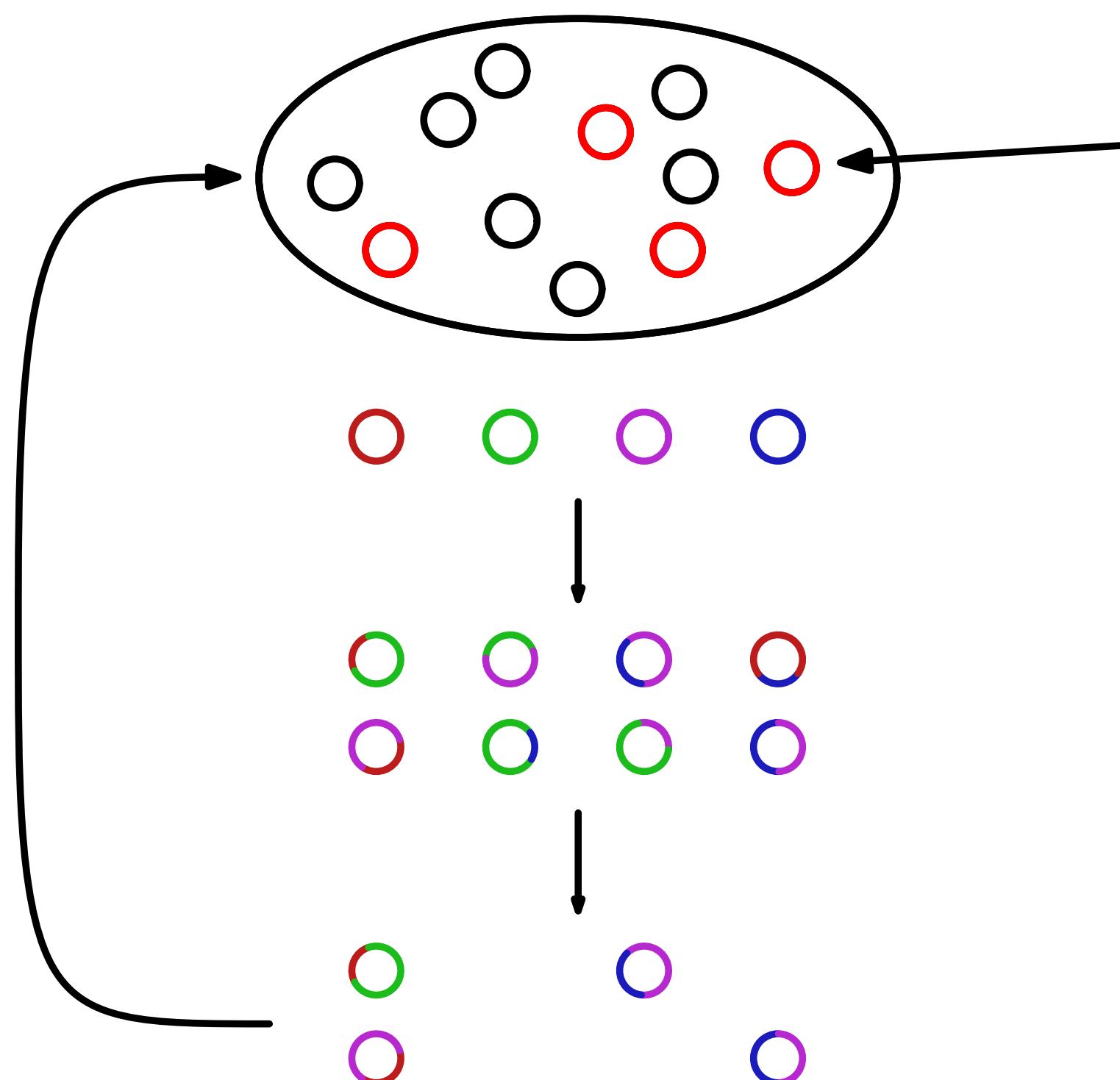


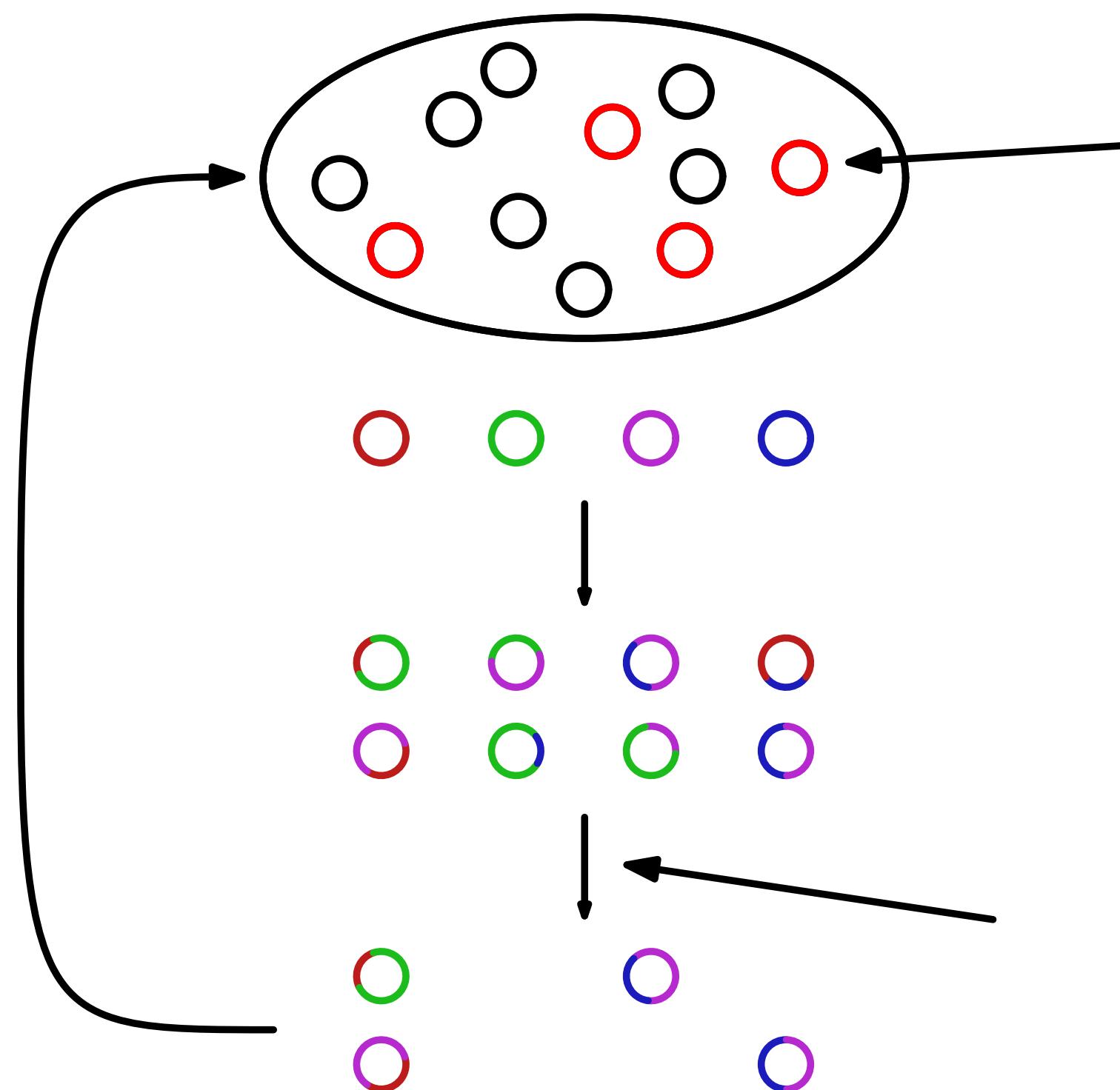
Fig. 64. Scheme to illustrate a method of crossing over of the chromosomes.

en.wikipedia.org/wiki/Chromosomal_crossover



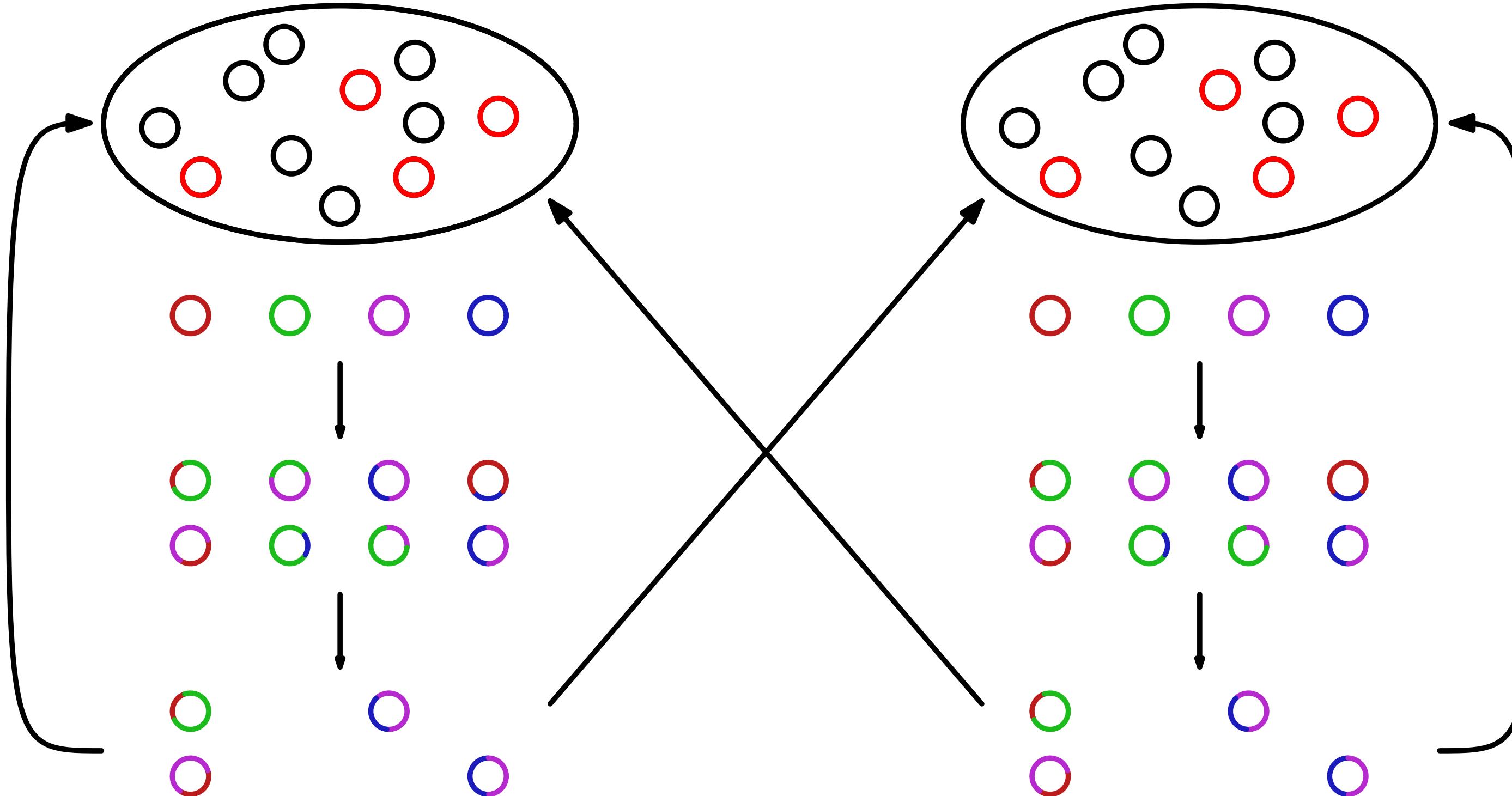






Feasible

Infeasible



Modelling Video Games' Landscapes by Means of Genetic Terrain Programming - A New Approach for Improving Users' Experience

Miguel Frade¹, F. Fernandez de Vega², and Carlos Cotta³

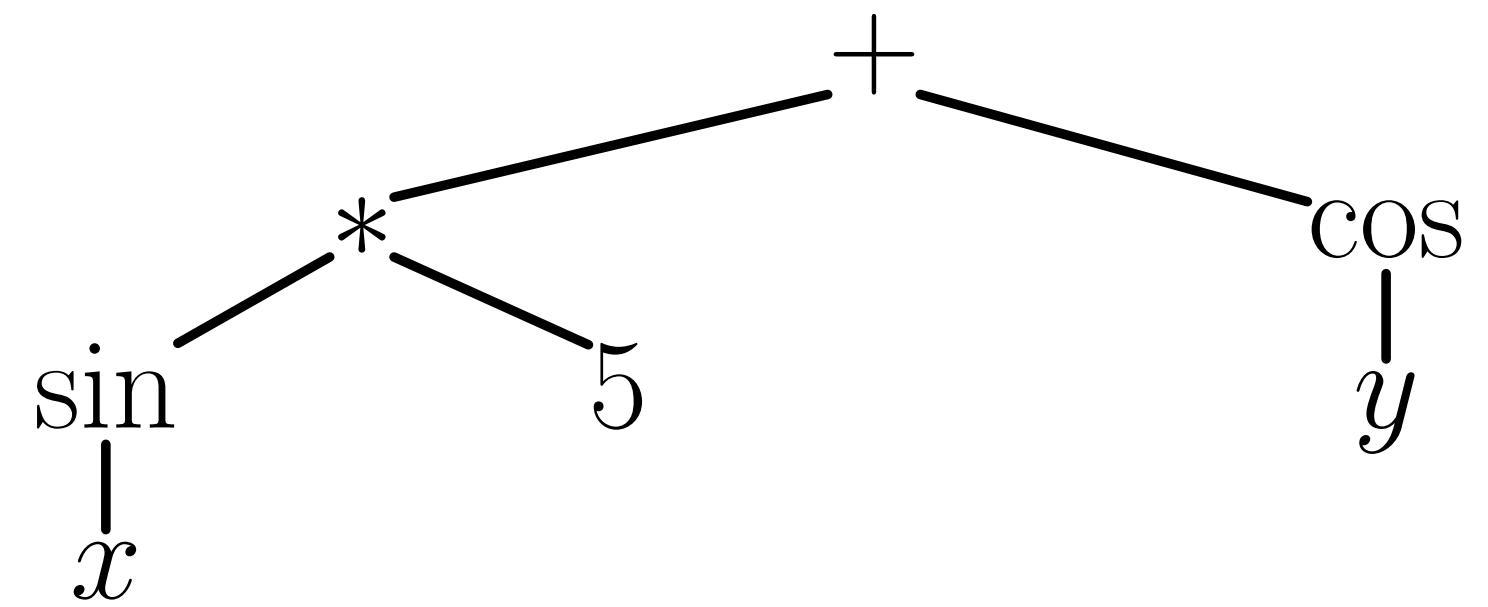
¹ School of Technology and Management, Polytechnic Institute of Leiria, Portugal
mfrade@estg.ipoleiria.pt

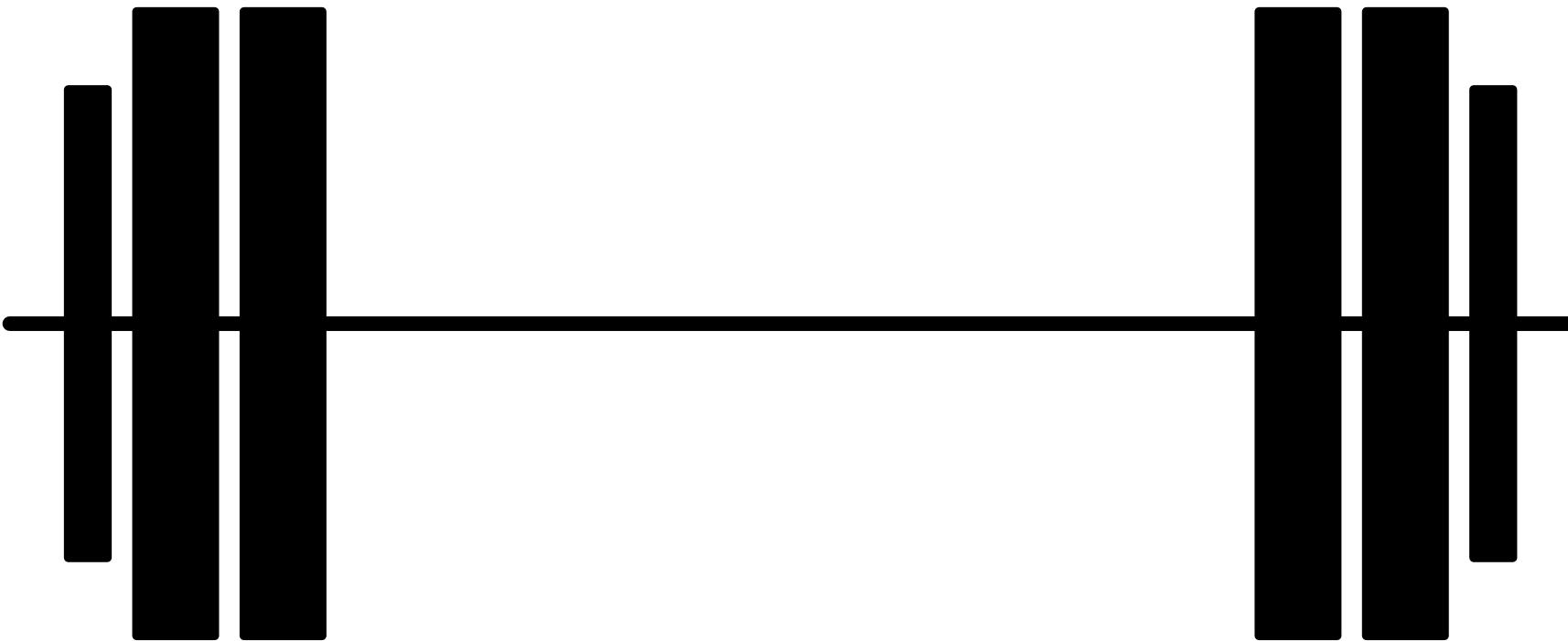
² Centro Universitario de Merida, Universidad de Extremadura, Spain
fcofdez@unex.es

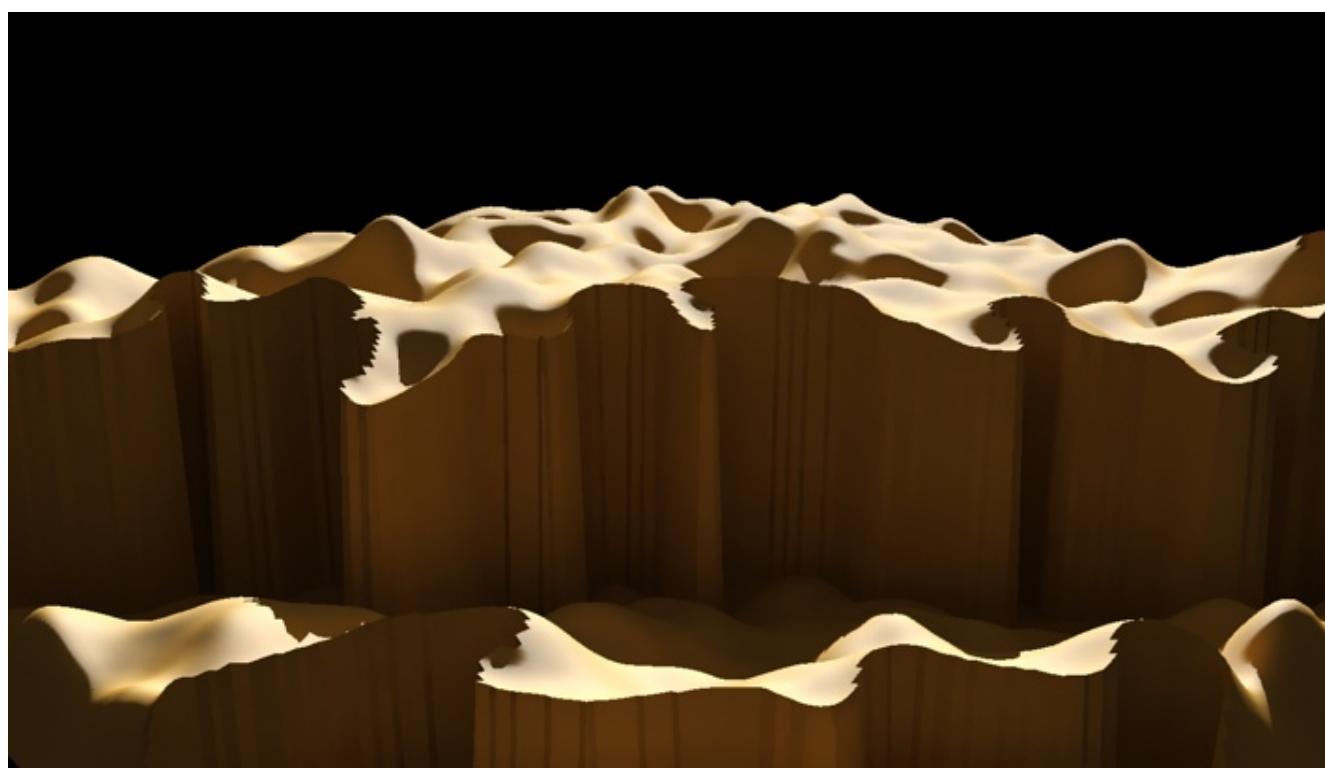
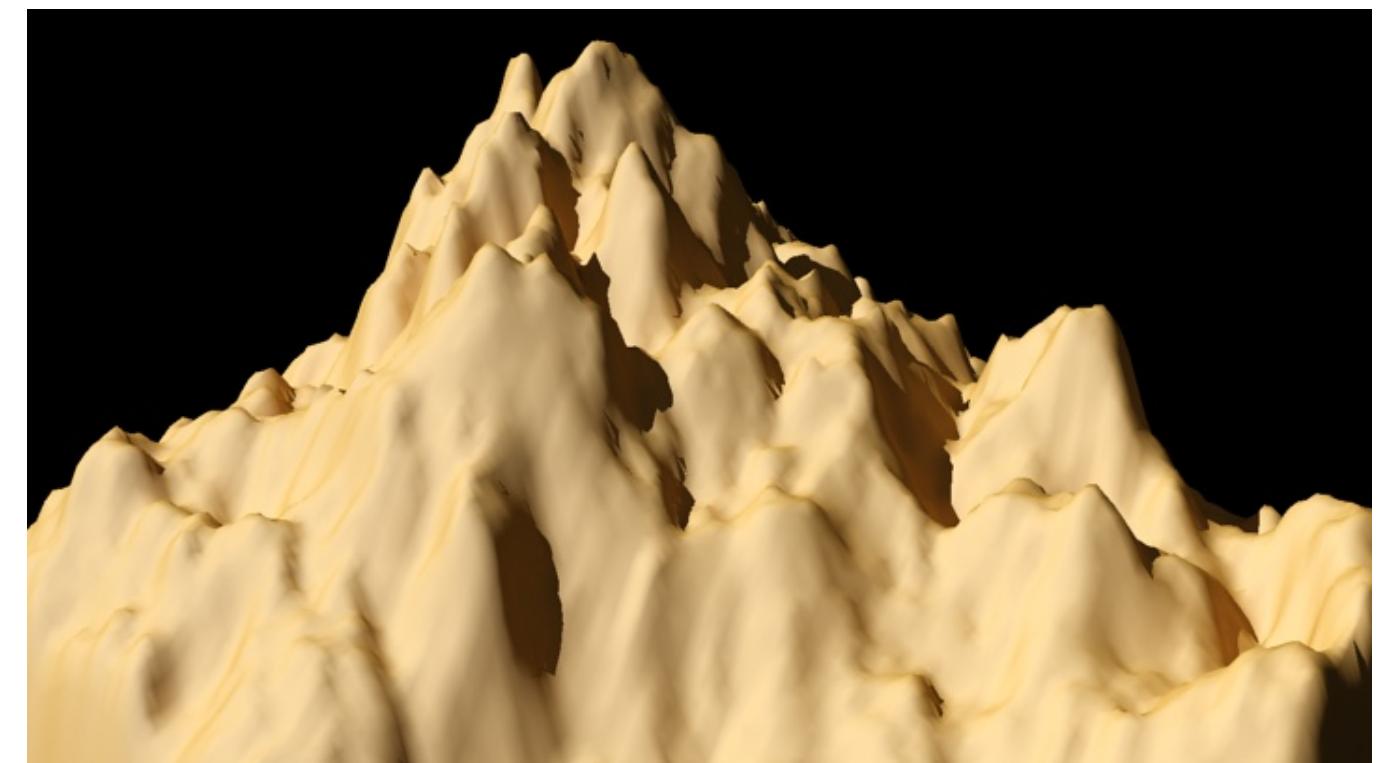
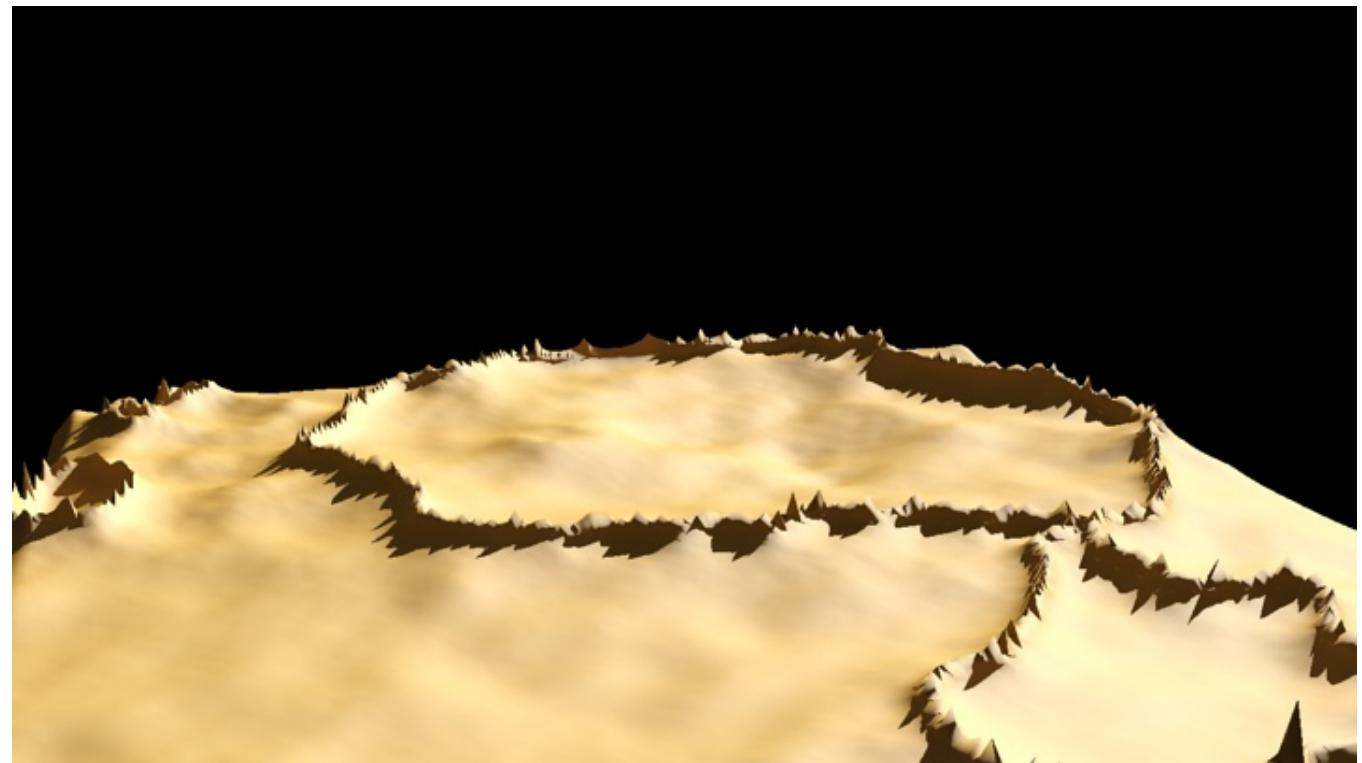
³ ETSI Informática, Campus de Teatinos, Universidad de Málaga, Spain
ccottap@lcc.uma.es

$$\textcolor{red}{O} = f(x,y)$$

$$\textcolor{red}{O} = f(x, y)$$







Assignment 1

Name two benefits of working with a terrain generation function instead of a mesh.

A Procedural Method for Automatic Generation of Spelunky Levels

Walaa Baghdadi¹, Fawzya Shams Eddin¹, Rawan Al-Omari¹, Zeina Alhalawani¹,
Mohammad Shaker² and Noor Shaker³

¹Information Technology Engineering of Damascus, Damascus, Syria

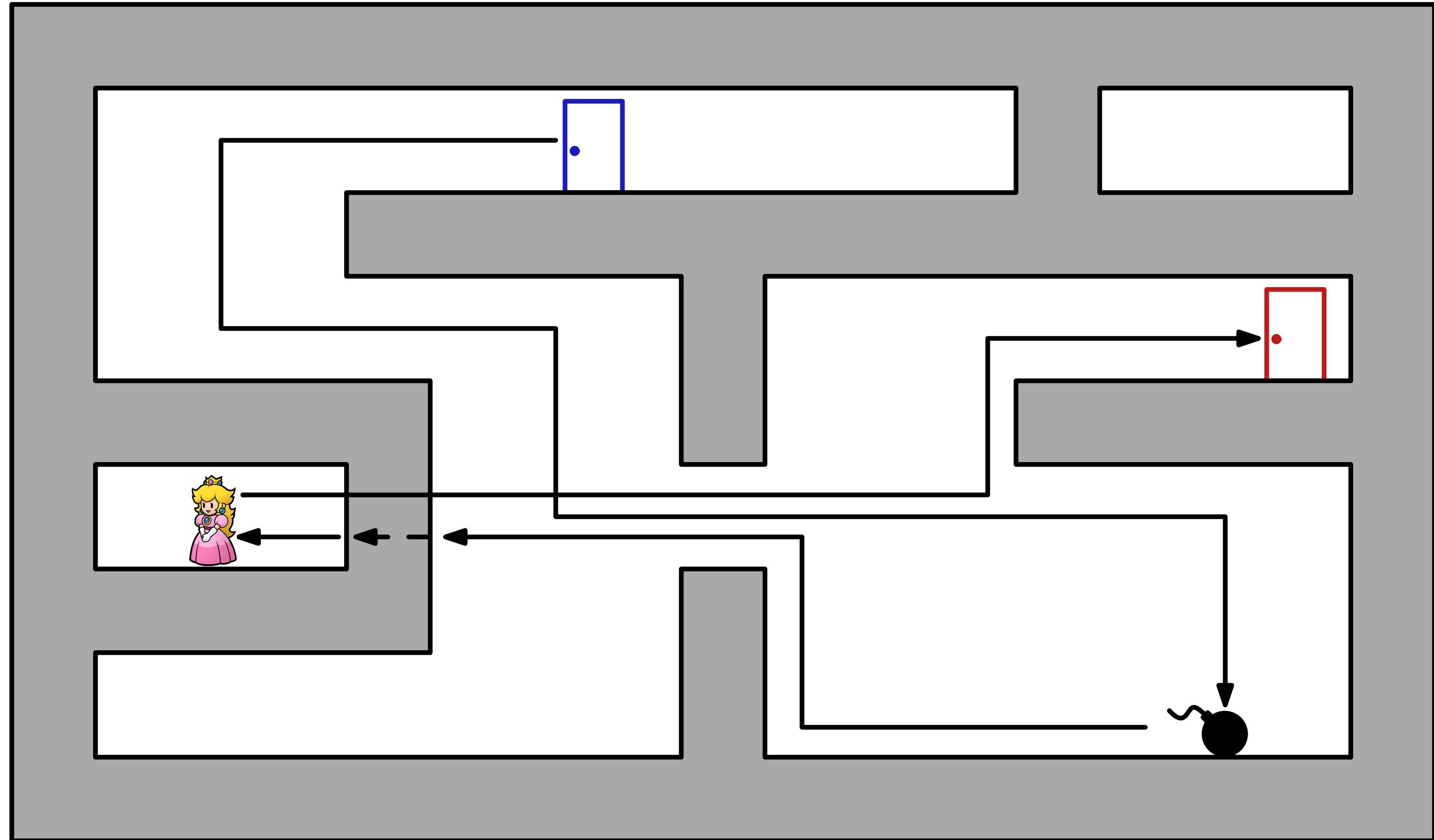
²Joseph Fourier University, Grenoble, France

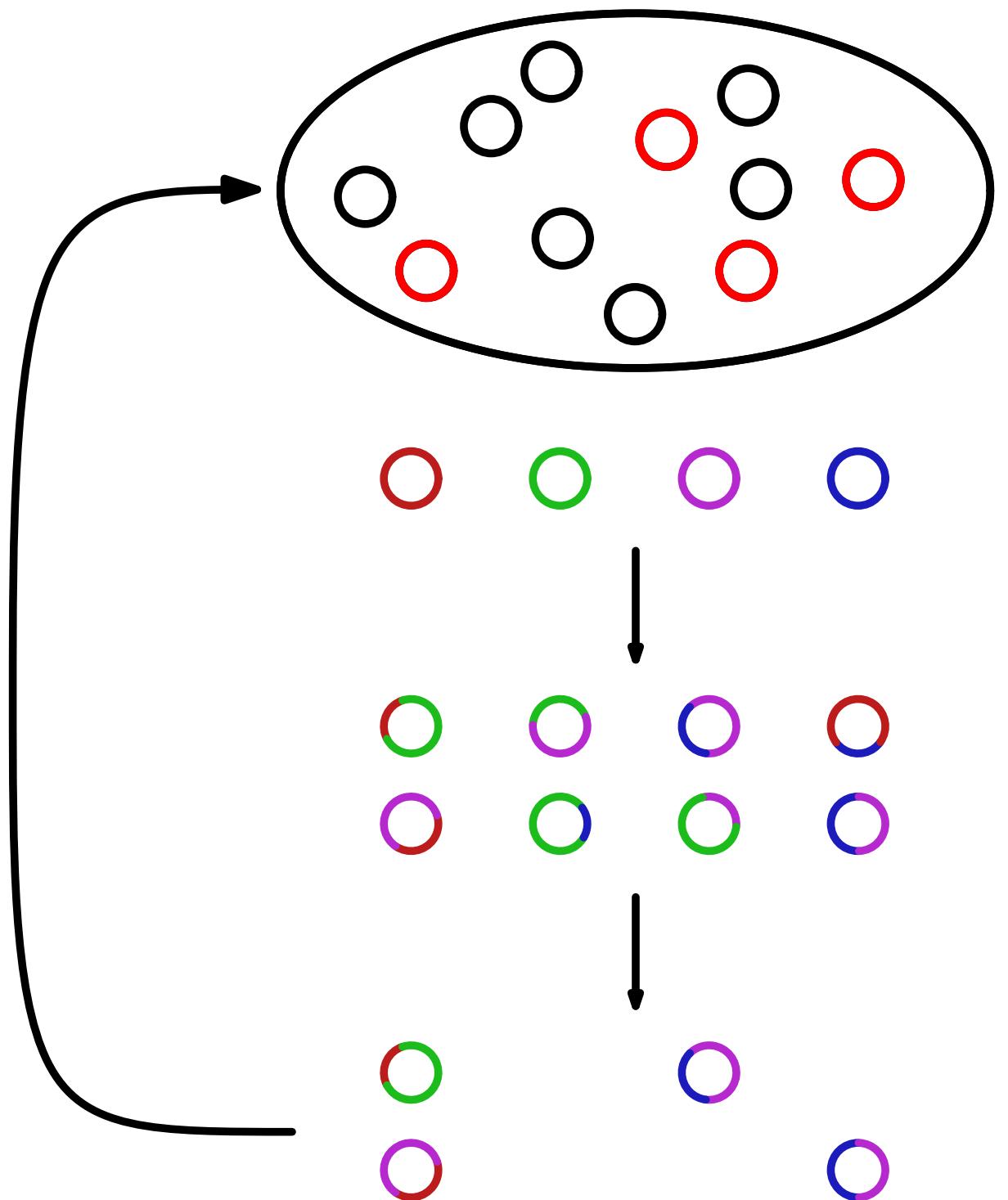
³IT University of Copenhagen, Copenhagen, Denmark

walaabaghdadi, fawziashamseddin91, rawan.alomari91, zeina.helwani,
mohammadshakergr@gmail.com, nosh@itu.dk



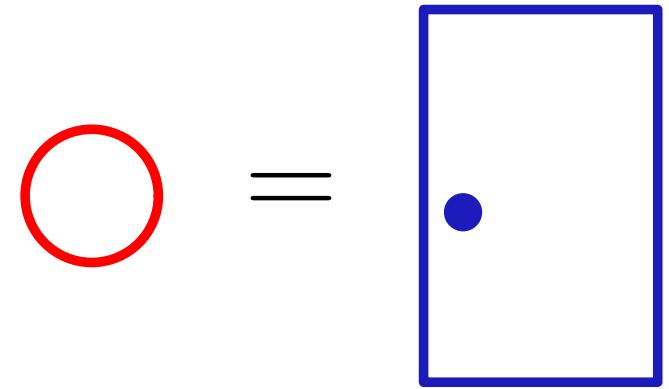
spelunkyworld.com



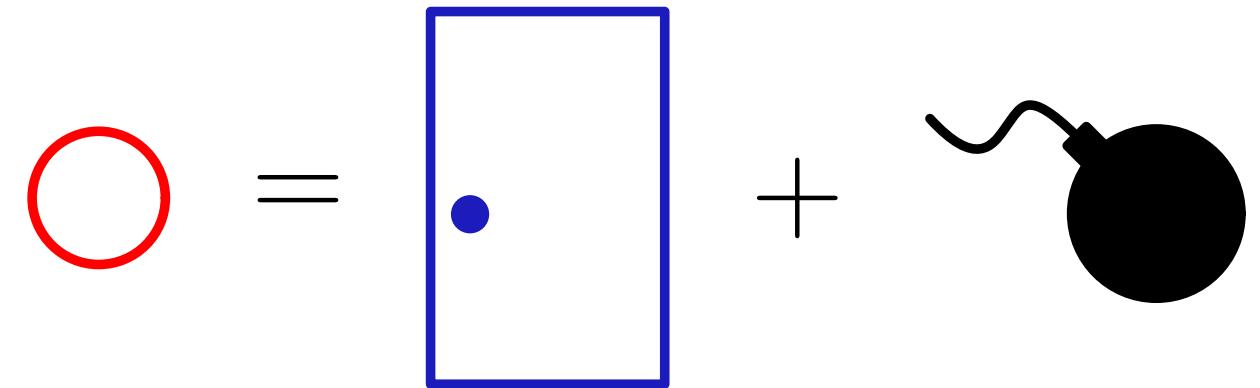


○ =

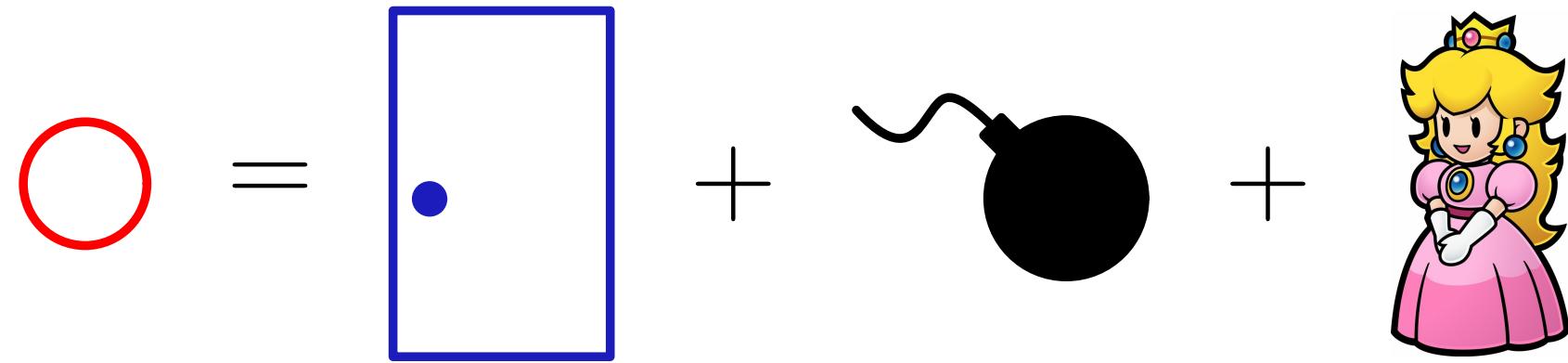
0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15



0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15



0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

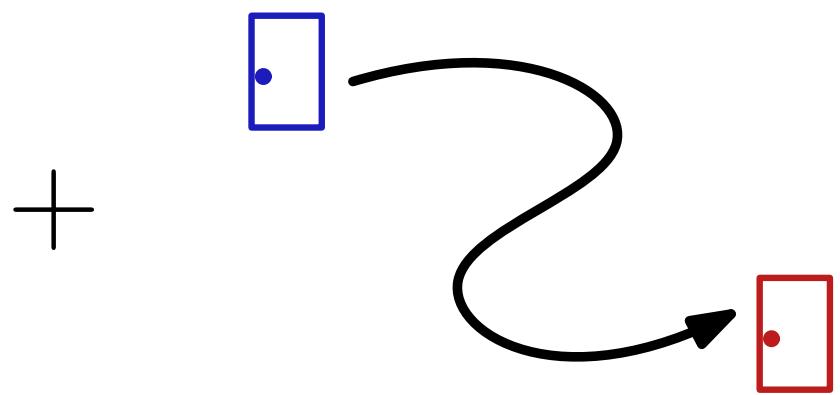


0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

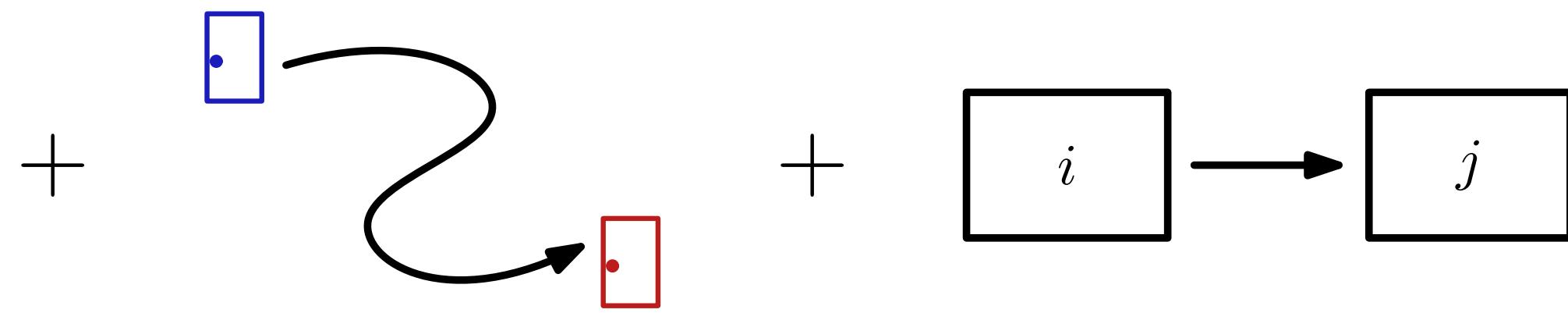
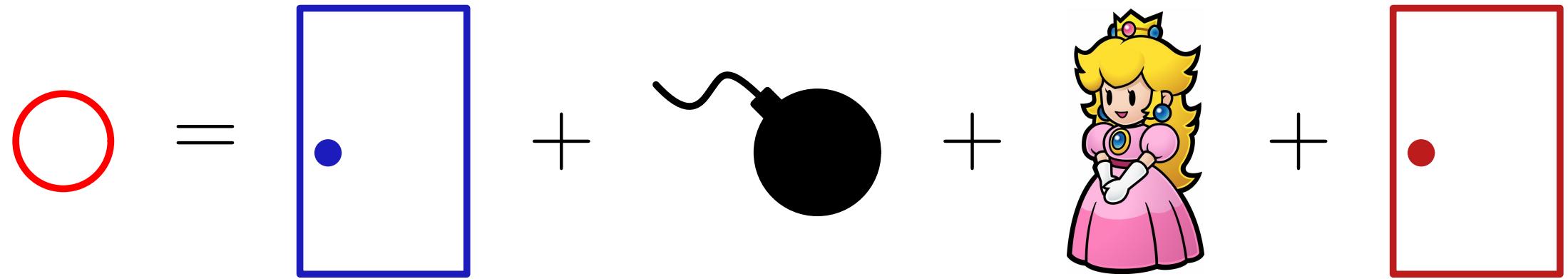
$$\textcircled{1} = \boxed{\bullet} + \text{Bomb} + \text{Princess Peach} + \boxed{\bullet}$$

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

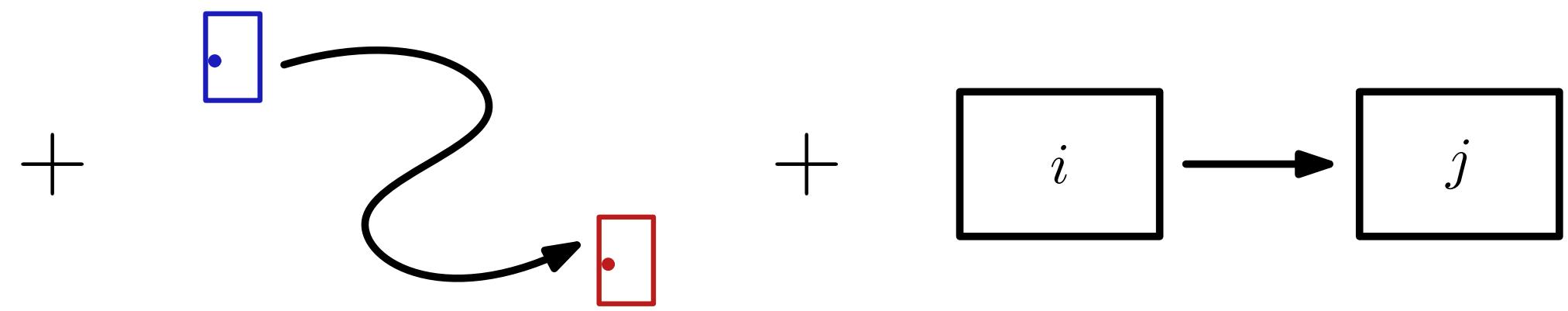
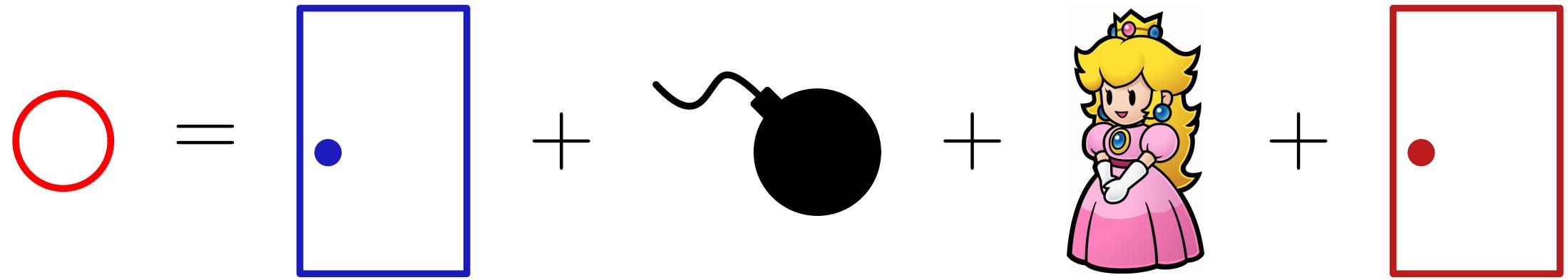
$$\textcircled{1} = \boxed{\bullet} + \text{Bomb} + \text{Princess} + \boxed{\bullet}$$



0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

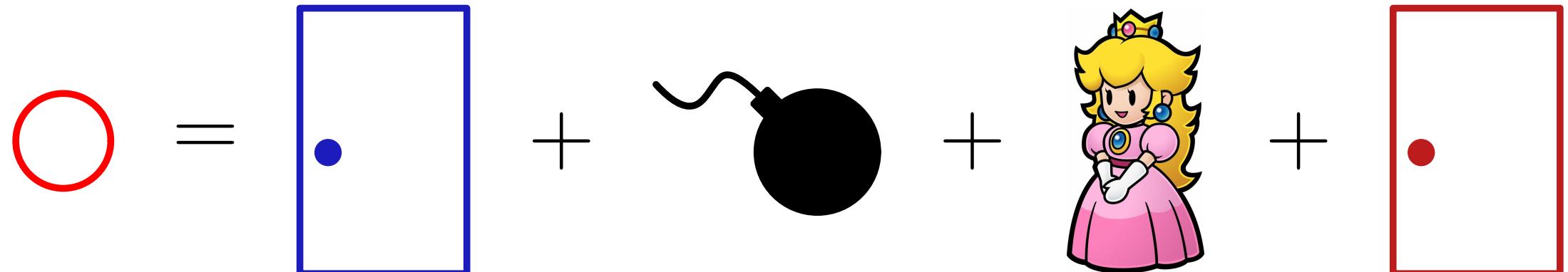


0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15


$$+ \quad \# \text{蛇}$$

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

GENOME

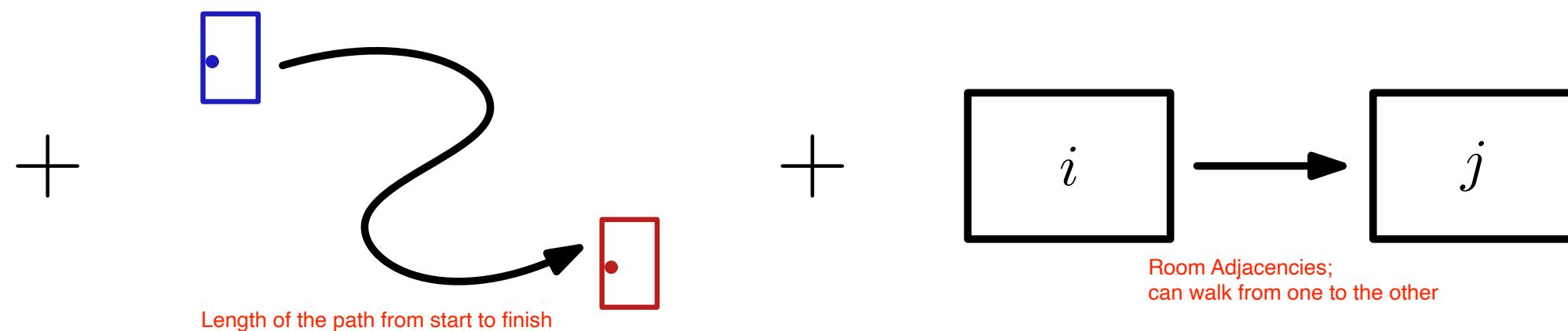


Start

Bomb

Damsel in distress

Finish



Length of the path from start to finish

Room Adjacencies;
can walk from one to the other

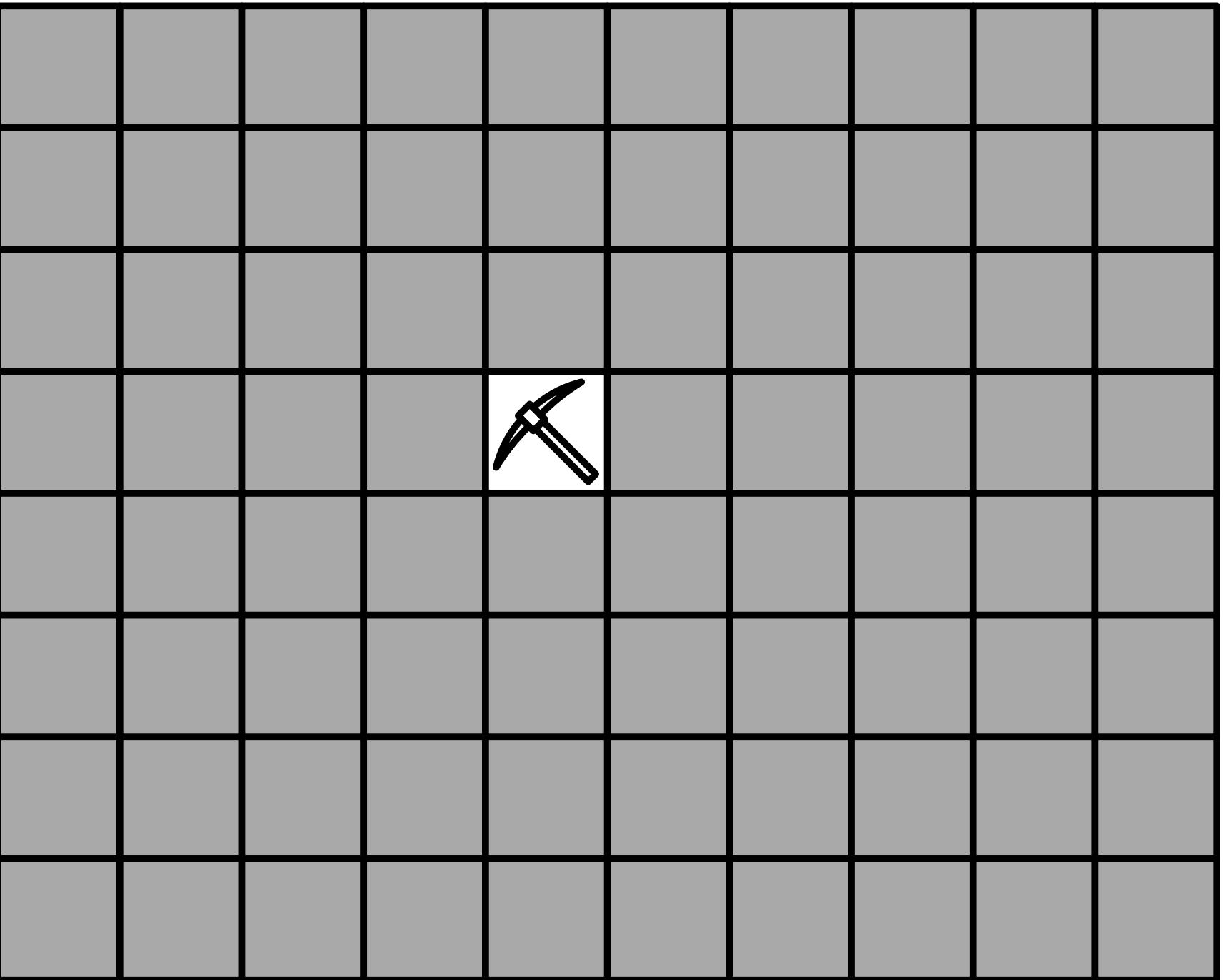


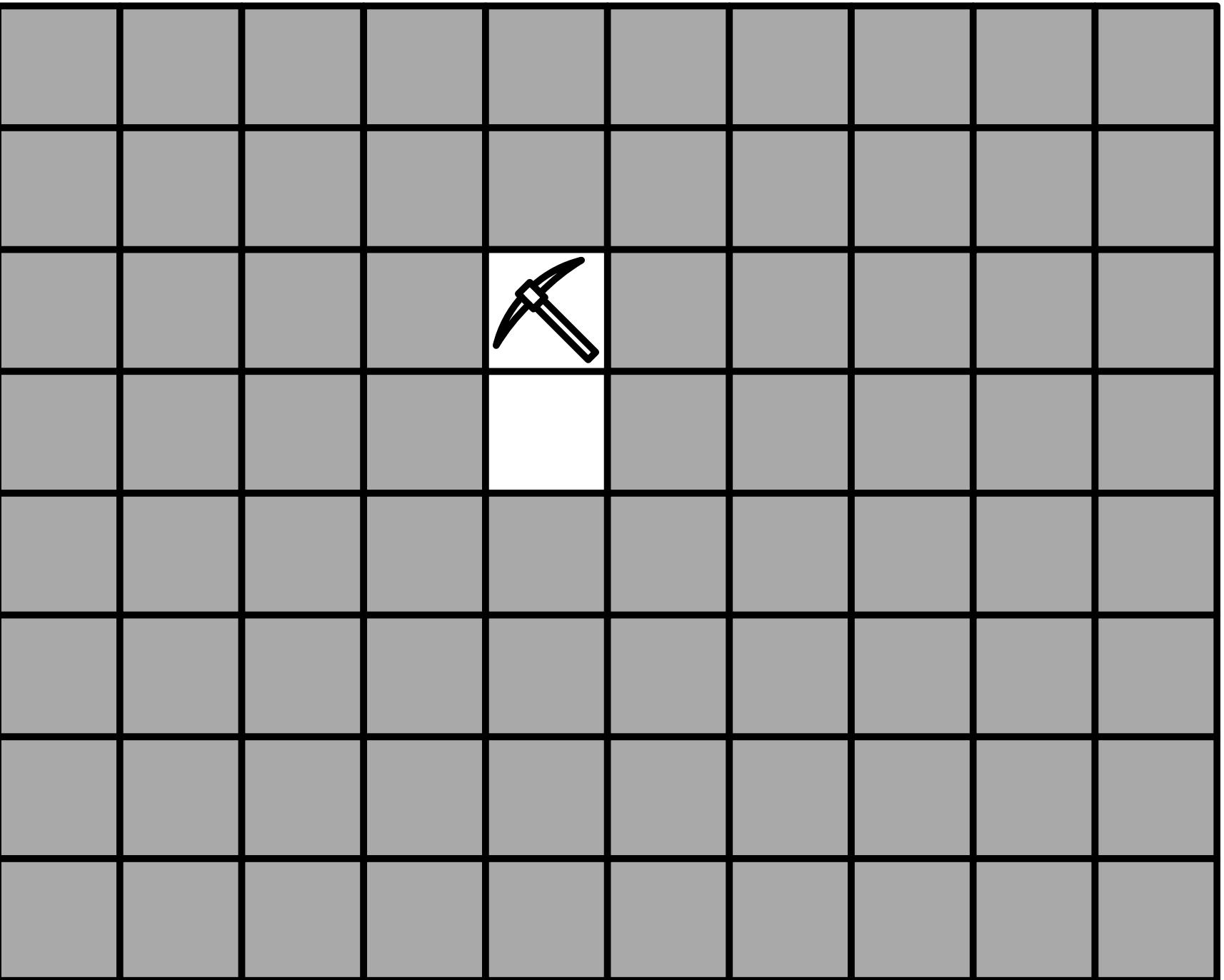
No. of enemies

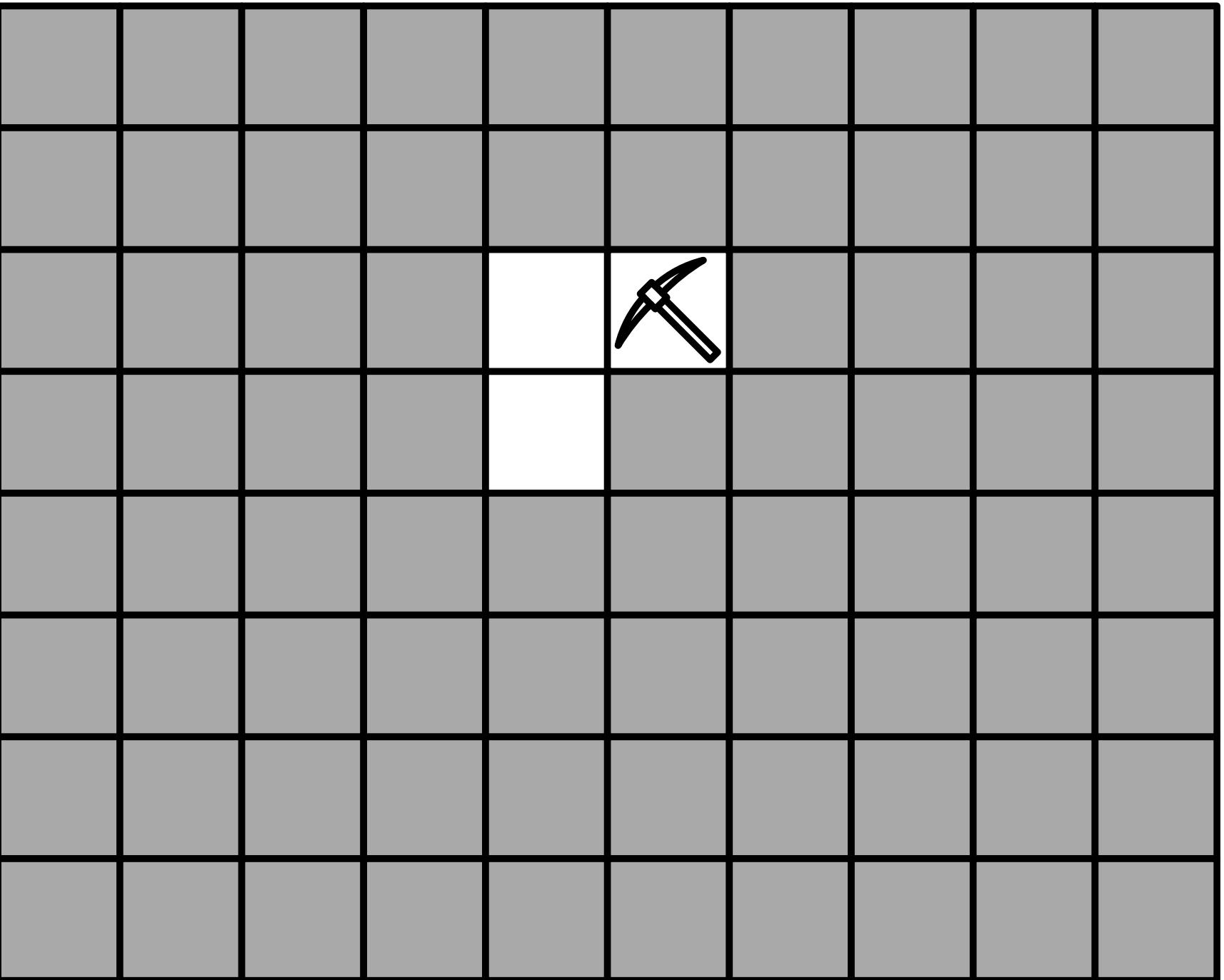
No. of rewards

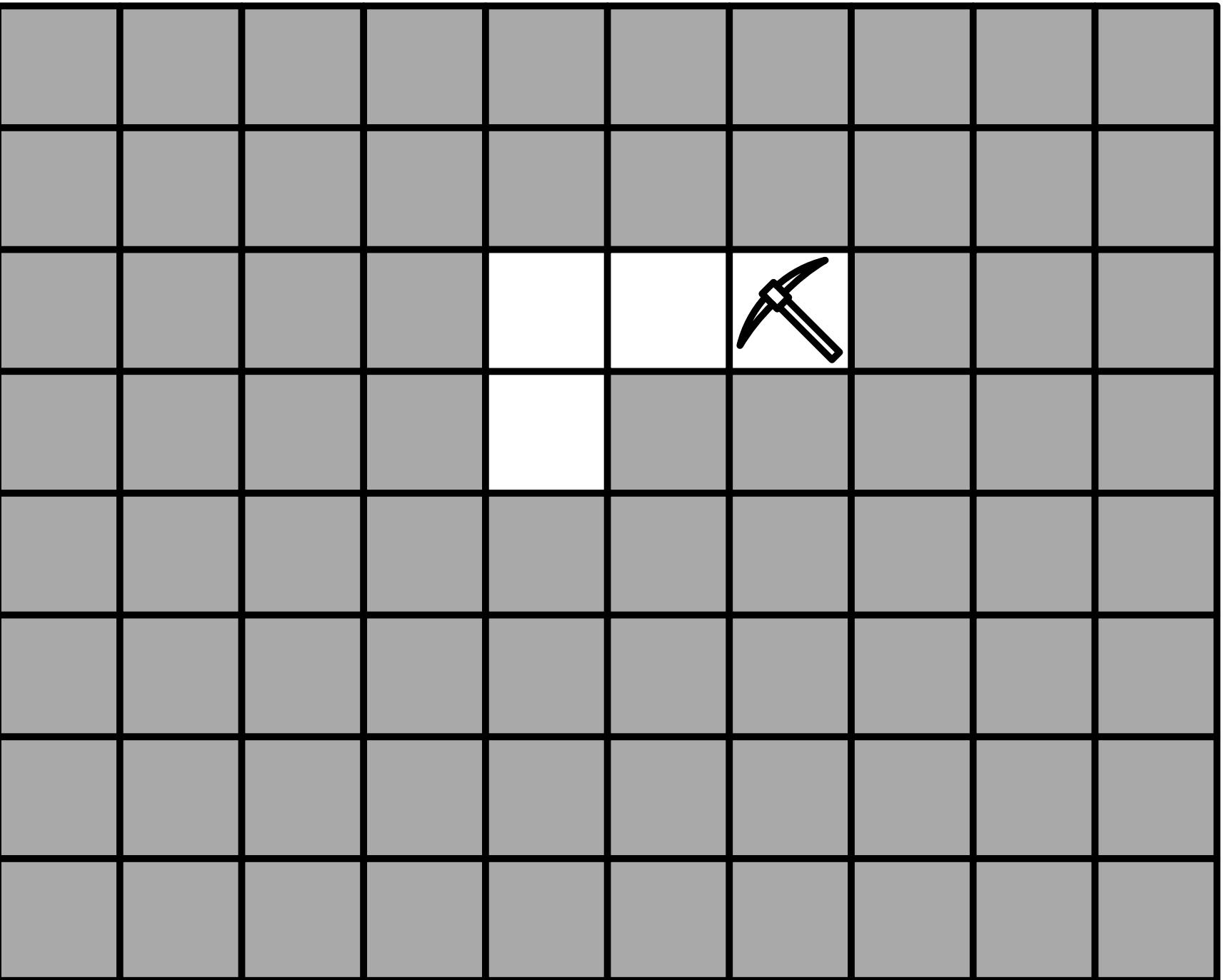
Rooms with indexes

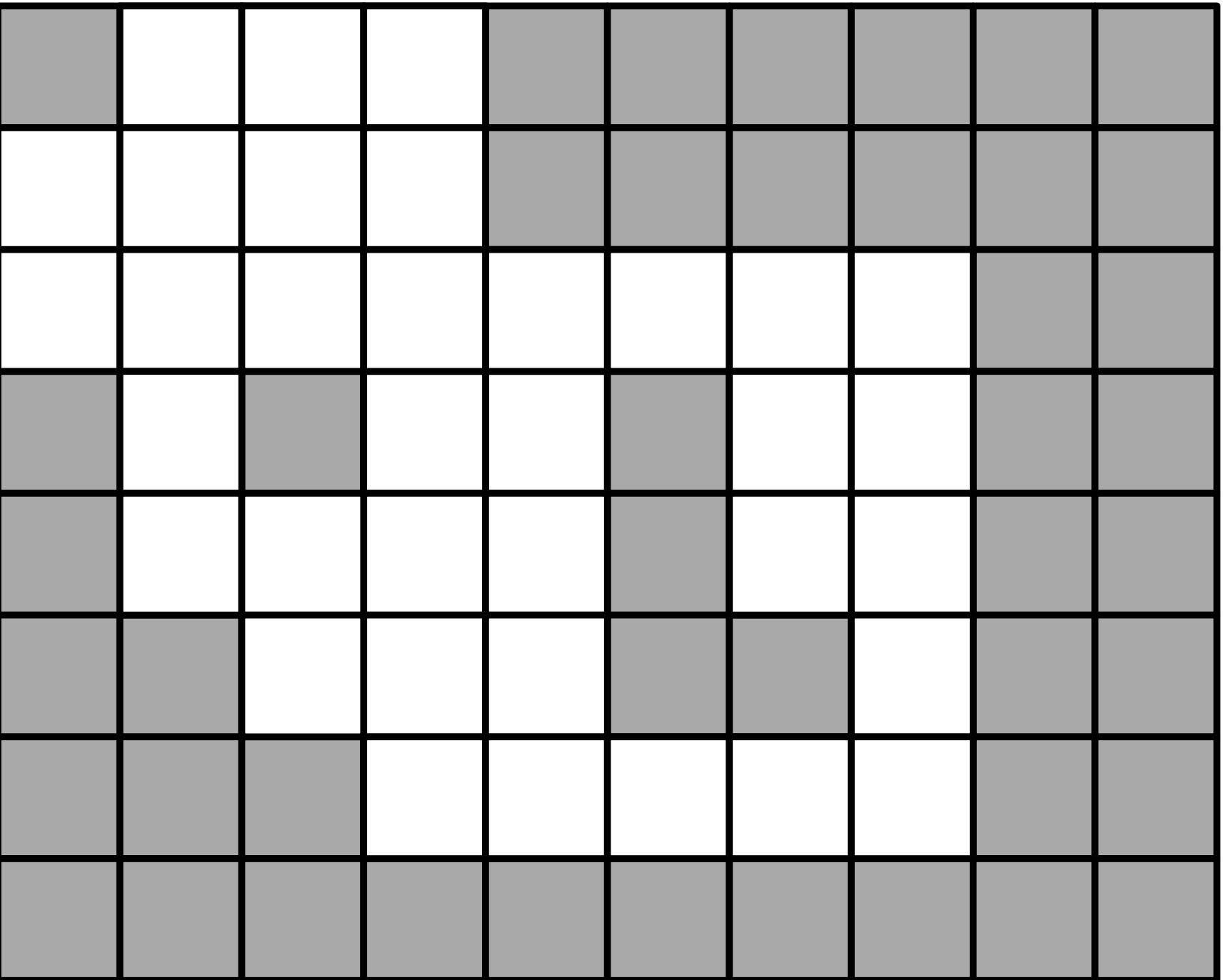
0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

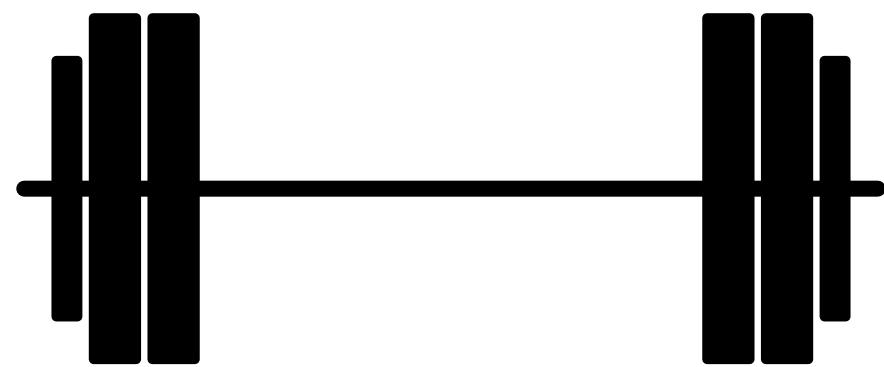


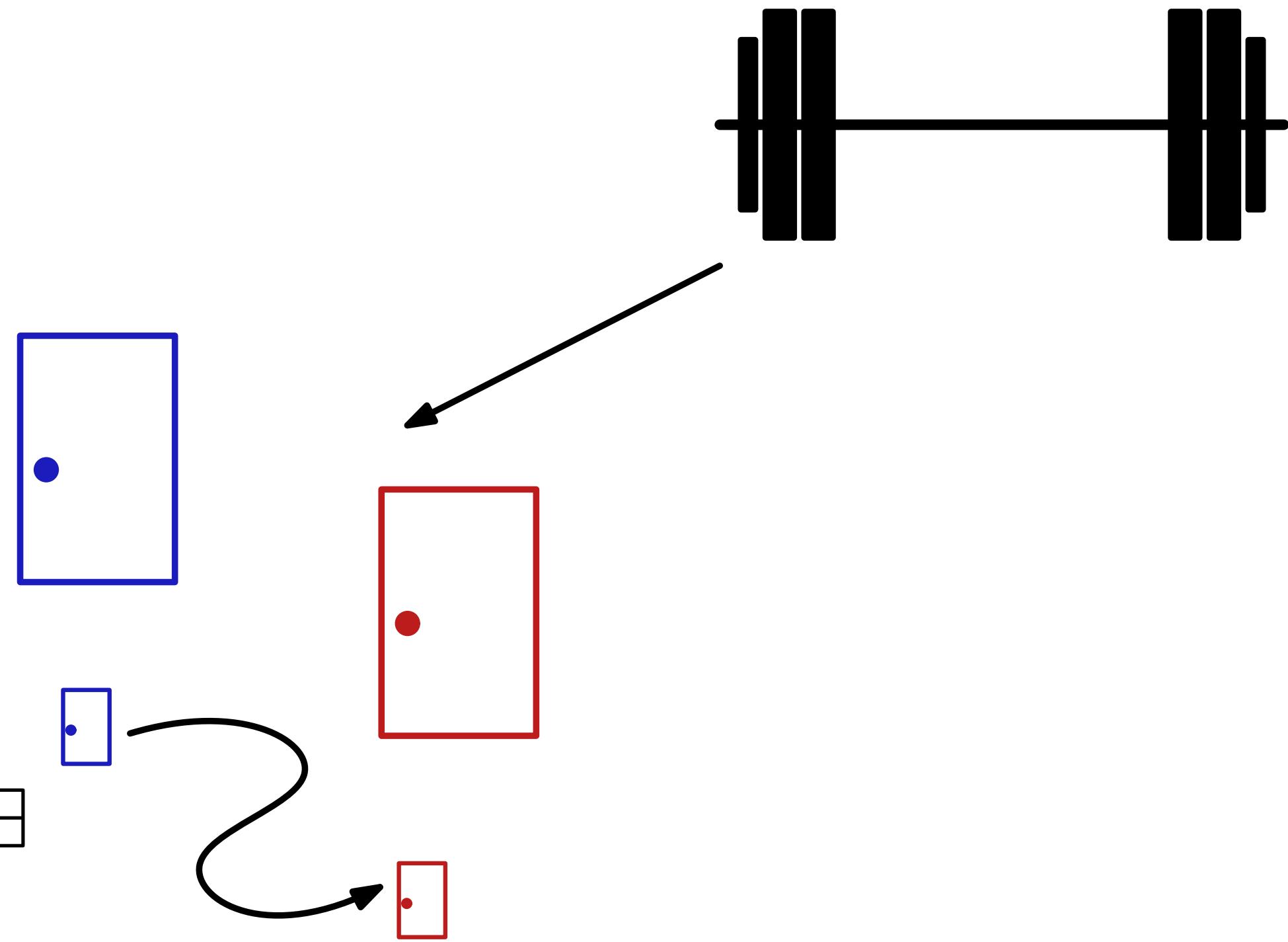


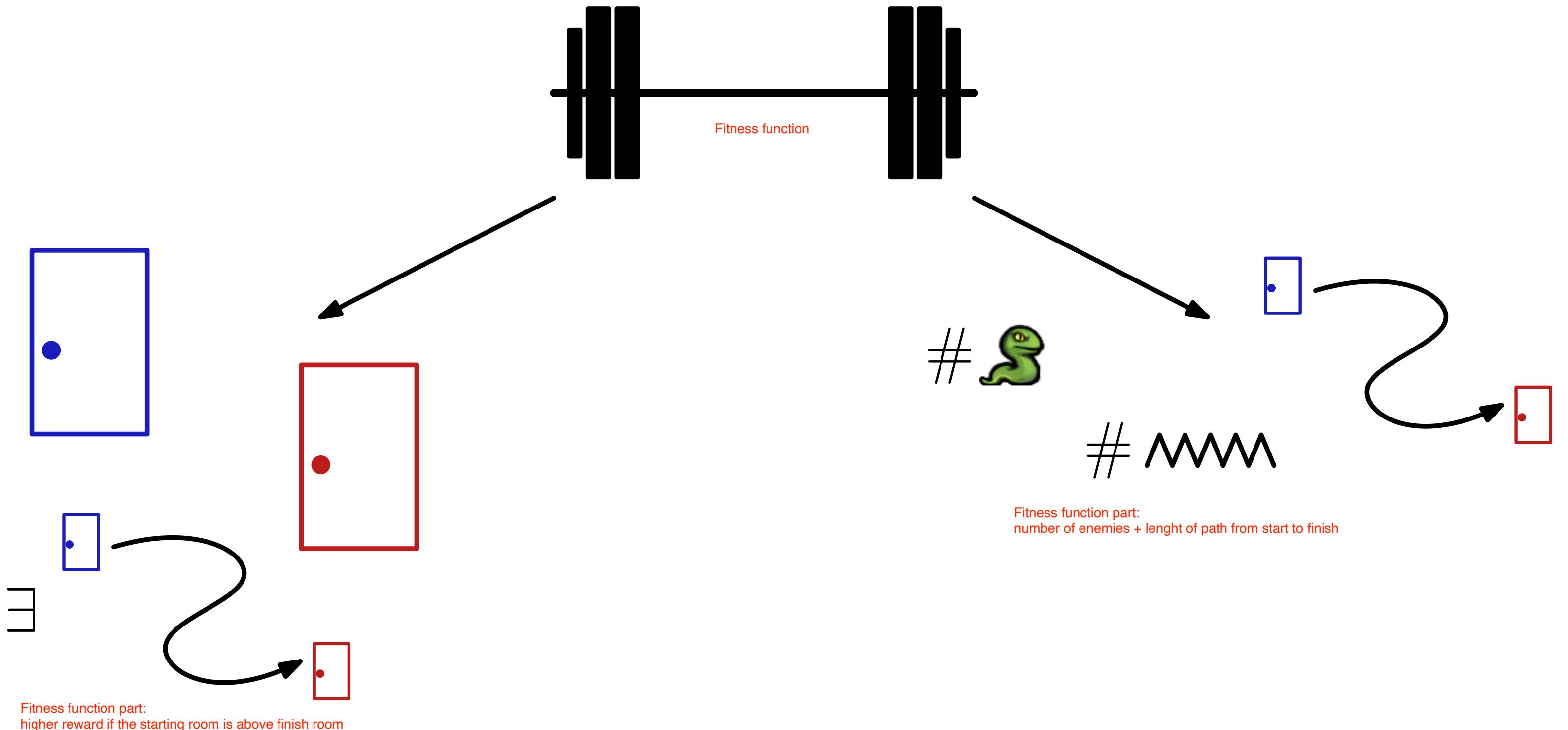




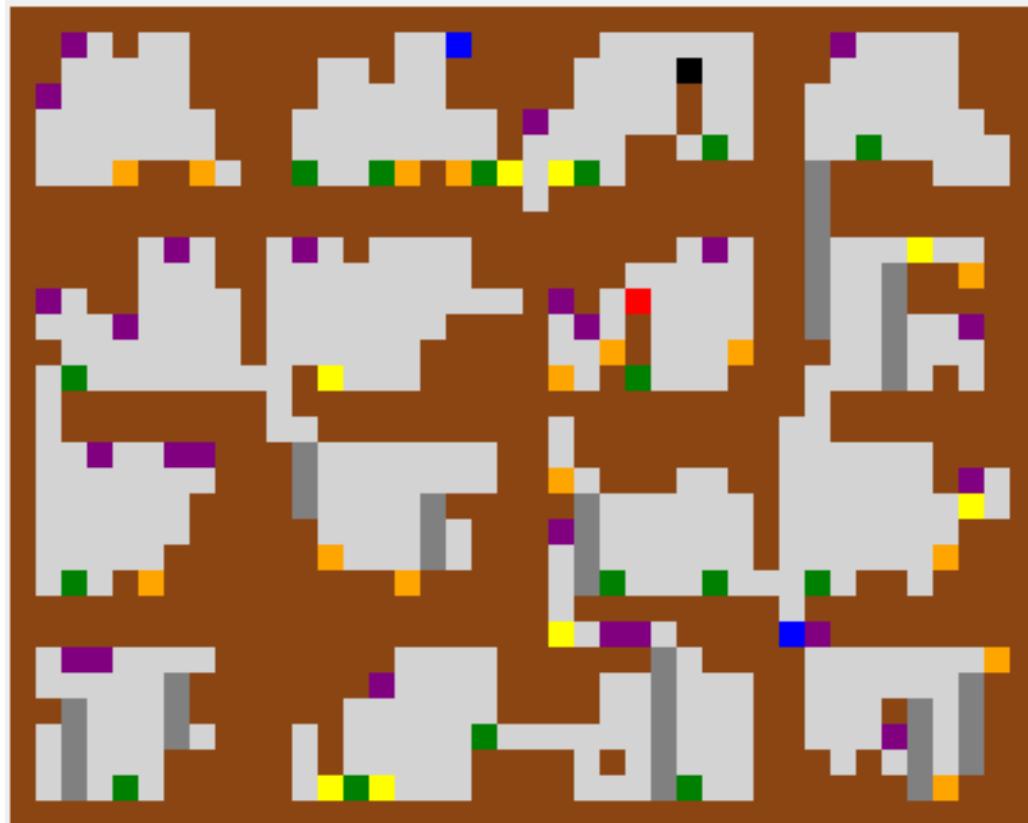
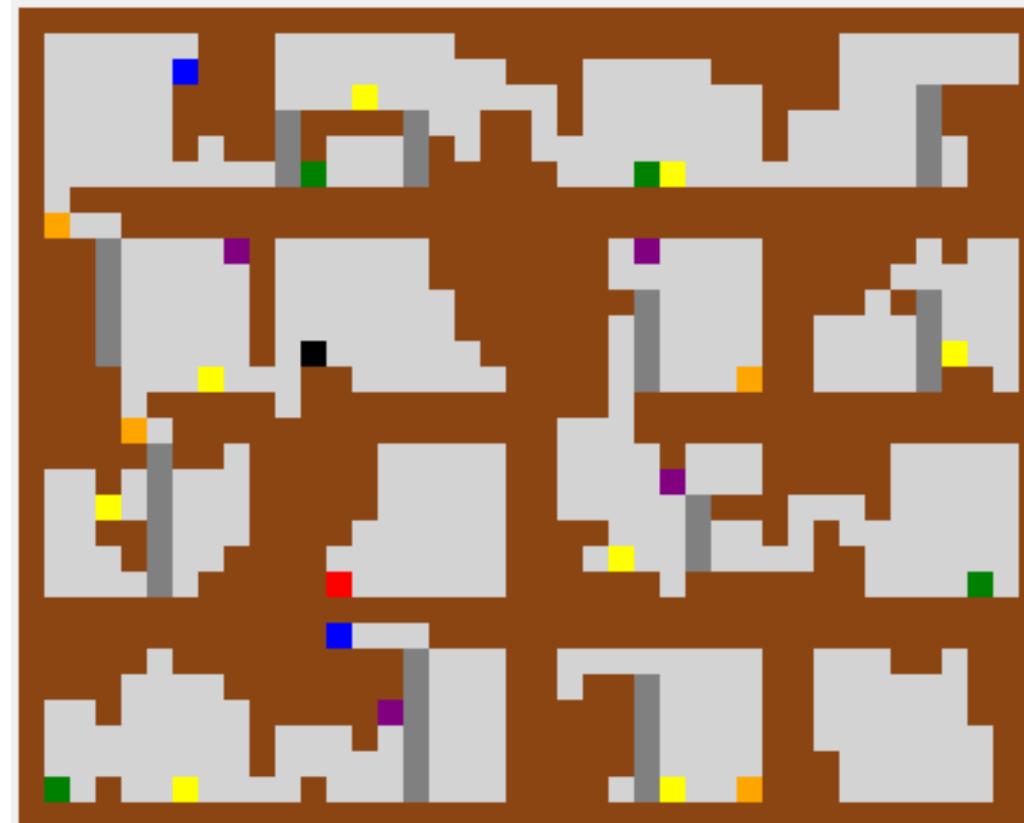








Fitness function: from left to right the difficulty is rewarded



Assignment 2

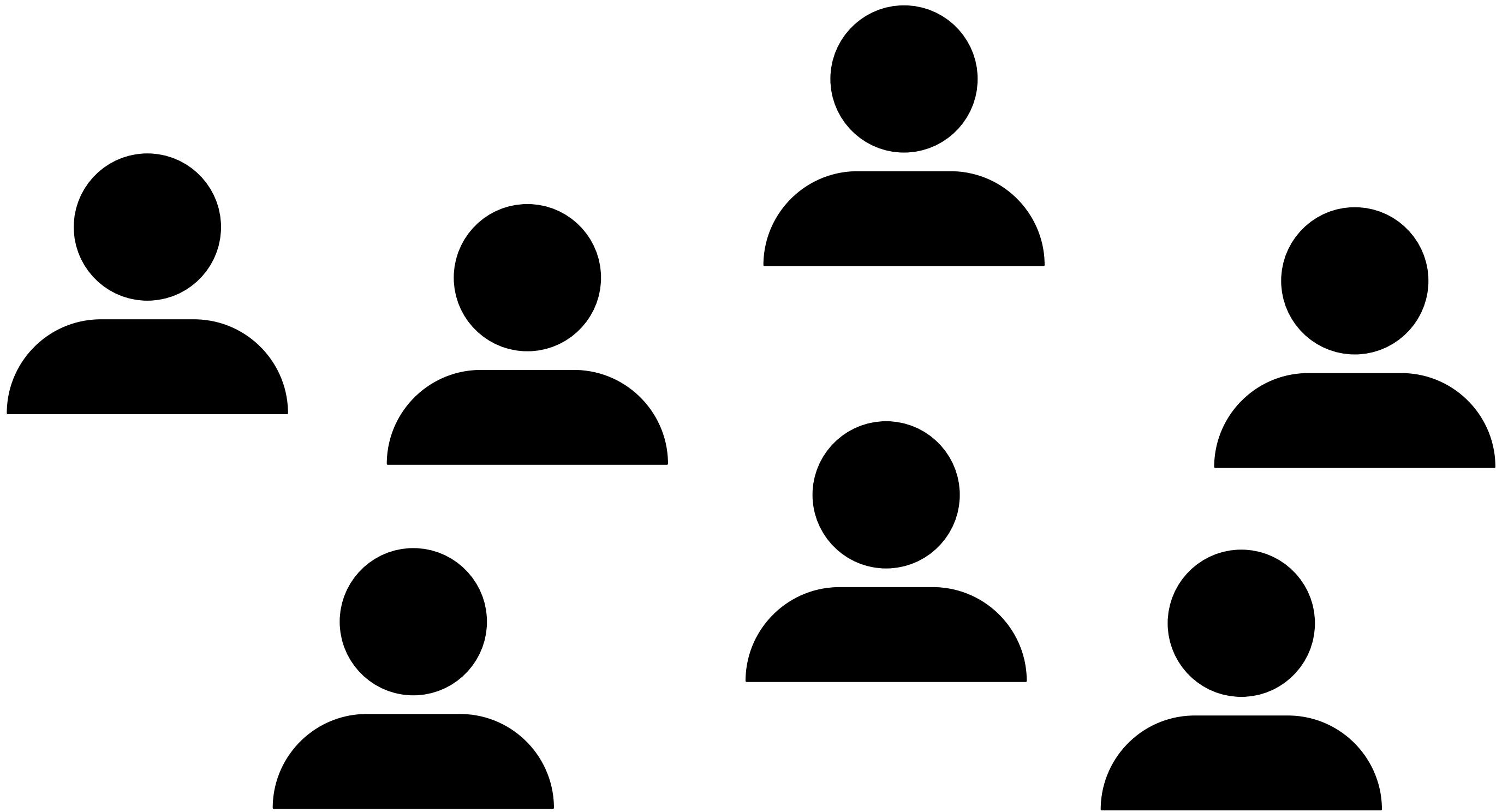
Why do we include the existence of a path to the exit in the fitness function, instead of throwing out all individuals where such a path does not exist?

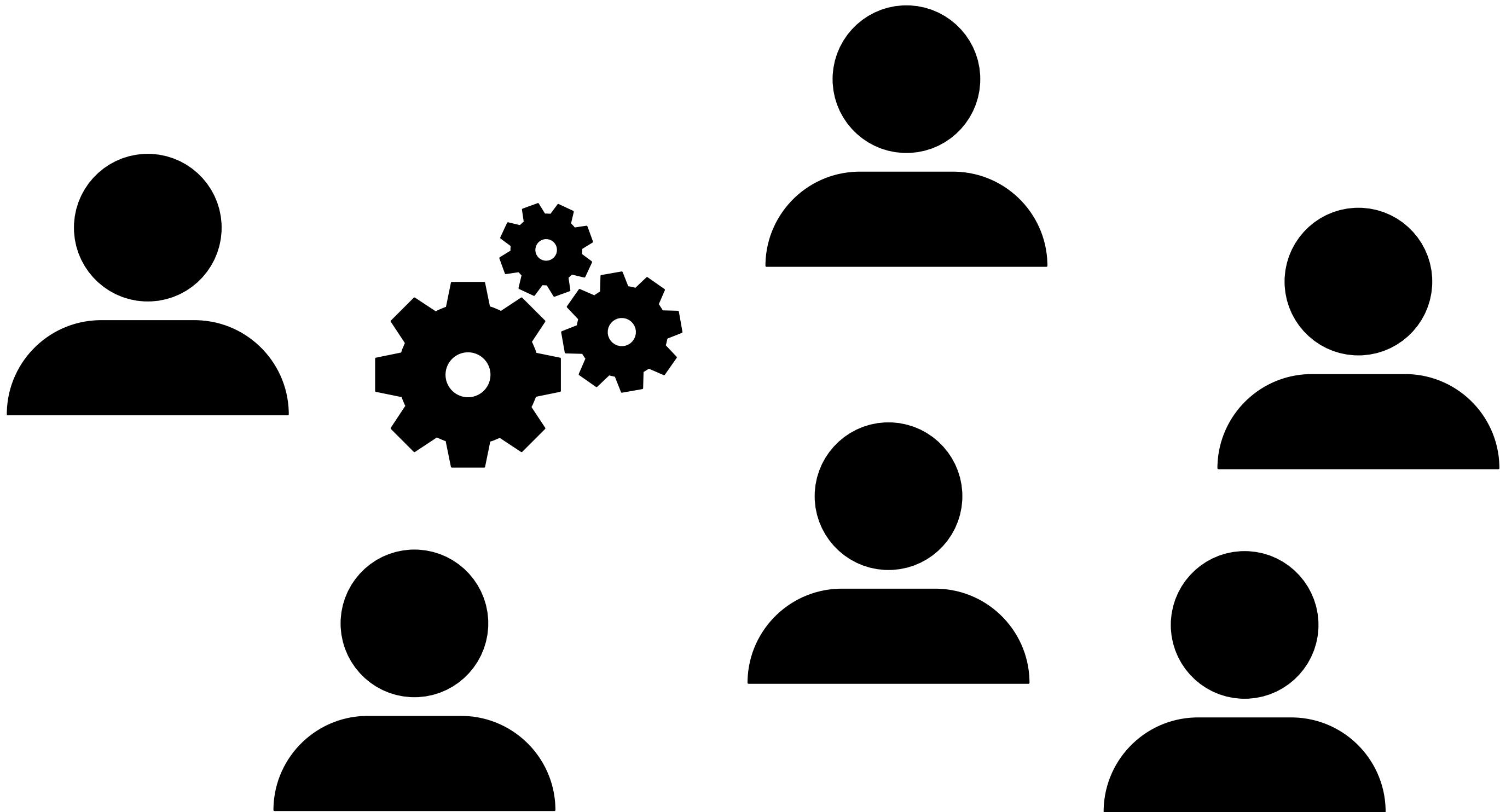
The ANGELINA Videogame Design System, Part I

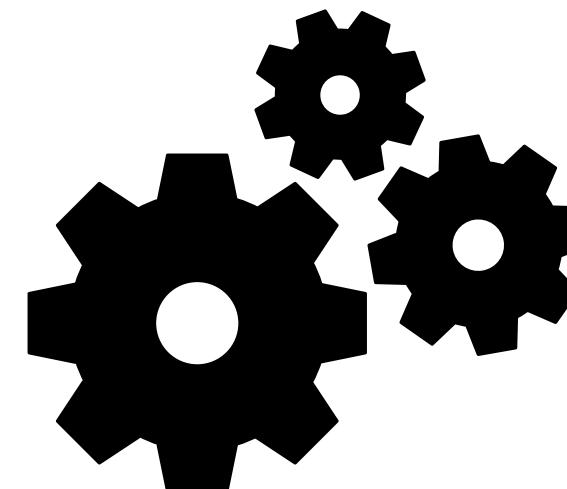
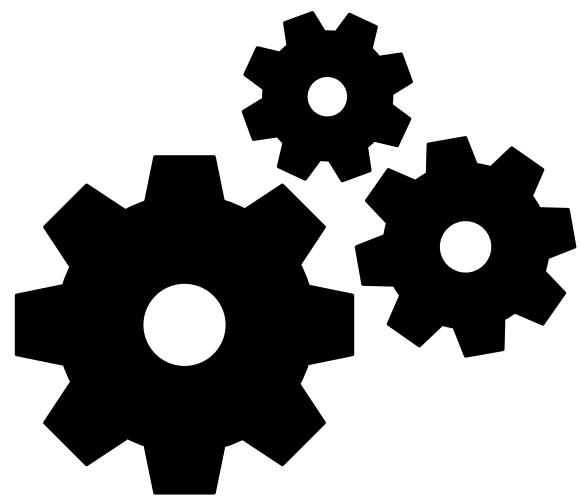
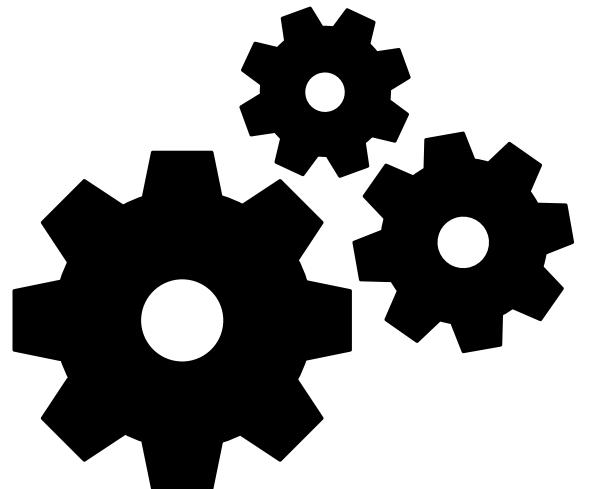
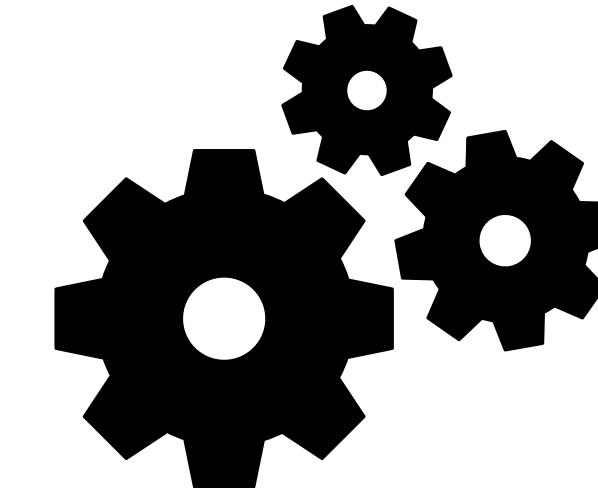
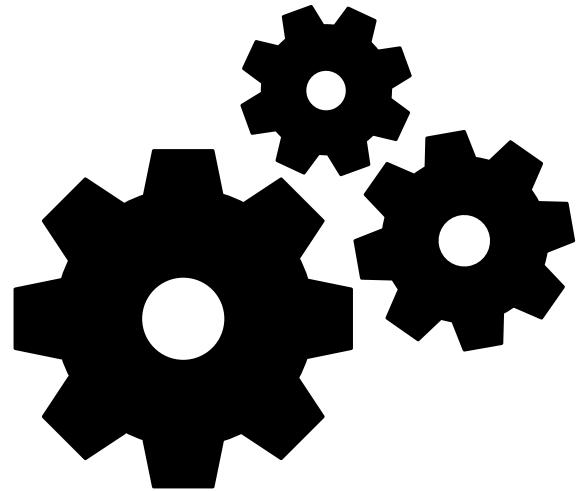
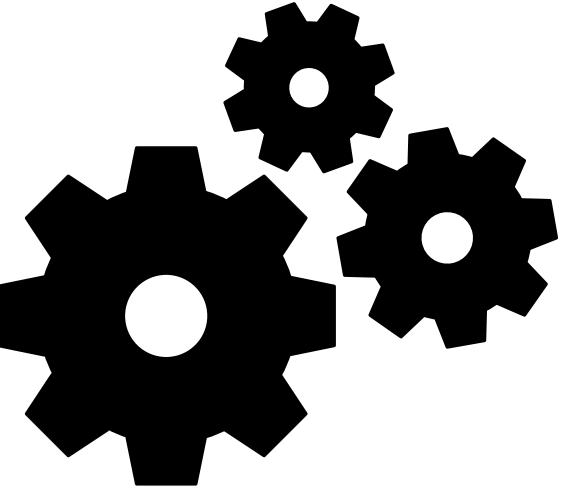
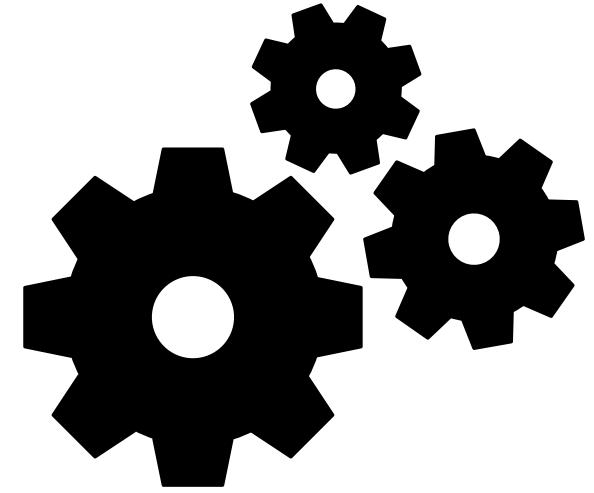
Michael Cook, Simon Colton, and Jeremy Gow
<http://ccg.doc.gold.ac.uk>

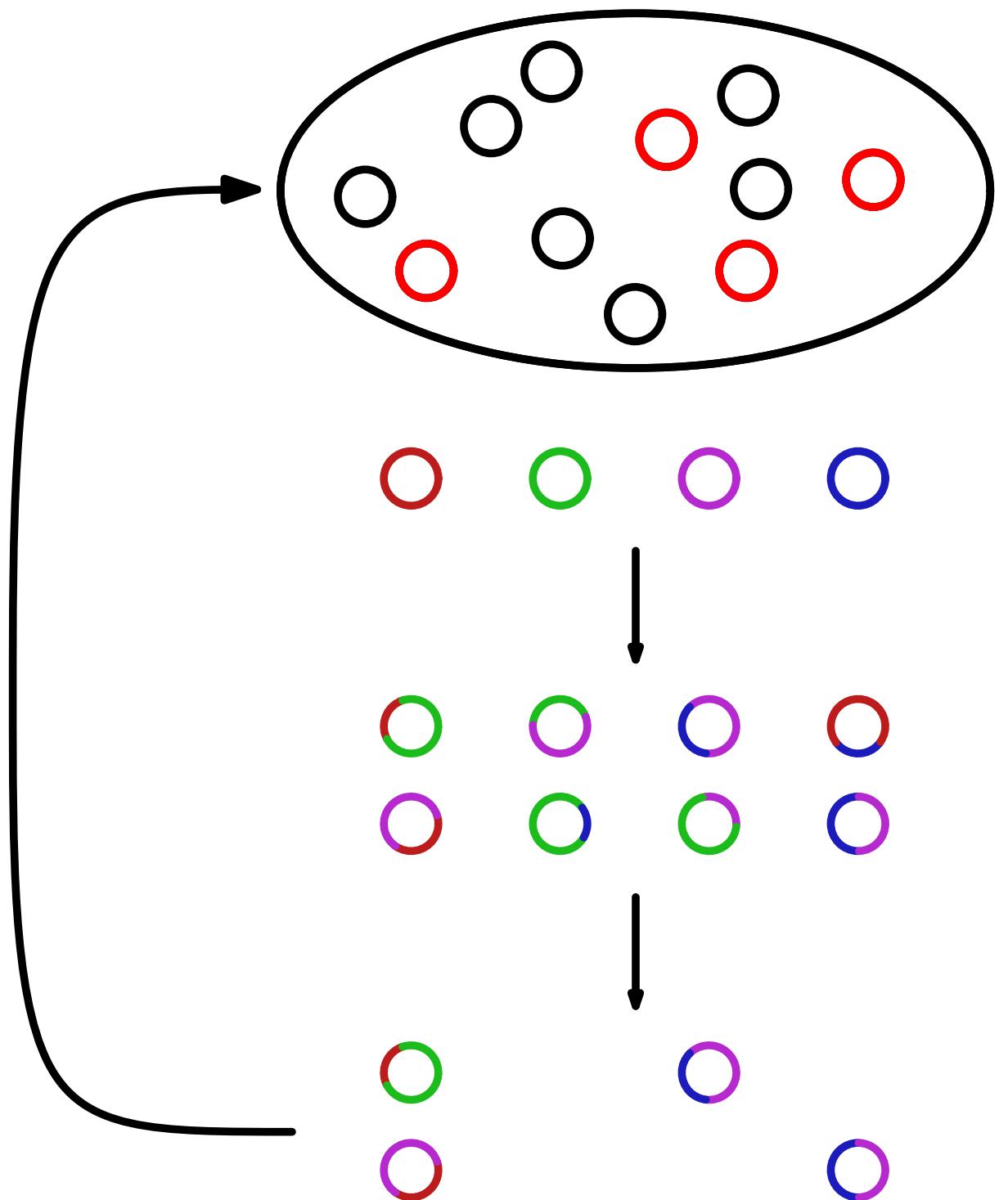
The ANGELINA Videogame Design System, Part II

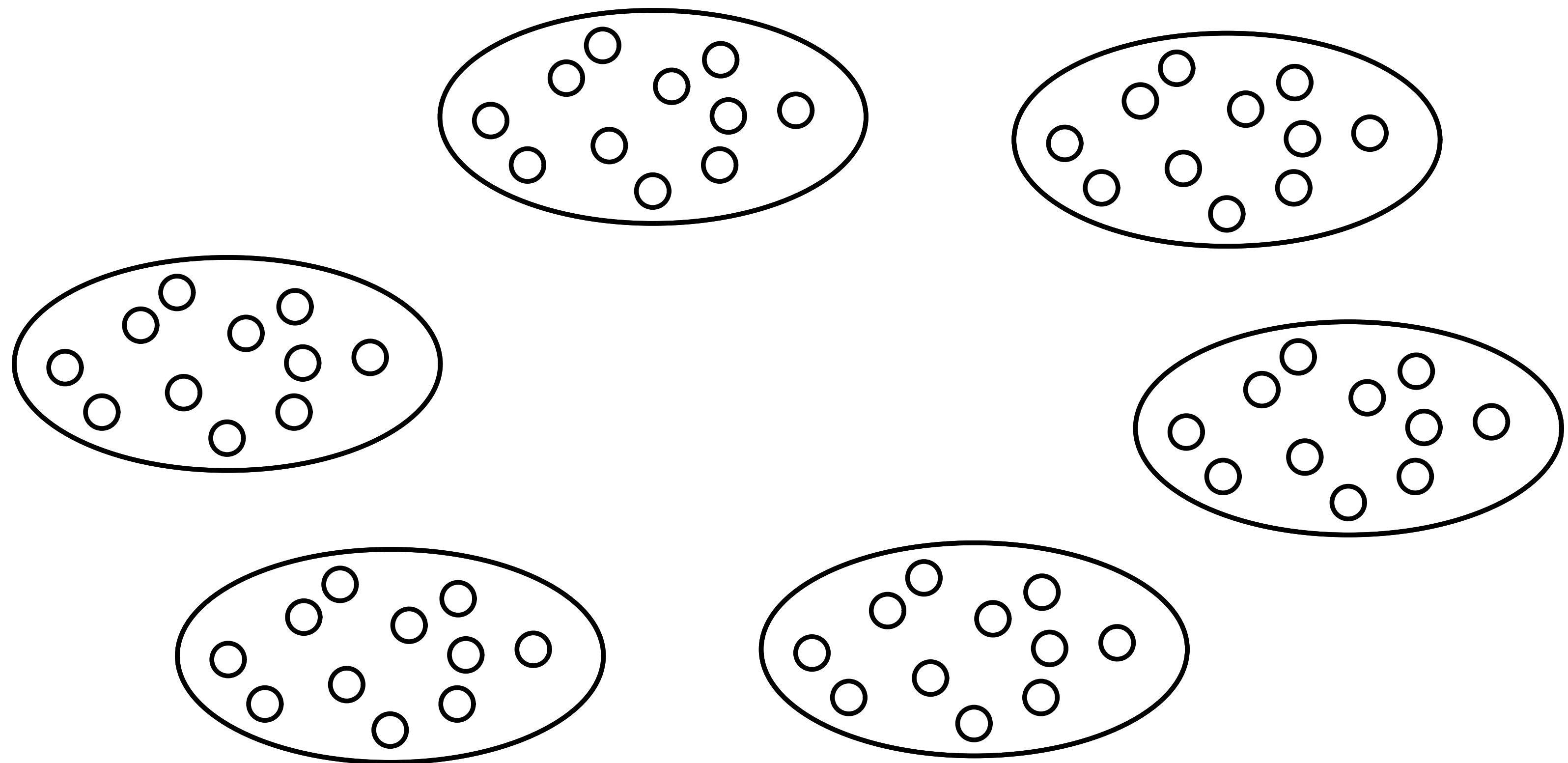
Michael Cook, Simon Colton, and Jeremy Gow
ccg.doc.gold.ac.uk



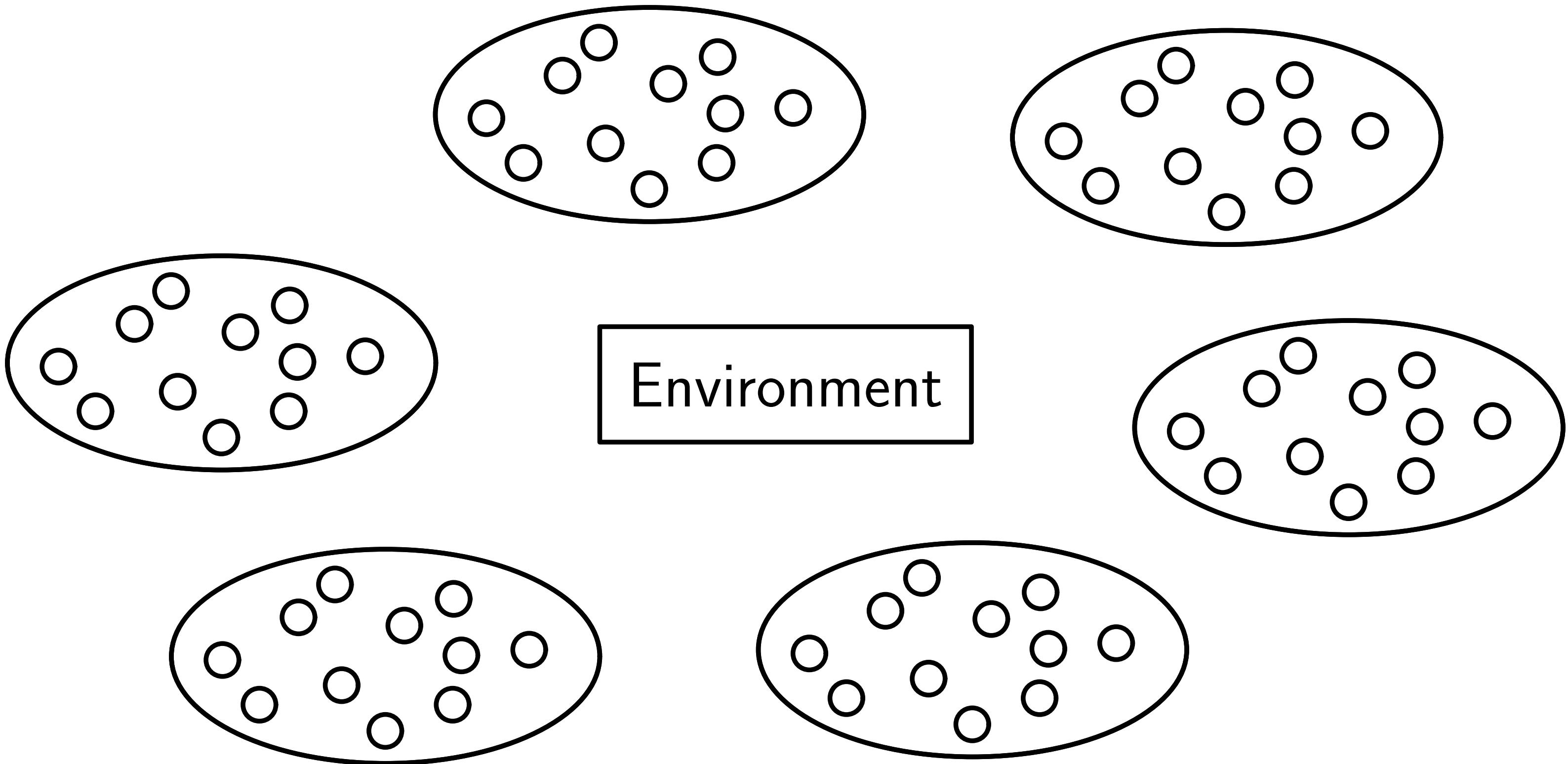


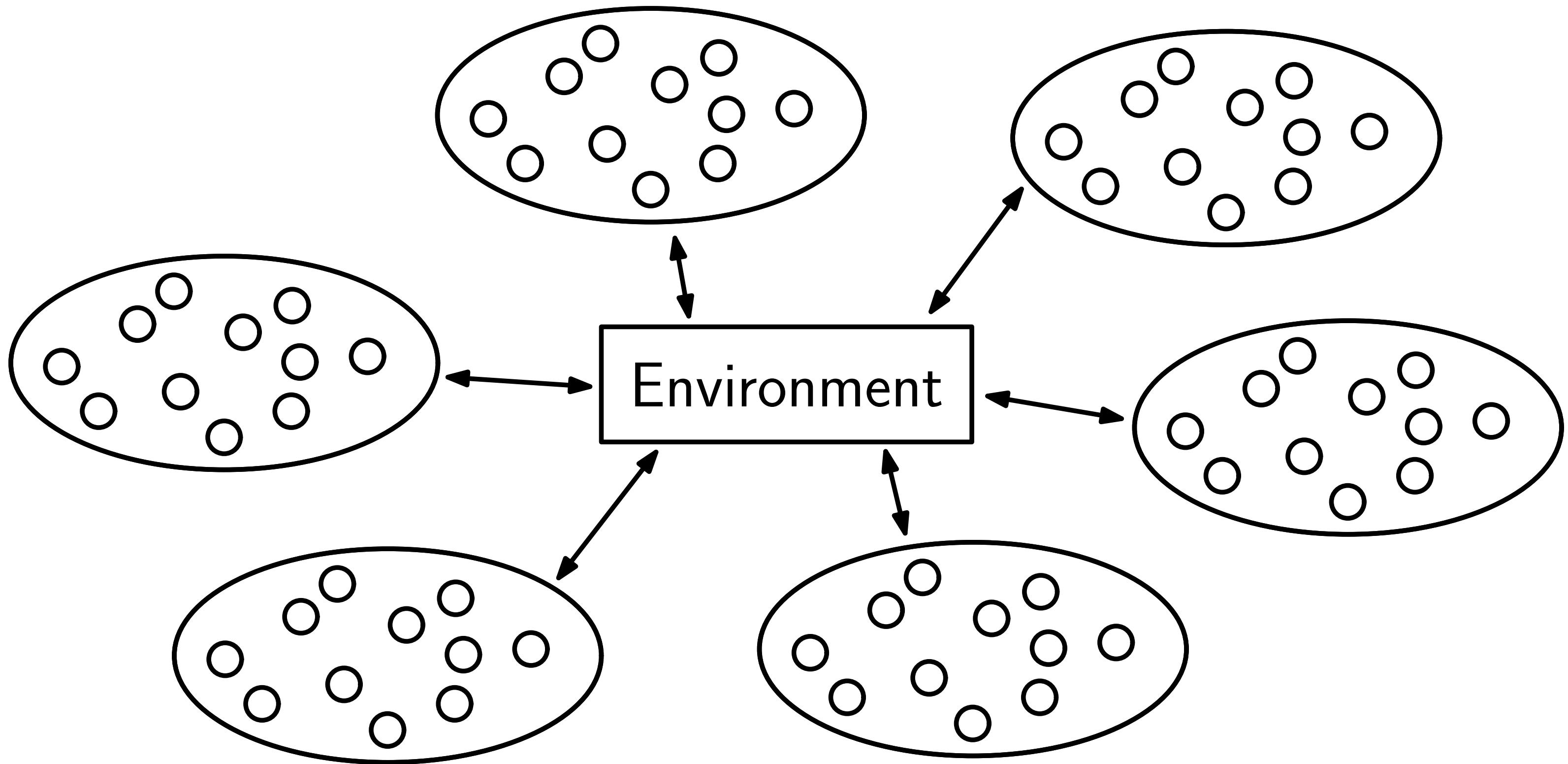


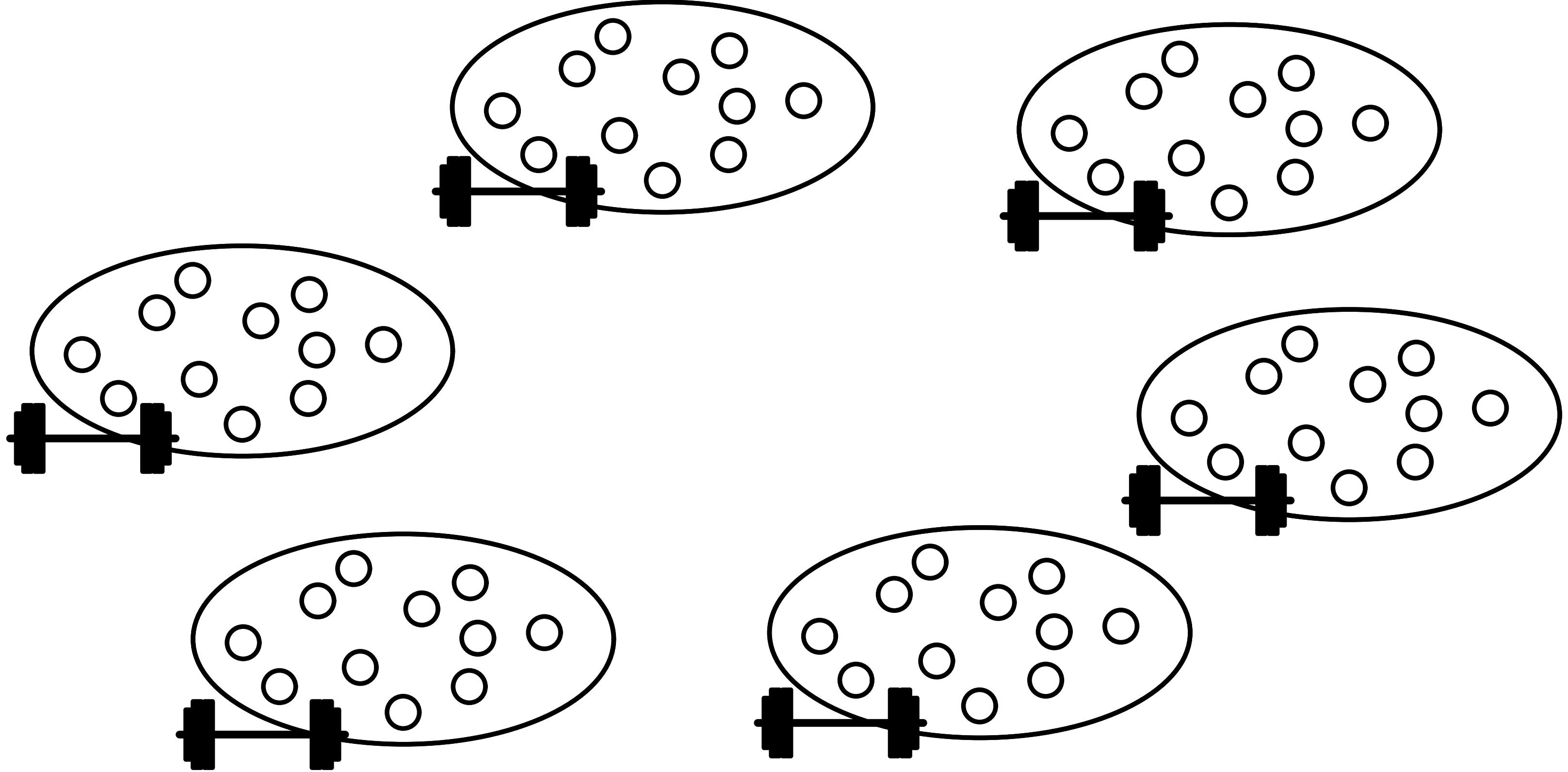


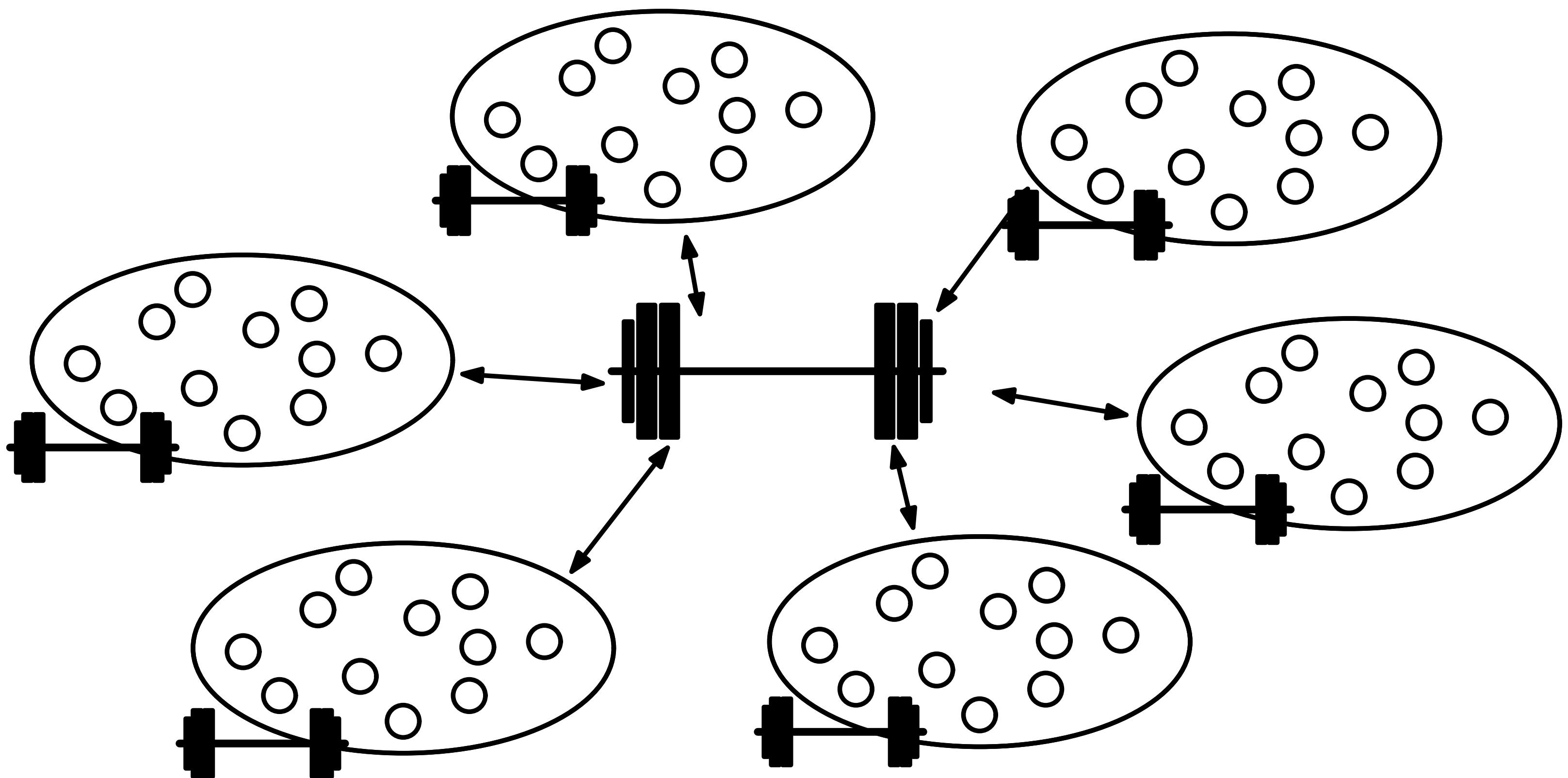


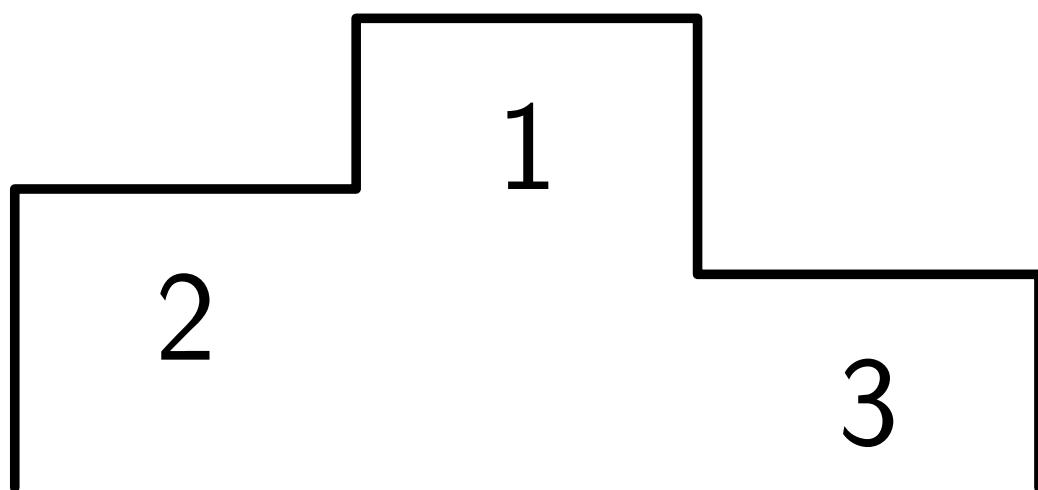
Environment

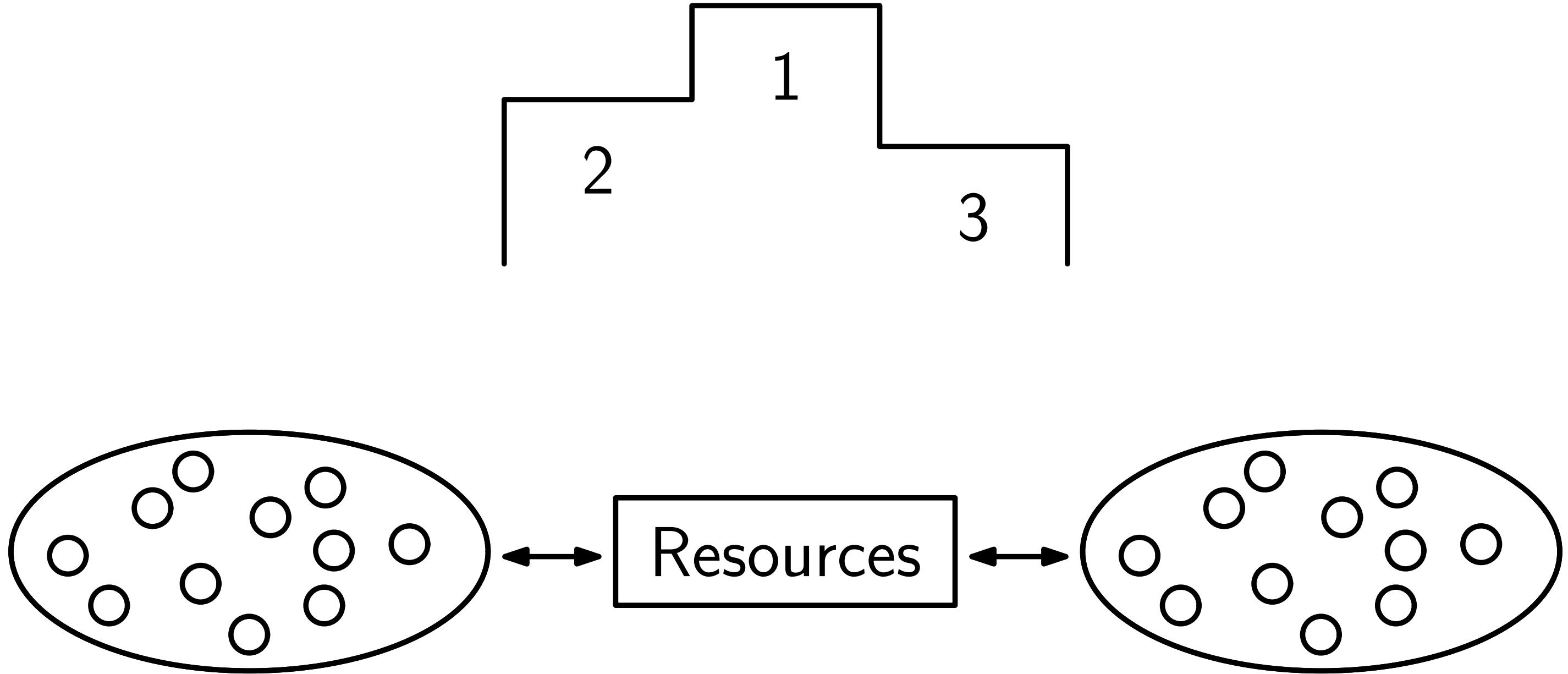






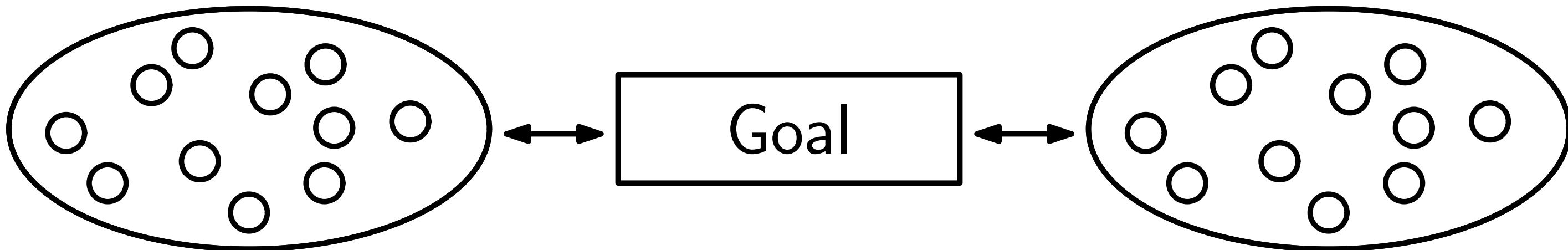




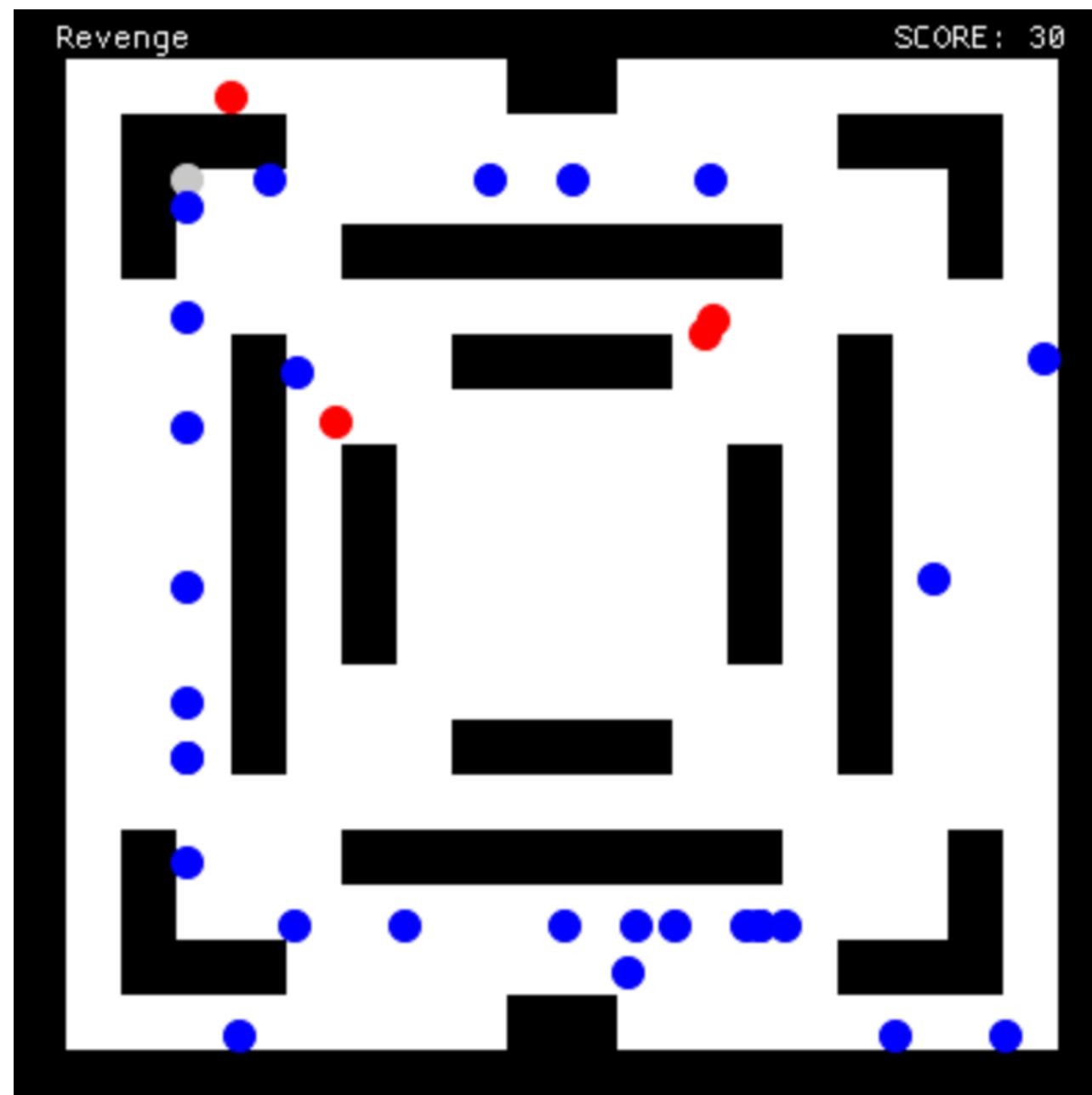




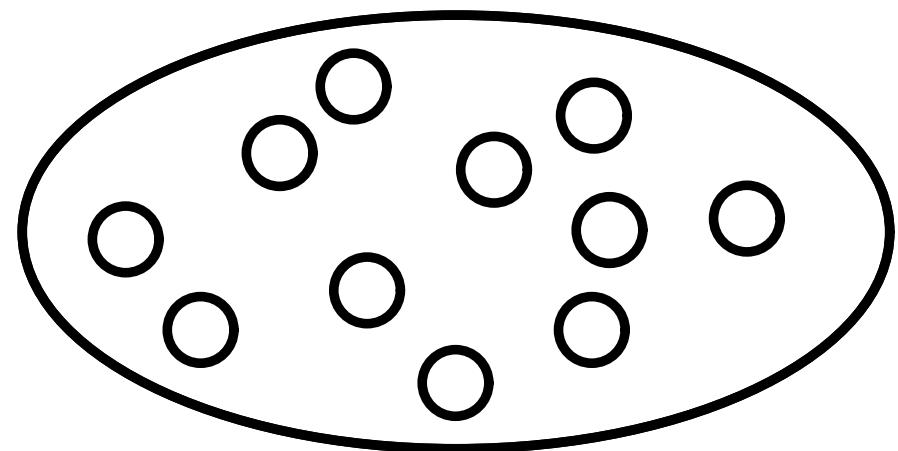
antarctica.gov.au



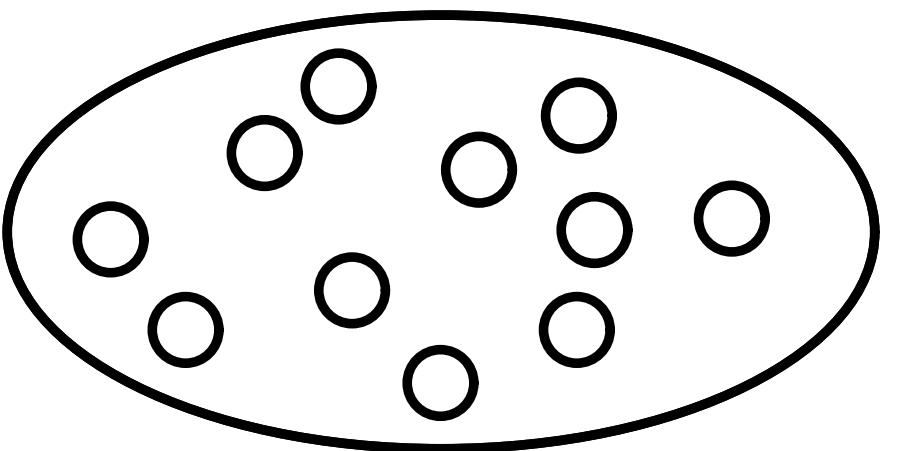
ANGELINA₁



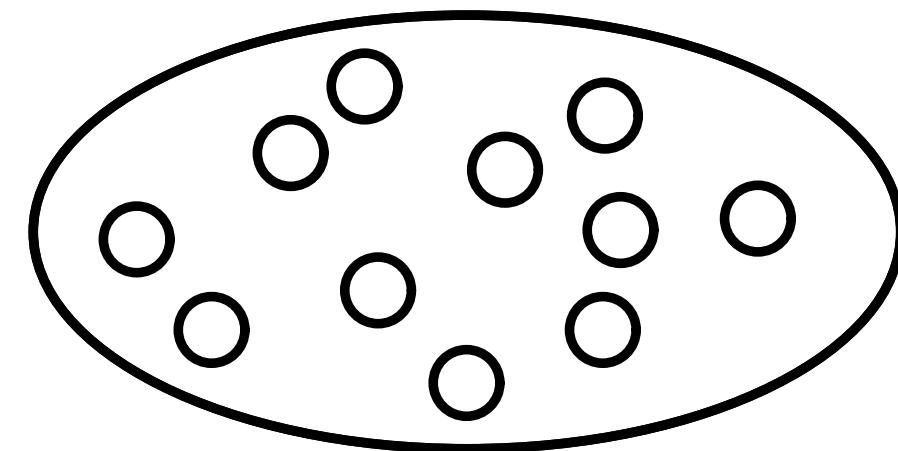
Rules



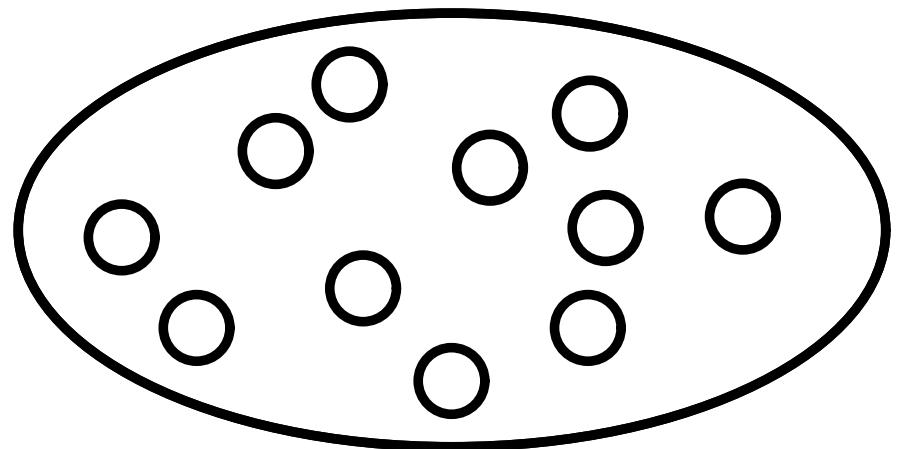
Levels



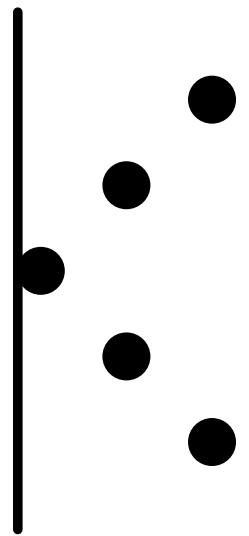
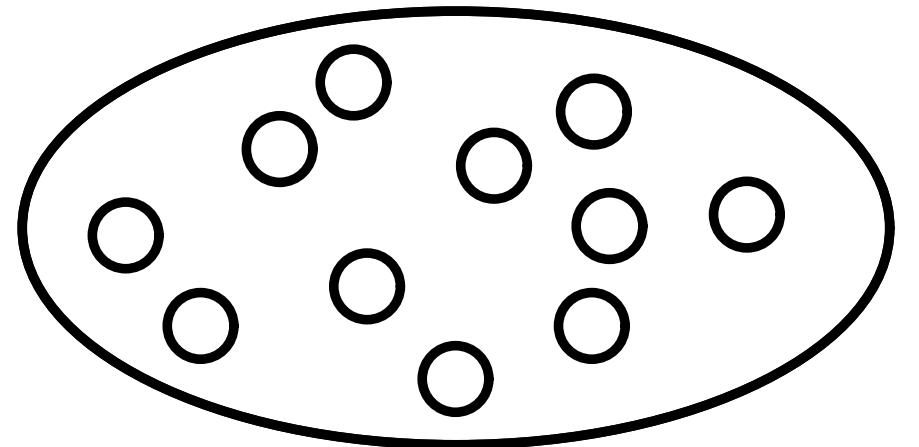
Layouts



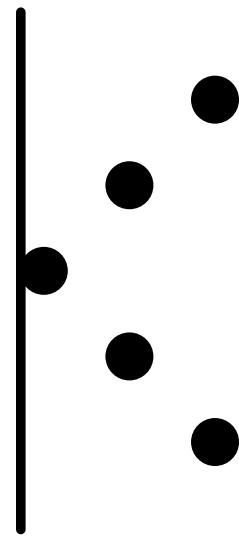
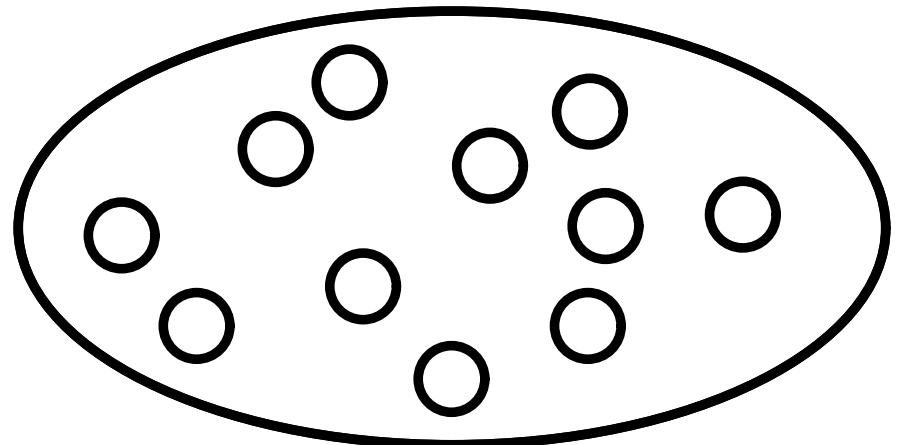
Rules



Rules

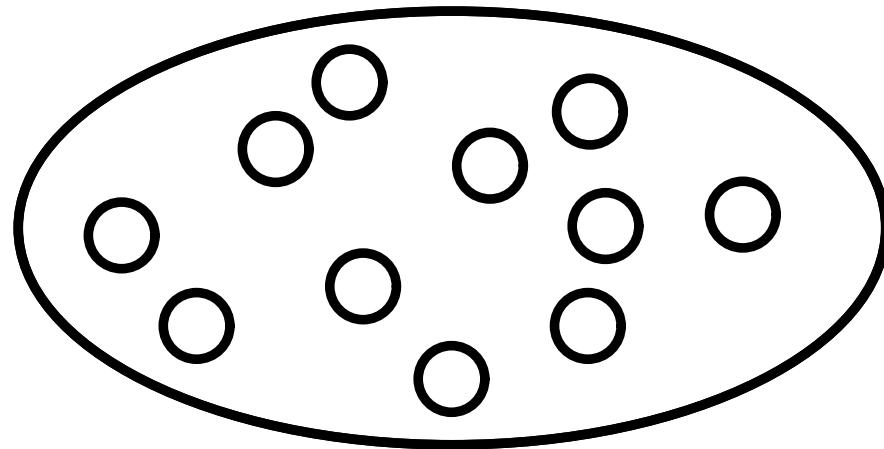


Rules

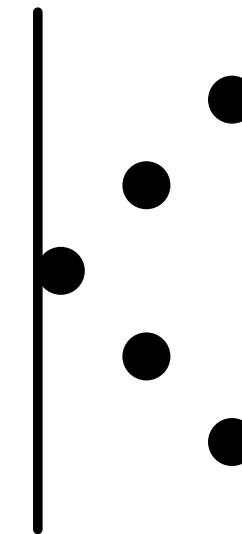


$A_1 + A_2 : X_1 + X_2 + \text{Score}$

Rules



If player collides with a dot, you get score/lose life
If player collides with a wall, you can't move forward



Entity 1
in the system

Entity 2
in the system

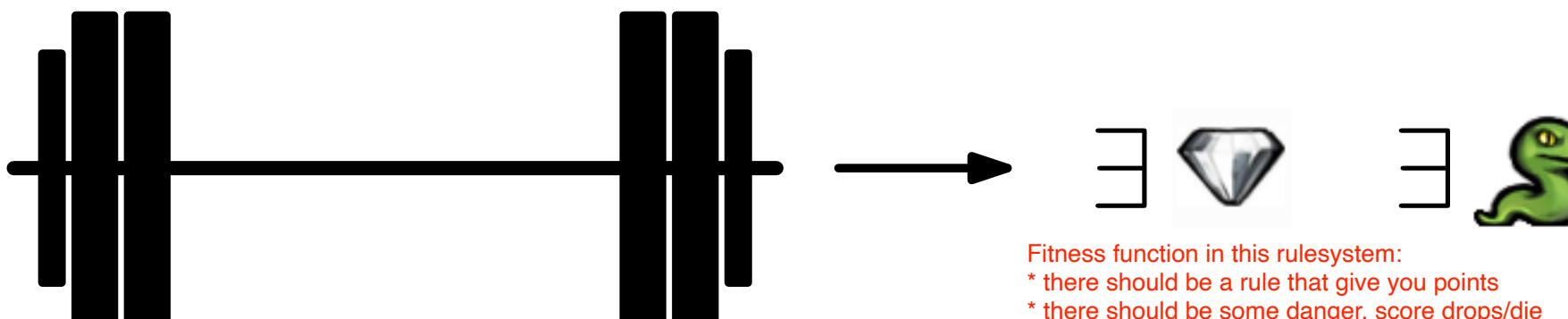
Effect X go to
entity 1

Effect X go to
entity 2

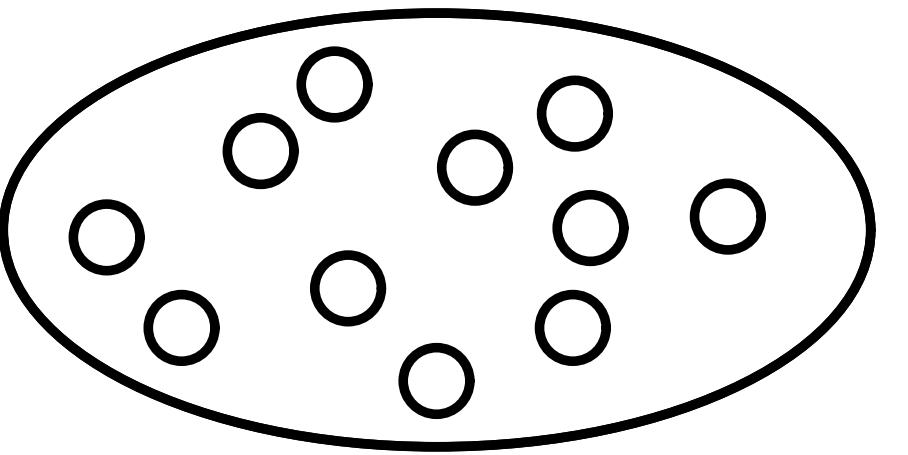
Is there any change
in the score for this collision?

$$A_1 + A_2 : X_1 + X_2 + \text{Score}$$

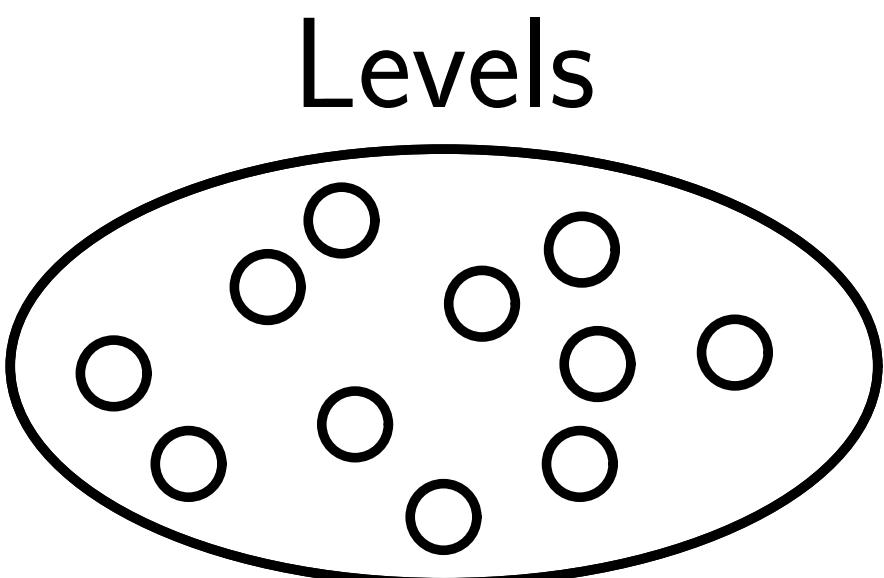
We apply this rule, when collision



Levels

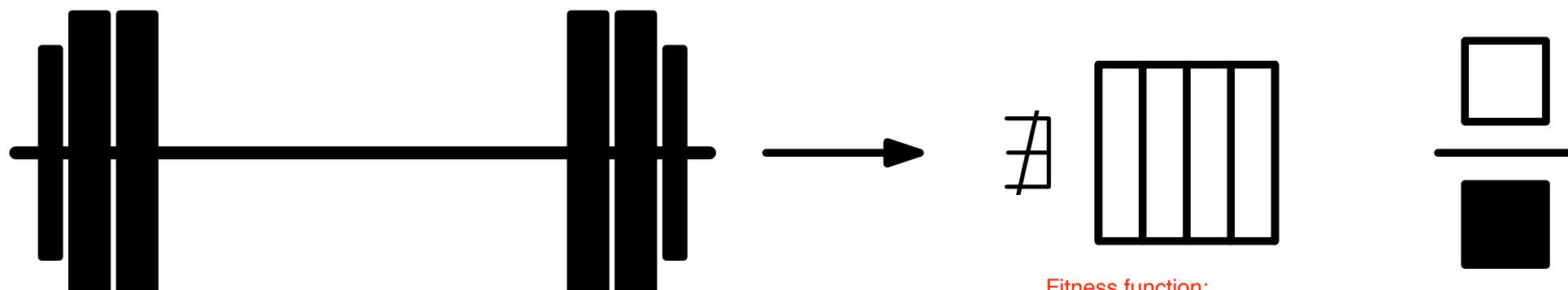
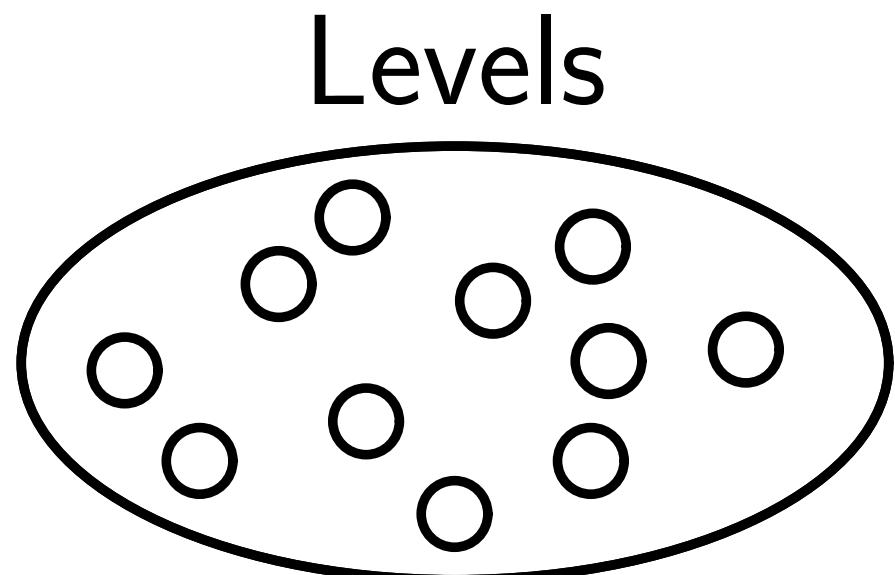


0	1	1	0
1	1	1	0
0	0	0	0
1	1	1	0



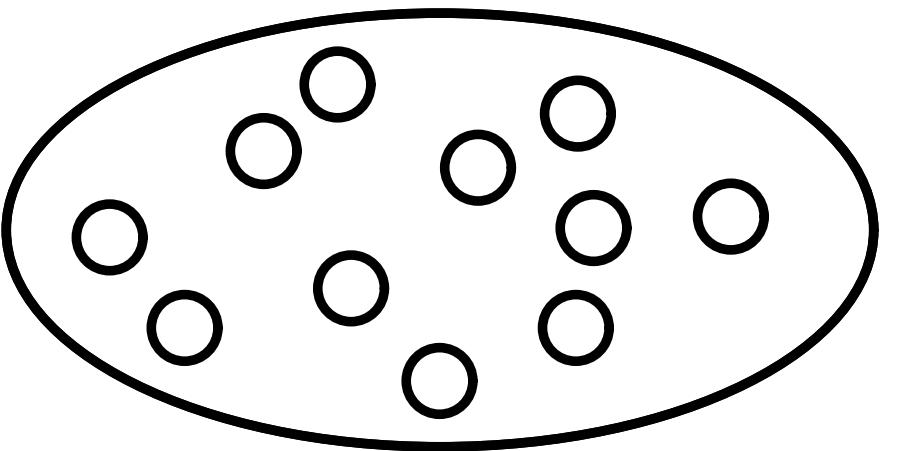
Binary array that tells if there is a hole in the wall or not
1 - equals room
0 - equals wall

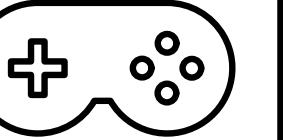
0	1	1	0
1	1	1	0
0	0	0	0
1	1	1	0



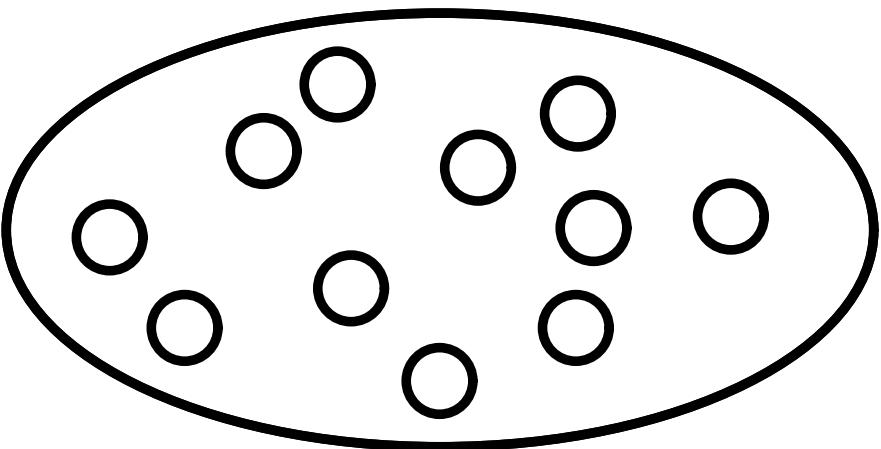
Fitness function:
* there should be no boarded off areas
* there should be a ratio between wall tiles and room tiles

Layouts



●	$\langle x, y \rangle$	
●	$\langle x, y \rangle$	
●	$\langle x, y \rangle$	—
●	$\langle x, y \rangle$	
●	$\langle x, y \rangle$	
●	$\langle x, y \rangle$	C

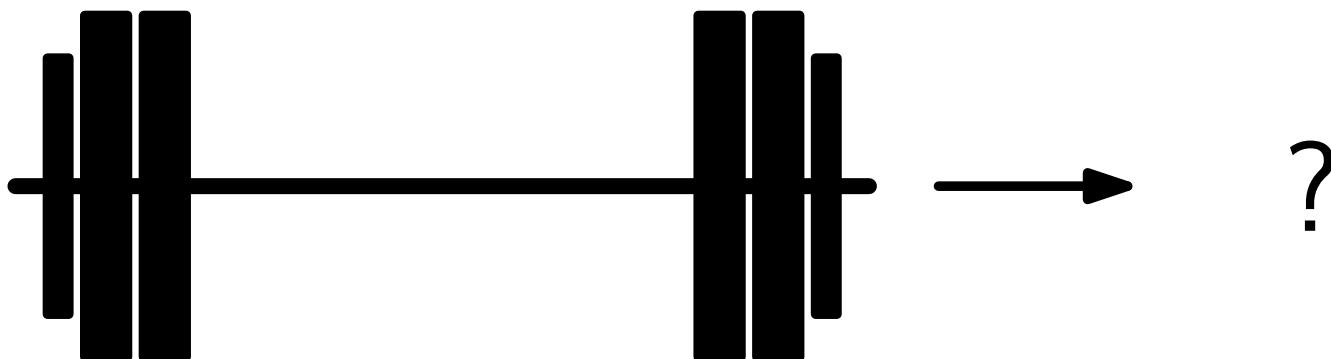
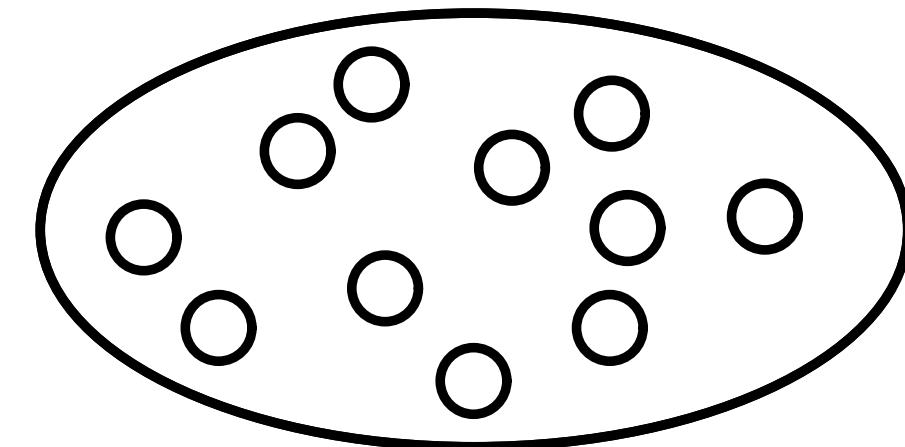
Layouts



	$\langle x, y \rangle$	
	$\langle x, y \rangle$	
	$\langle x, y \rangle$	—
	$\langle x, y \rangle$	
	$\langle x, y \rangle$	
	$\langle x, y \rangle$	C

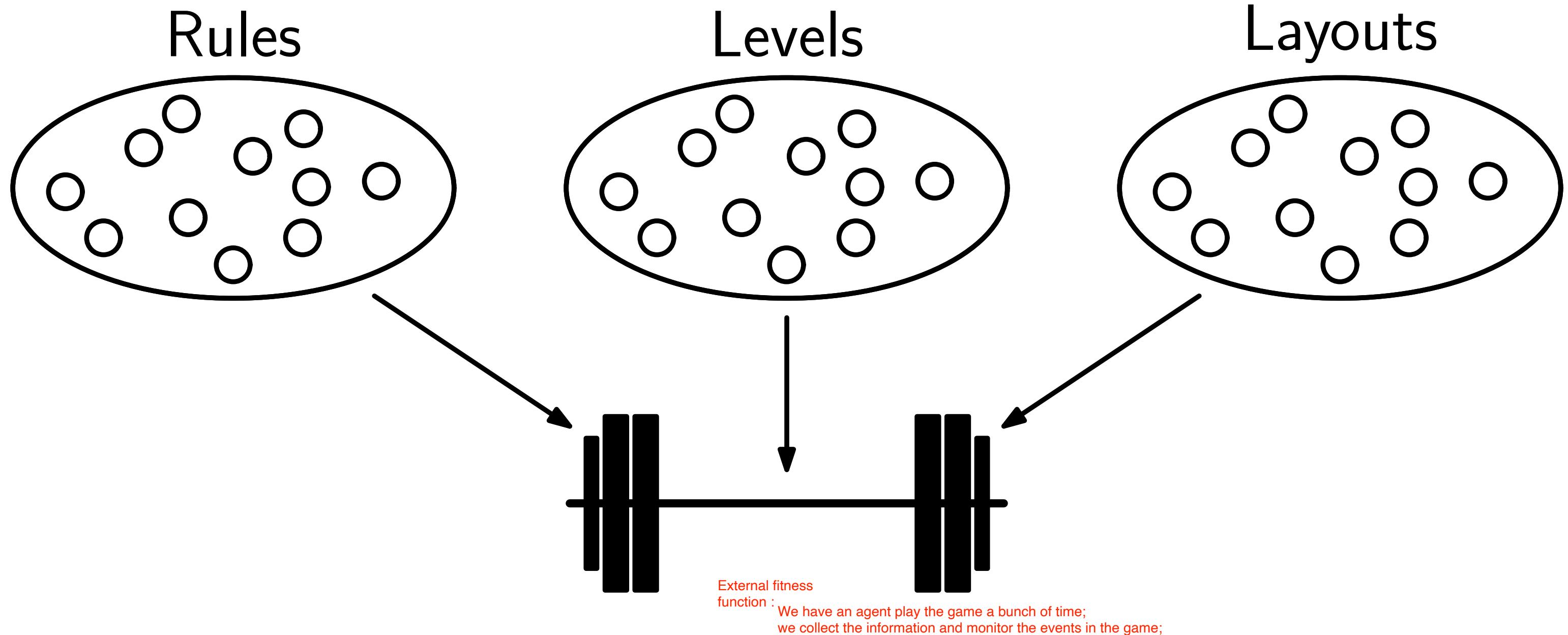
List of entities that exist in our game.
 They are represented by some asset (colored dots here).
 For each of these, there are coordinates (x,y) and
 a hard-coded control scheme (player controlled, random move, not move, etc).

Layouts



Assignment 3

Name two things we should include in the Layout fitness function of ANGELINA₁.









—





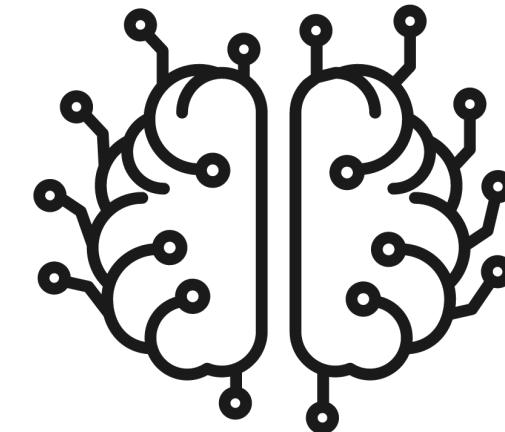
We have three agents.



Does nothing.



Random actions.



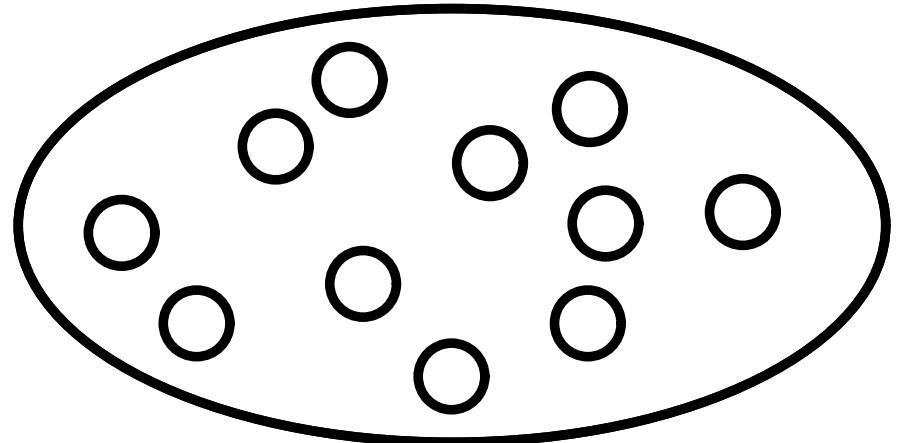
Knows the rules of the game,
tries to go towards things that have a positive effect on the score
and stay away from things that have a negative effect on the score.

ANGELINA₂

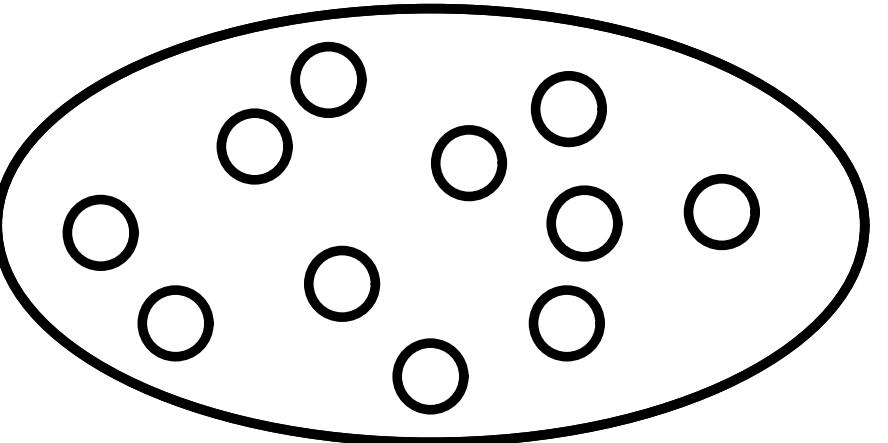


Powerups are added compared to Angelina1.
You can't exit without the powerups

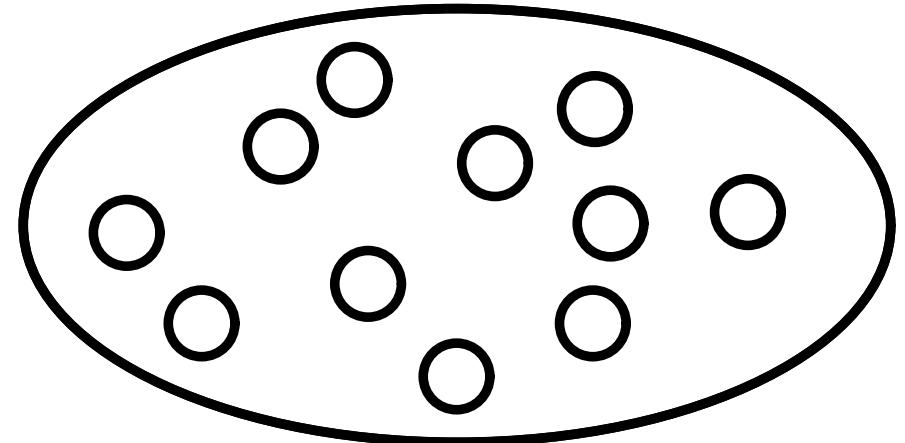
Powerups



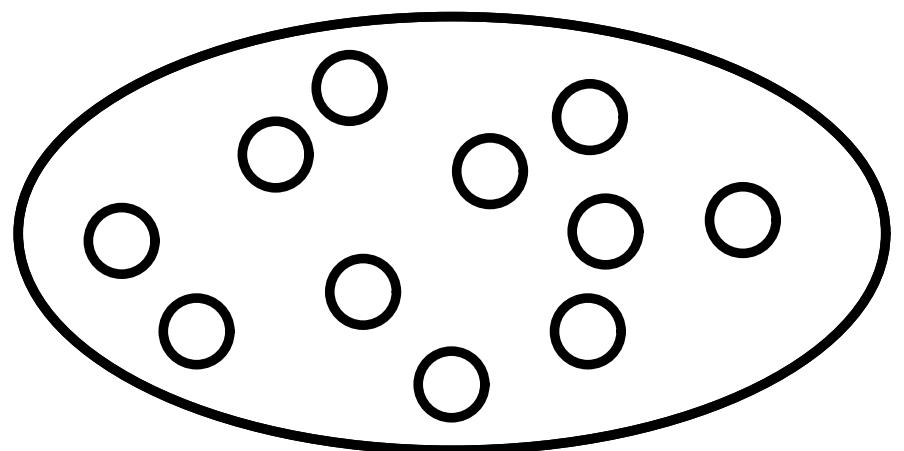
Levels



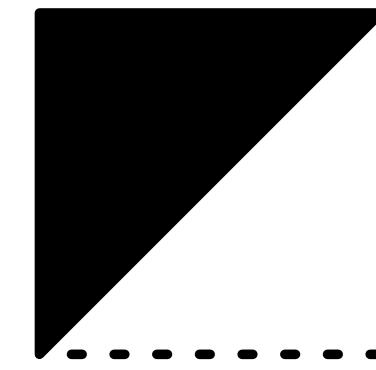
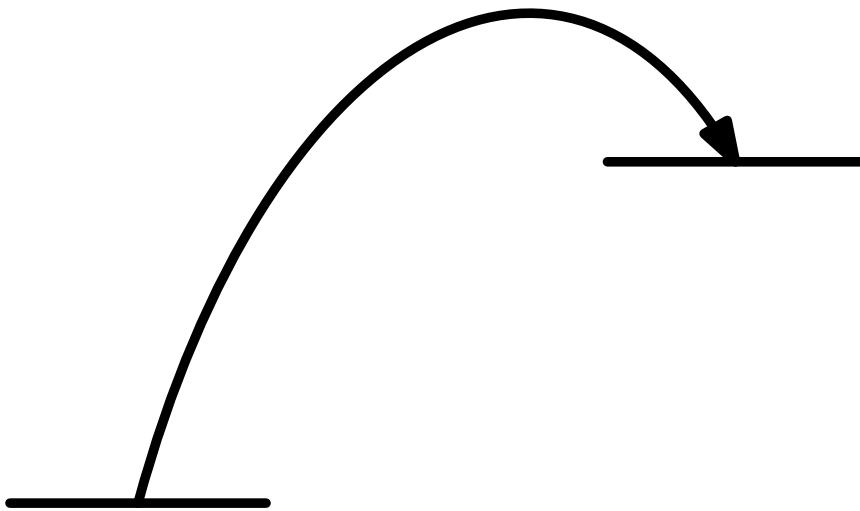
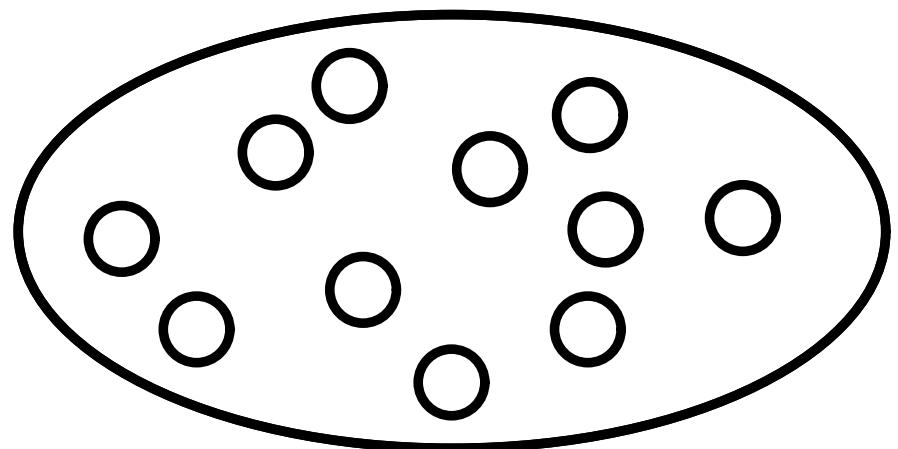
Layouts



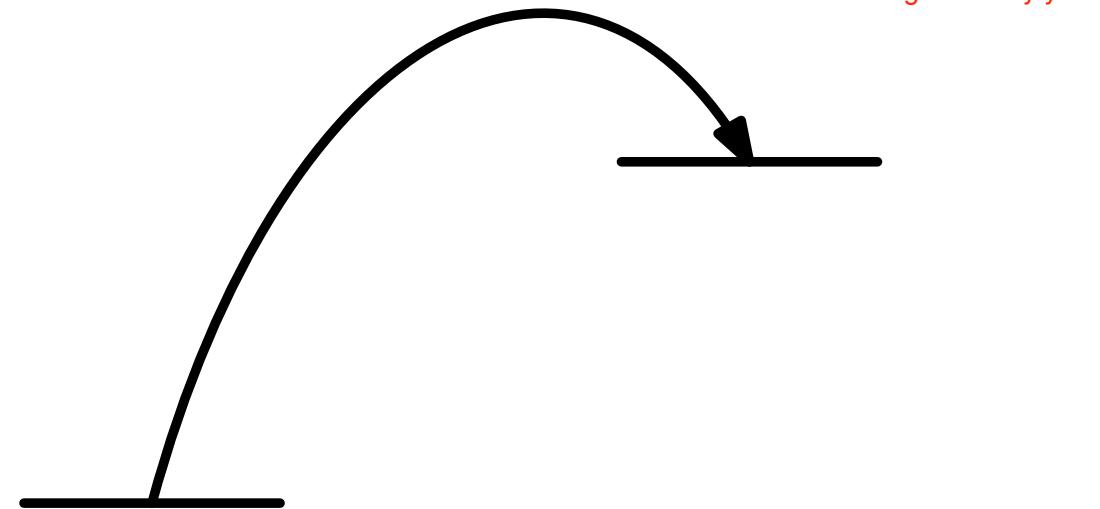
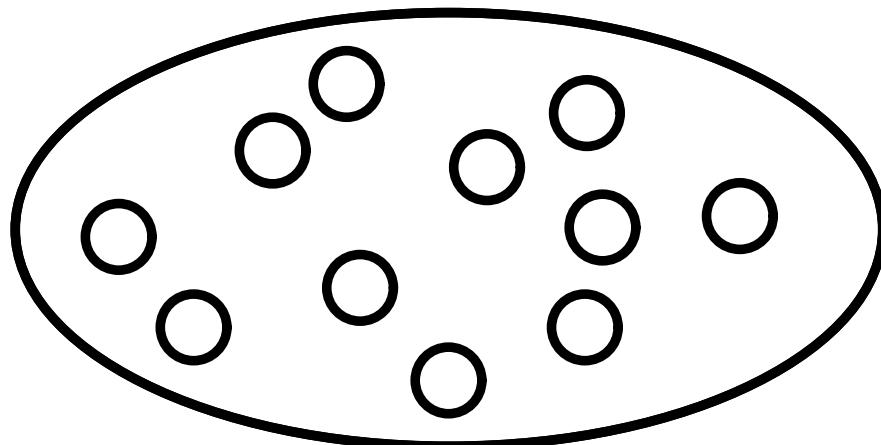
Powerups



Powerups

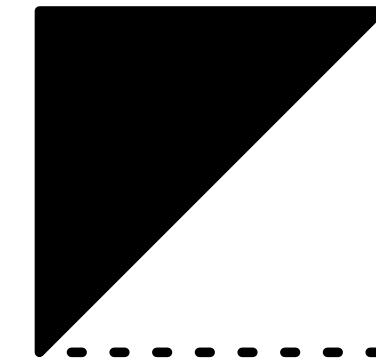


Powerups



Three types of powerups allowed to exist:

- * change jump height;
- * change the force of gravity;
- * change the way you interact with static elements of the level.

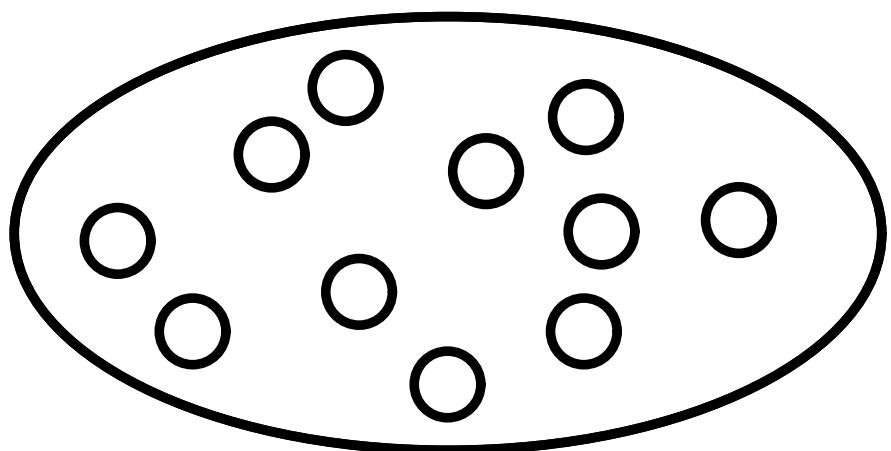


→ consistency

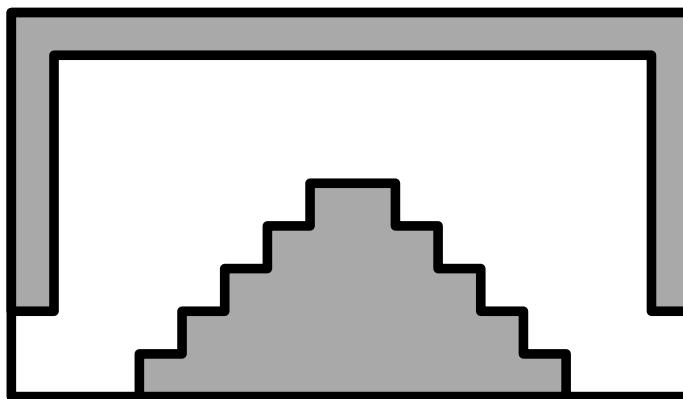
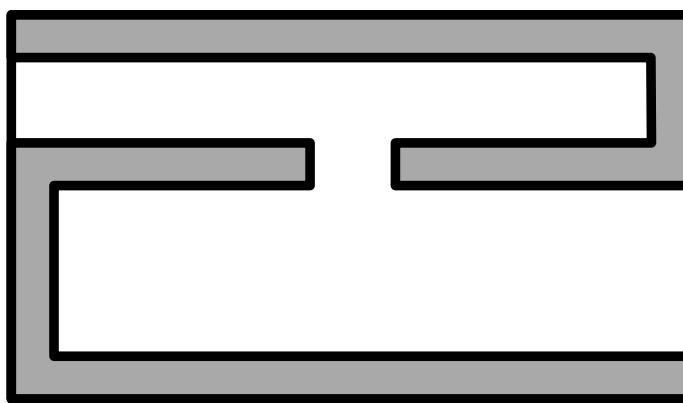
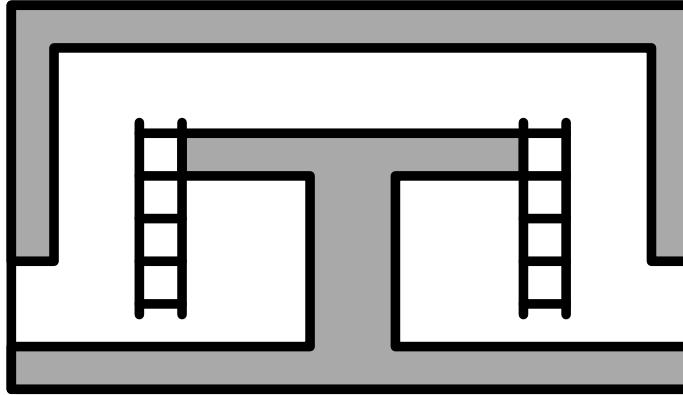
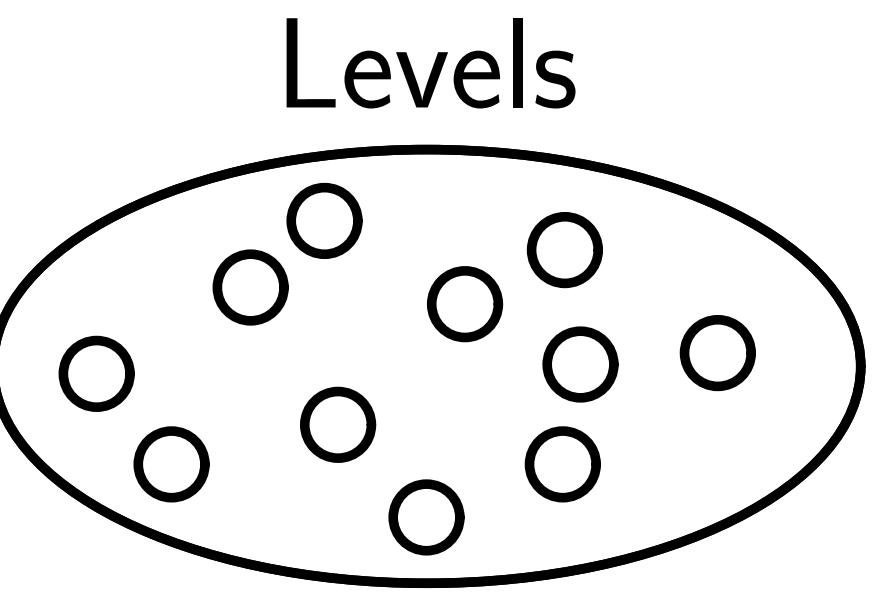
Fitness function:

- * there shouldn't be two powerups that cancel each other out!
- * there should not be two powerups that add nothing after first use or are redundant.

Levels



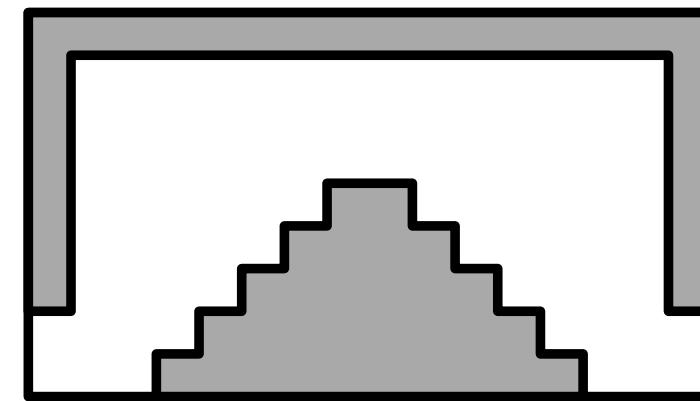
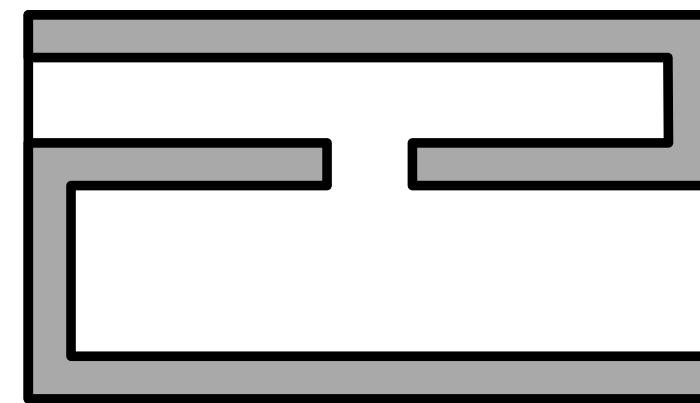
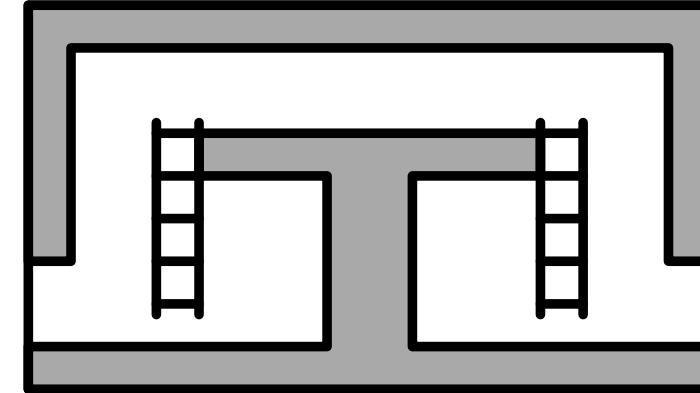
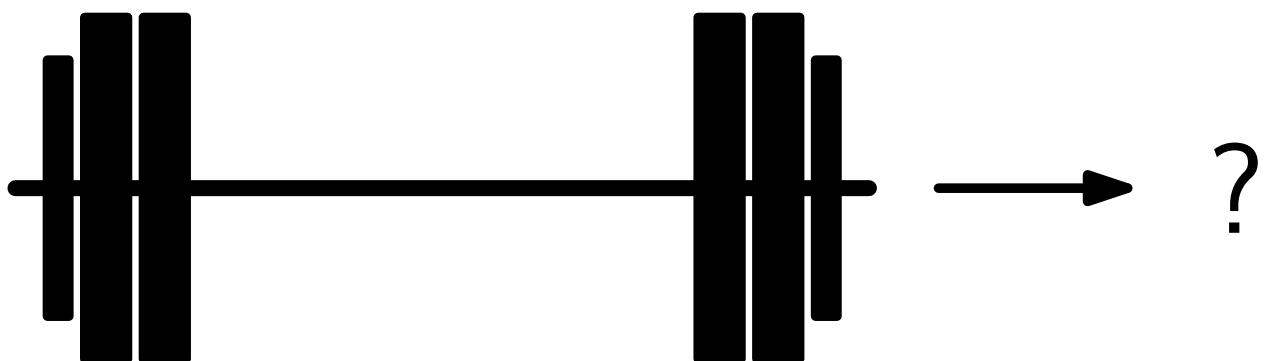
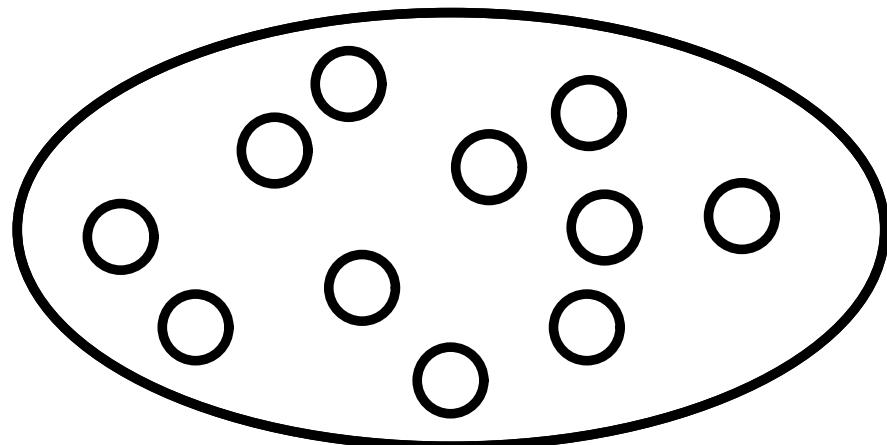
0	3	5	0
1	1	1	0
2	0	5	8
2	2	4	3



User-defined templates are placed in the rooms.

0	3	5	0
1	1	1	0
2	0	5	8
2	2	4	3

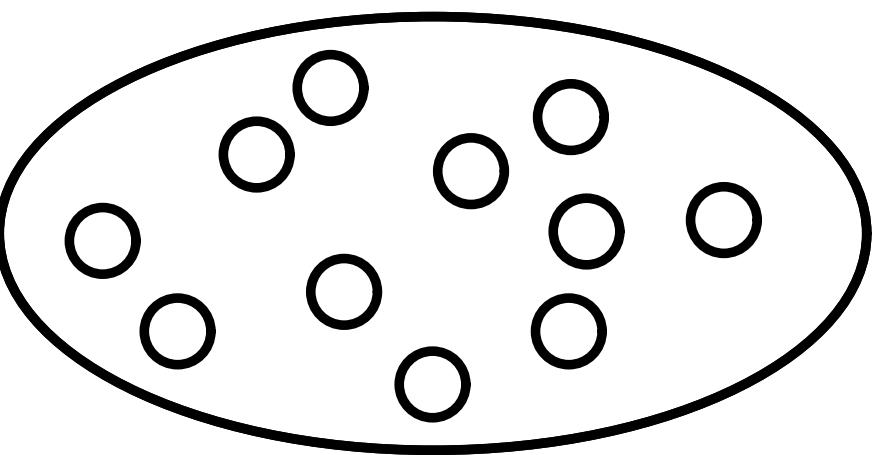
Levels

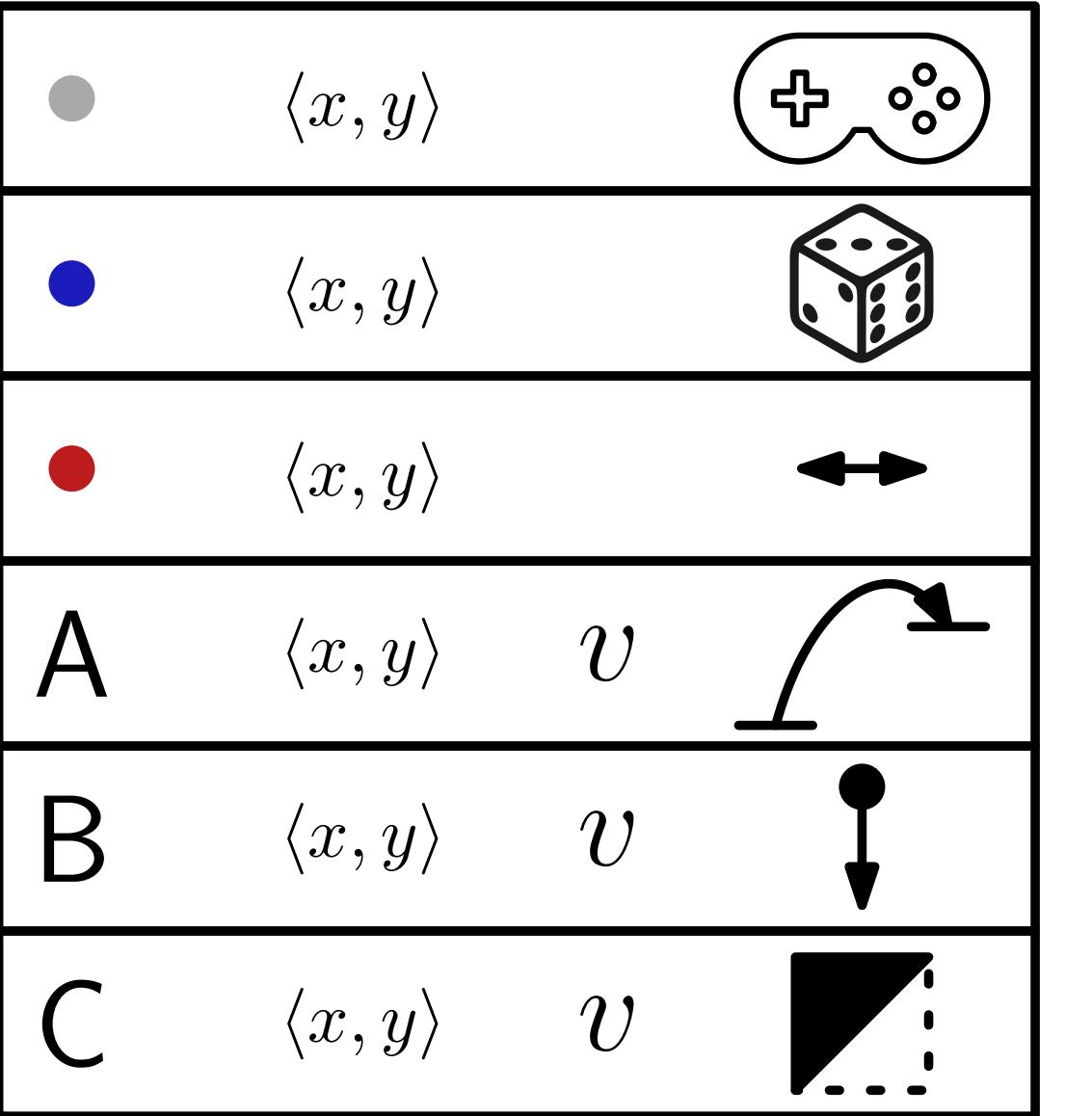


Assignment 4

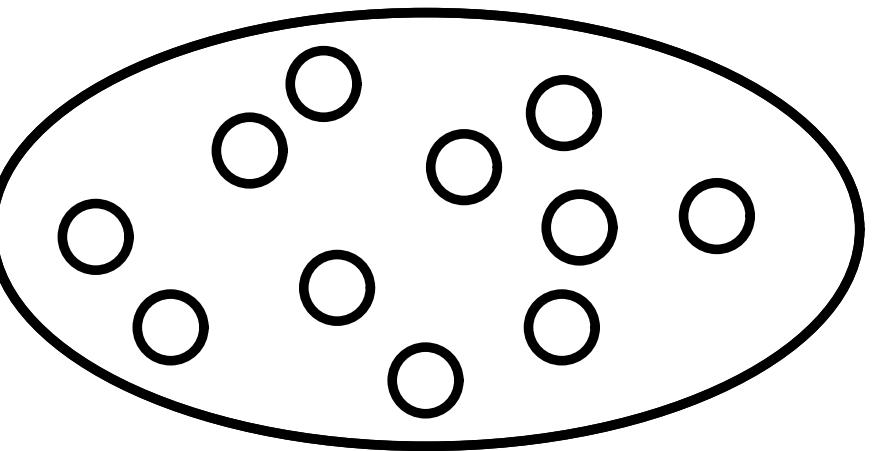
Name two things we should include in the Level fitness function of ANGELINA₂.

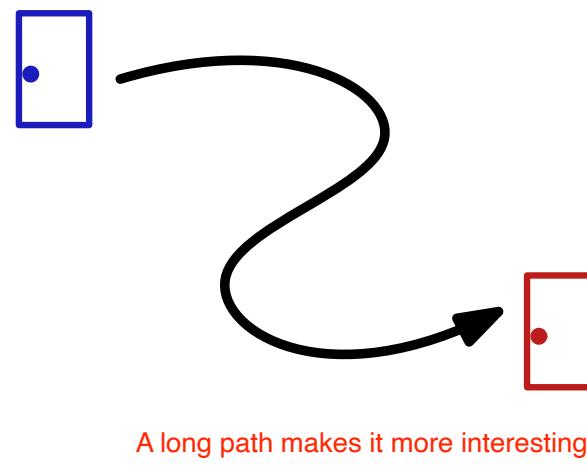
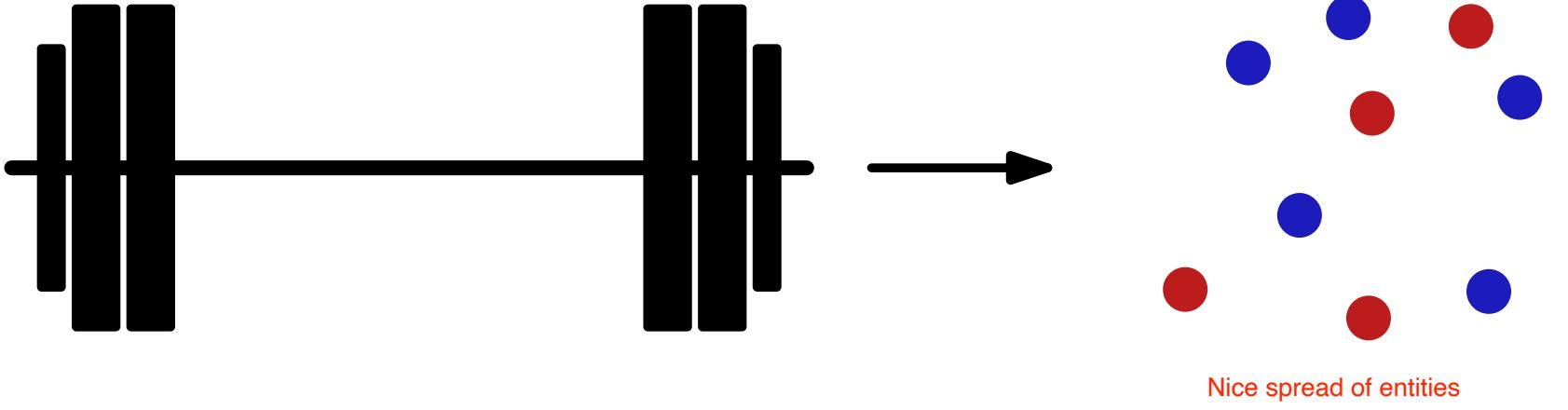
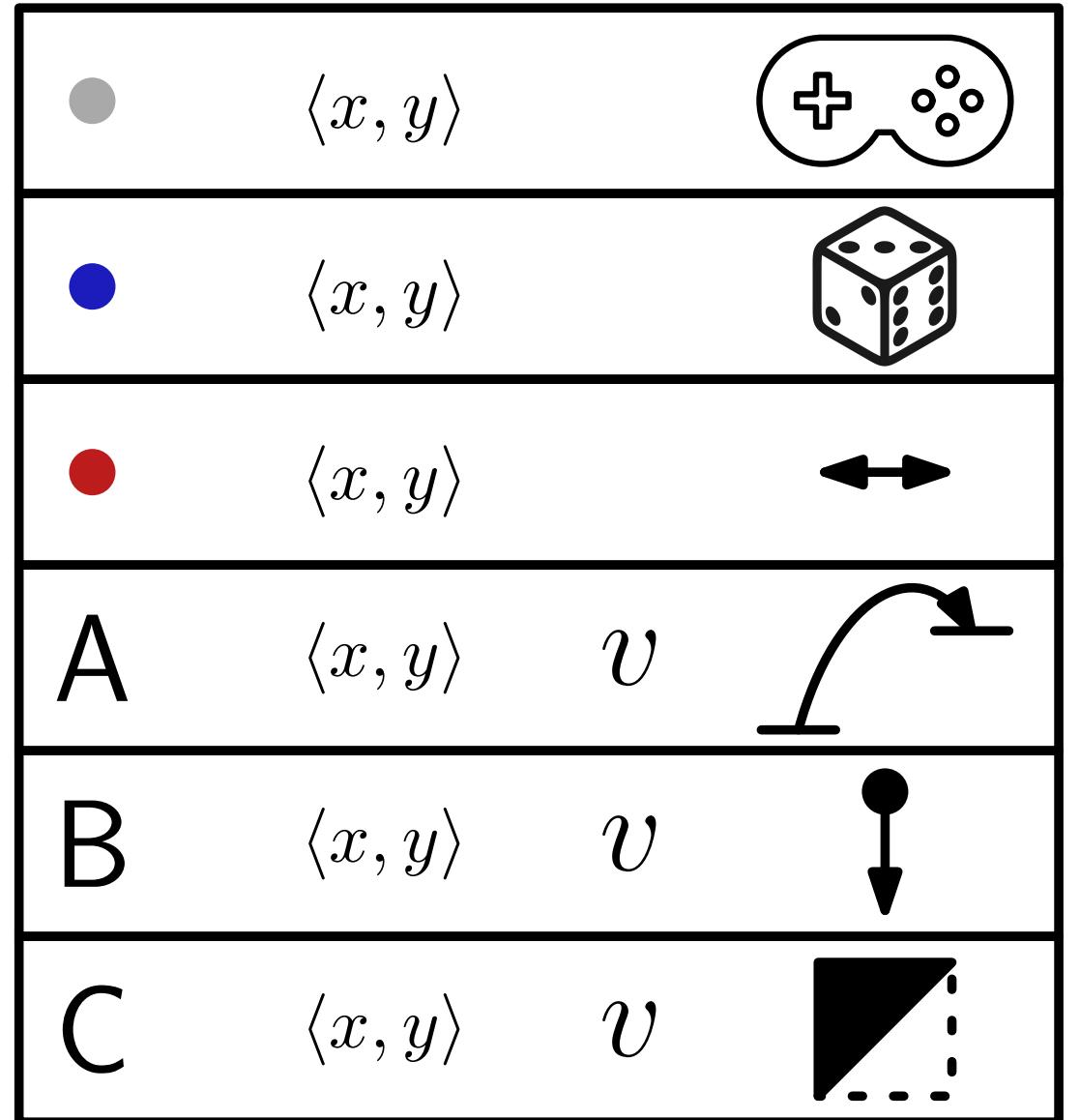
Layouts



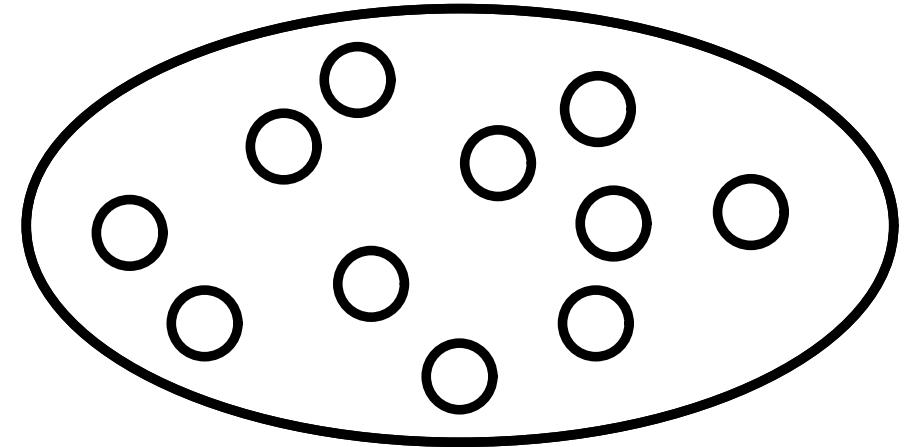


Layouts

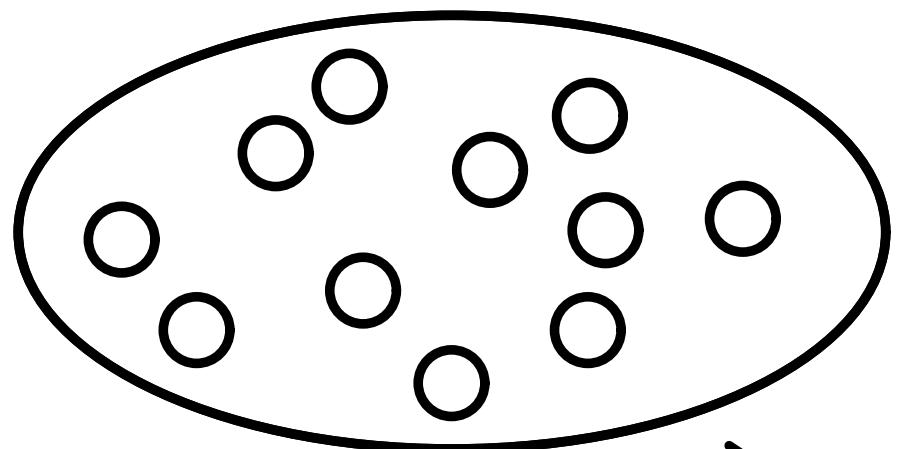




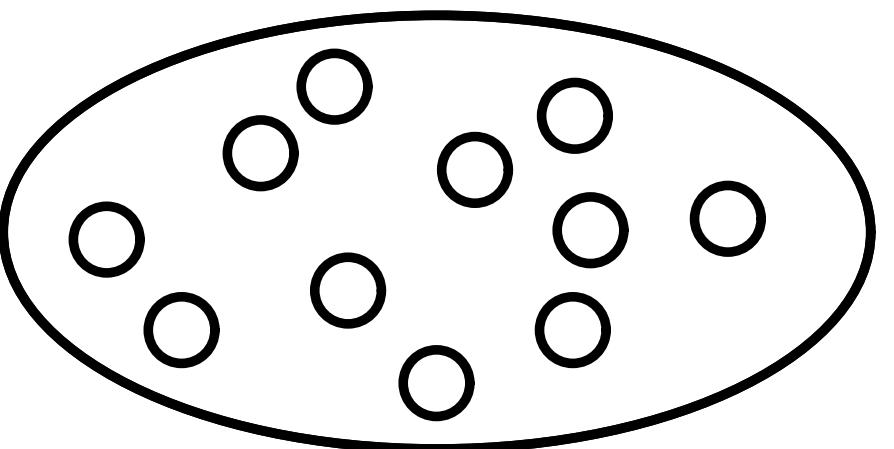
Layouts



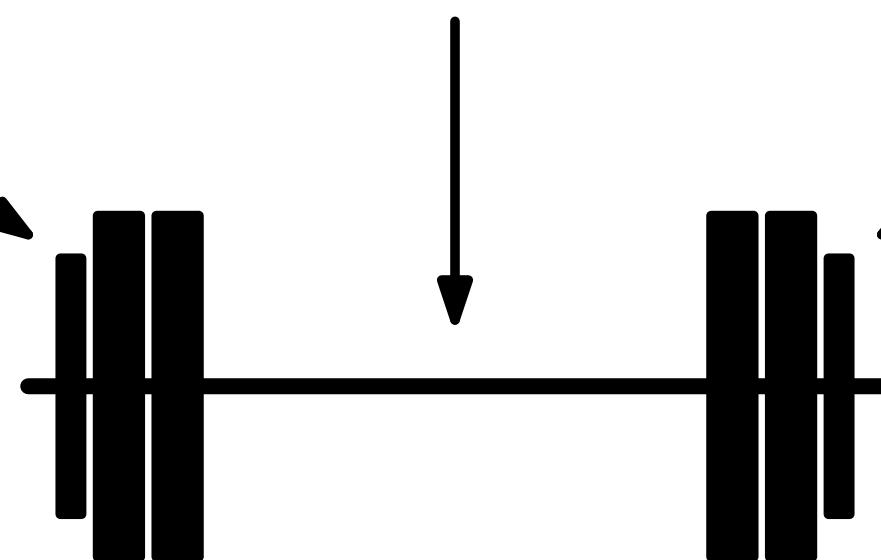
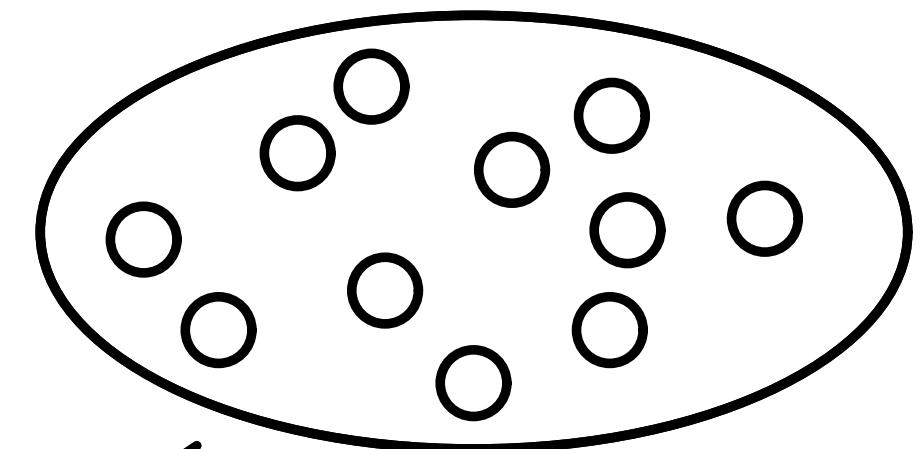
Powerups

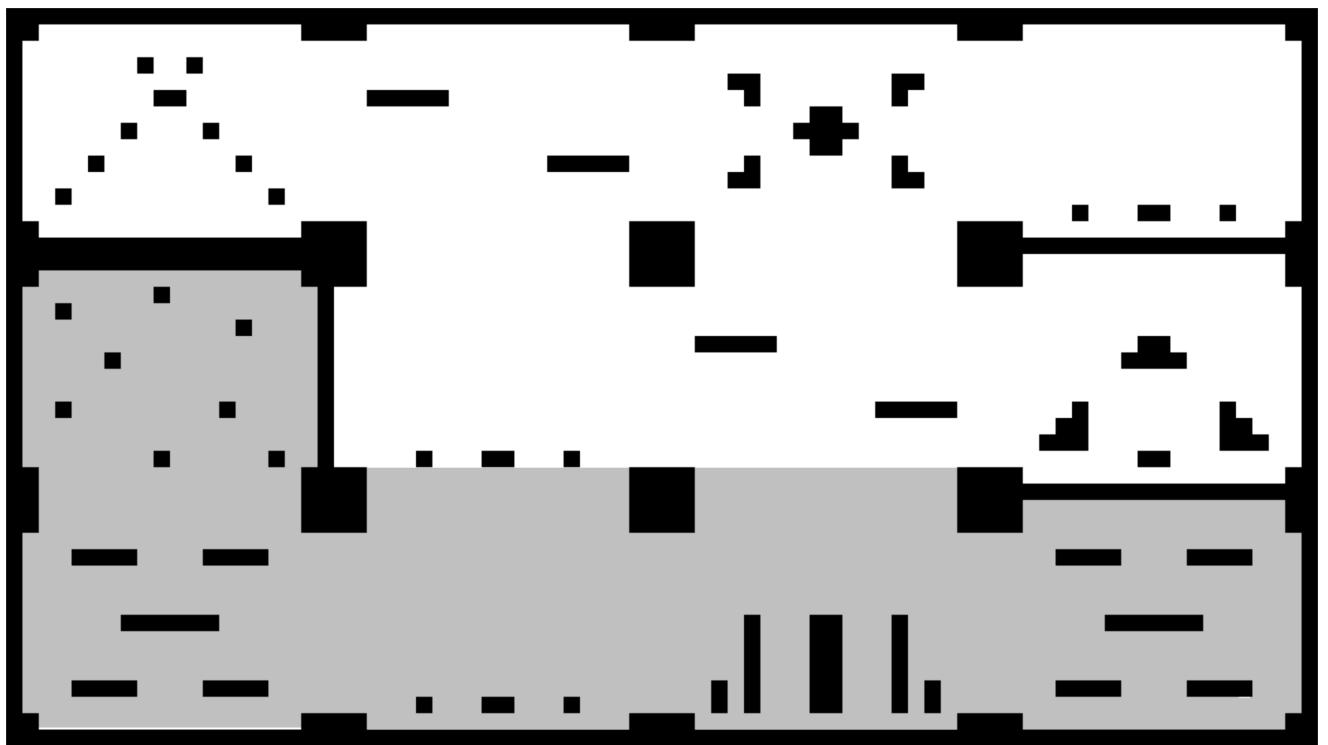
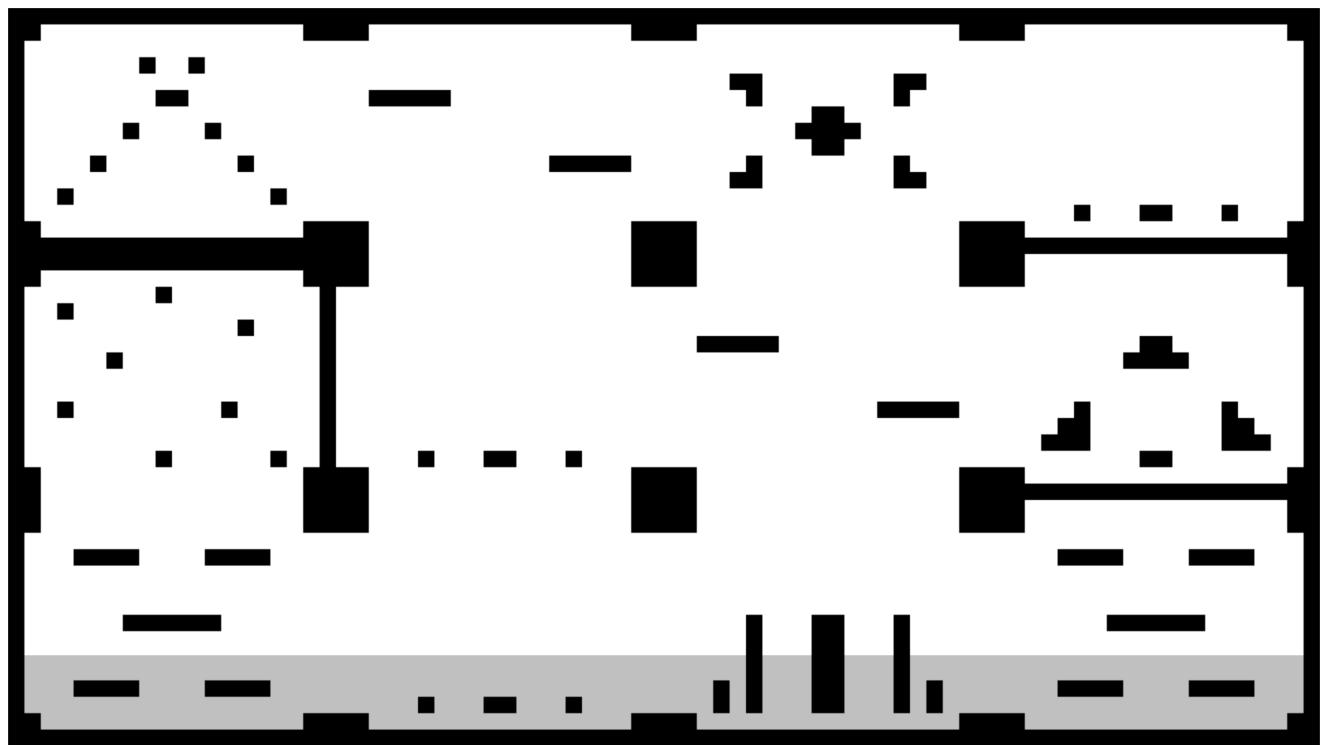


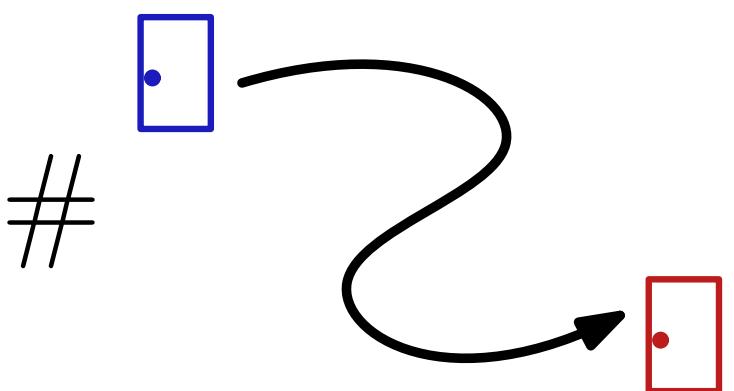
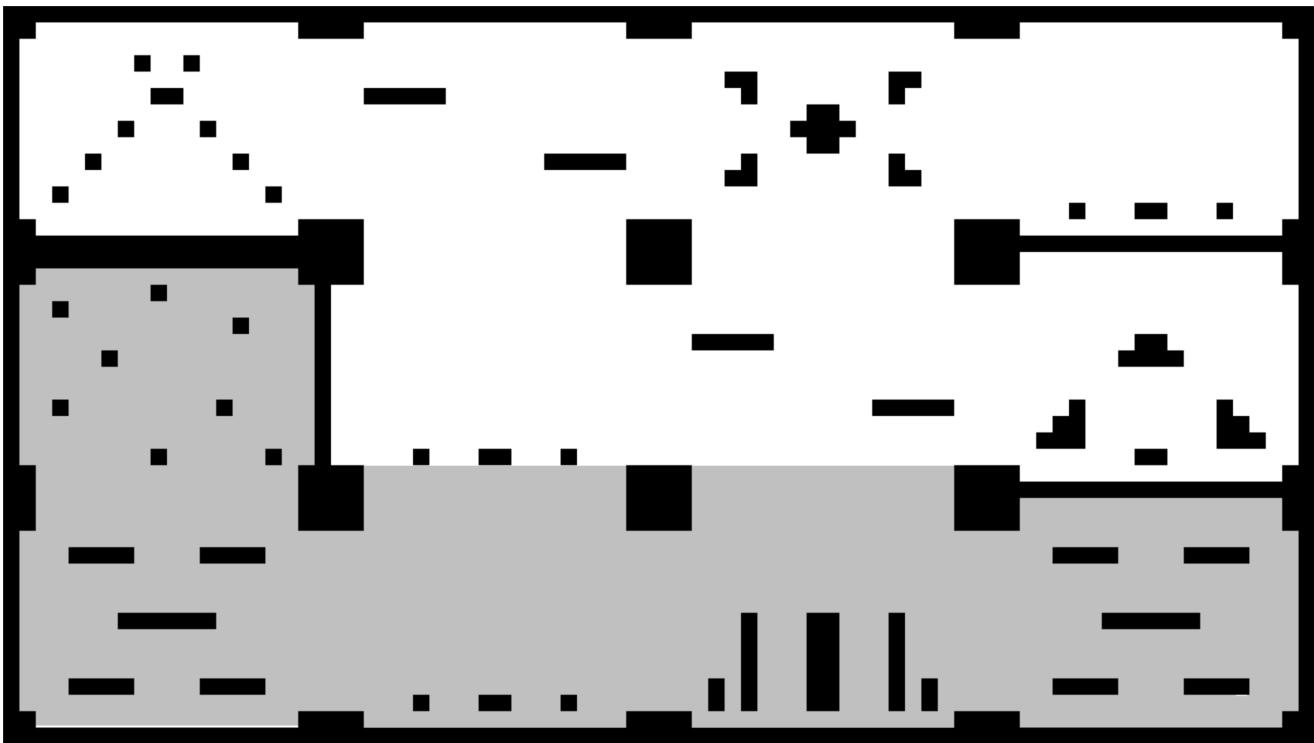
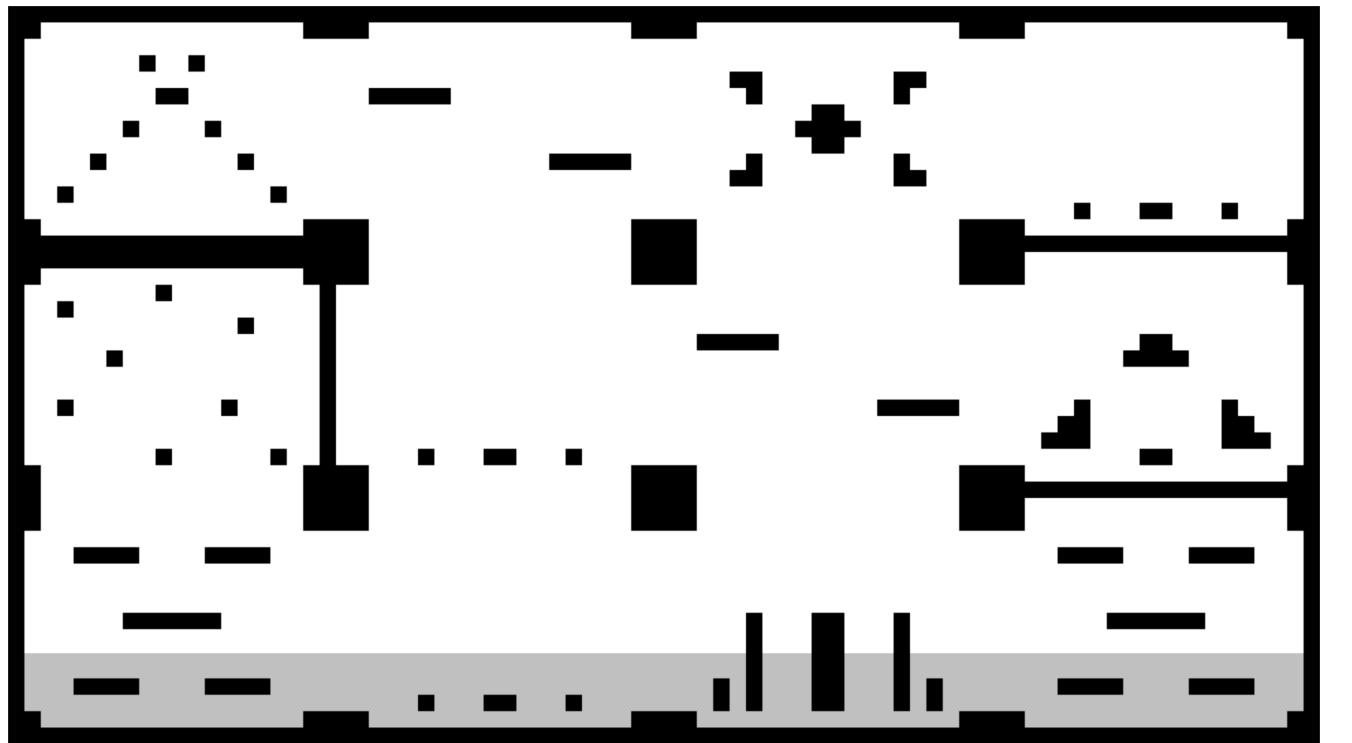
Levels

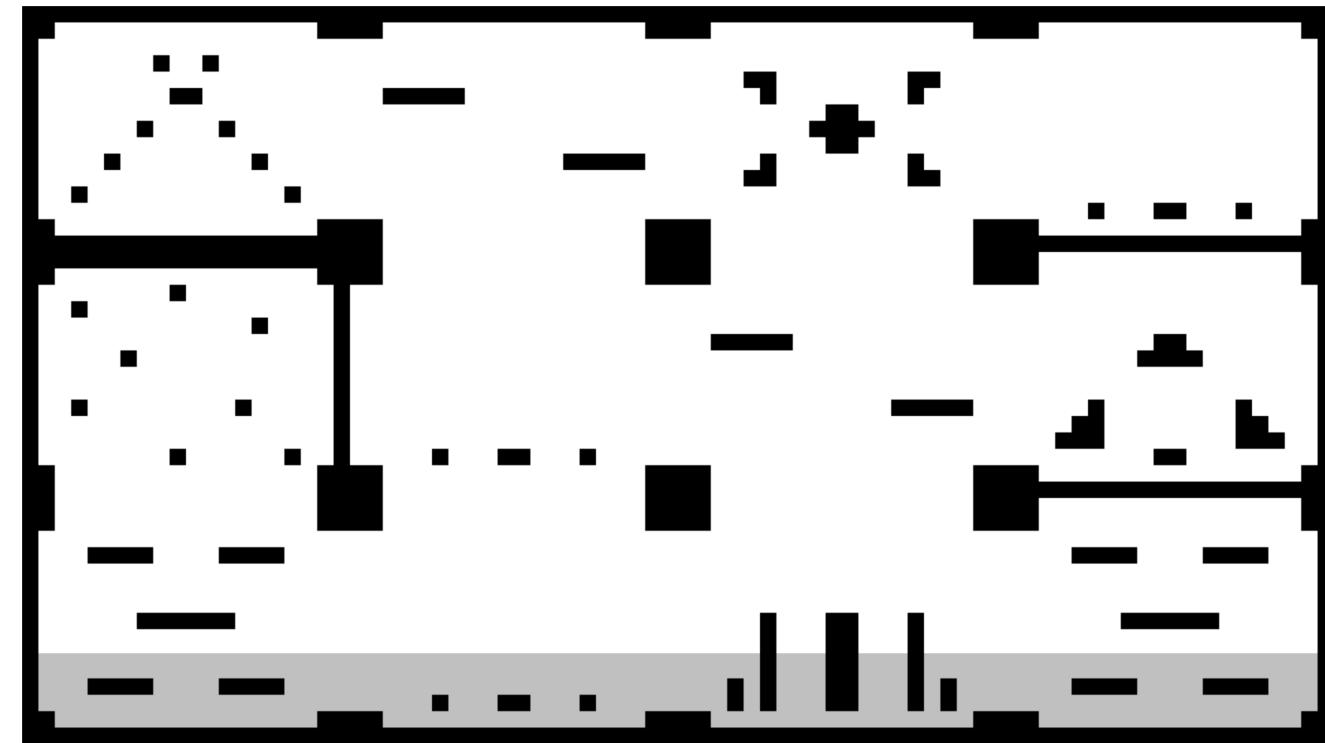


Layouts





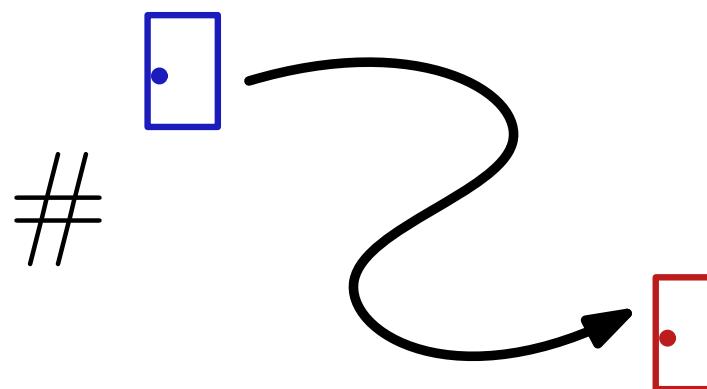




Not agents, but we still simulate



From starting point and player abilities, we can calculate regions where the player can reach.
If there was a powerup, we can calculate regions where the player can reach now.



No. of paths from start to finish

A → B → C

Progression, with gradual increase as we get more powerful throughout the level

gamesbyangelina.itch.io