

the **Algorithm Experience**



## WHAT?

Execute **different** algorithms which lead to the **same** result, by **hand**, on **small** instances.

## WHAT?

Execute **different** algorithms which lead to the **same** result, by **hand**, on **small** instances.

## WHY?

To make **tangible** the **cause** and the **effect** of efficient algorithms.

## WHAT?

Execute **different** algorithms which lead to the **same** result, by **hand**, on **small** instances.

## WHY?

To make **tangible** the **cause** and the **effect** of efficient algorithms.

## WHO?

Students, teachers,  
programmers, users,  
children, ...  
Everybody!

## WHAT?

Execute **different** algorithms which lead to the **same** result, by **hand**, on **small** instances.

## WHY?

To make **tangible** the **cause** and the **effect** of efficient algorithms.

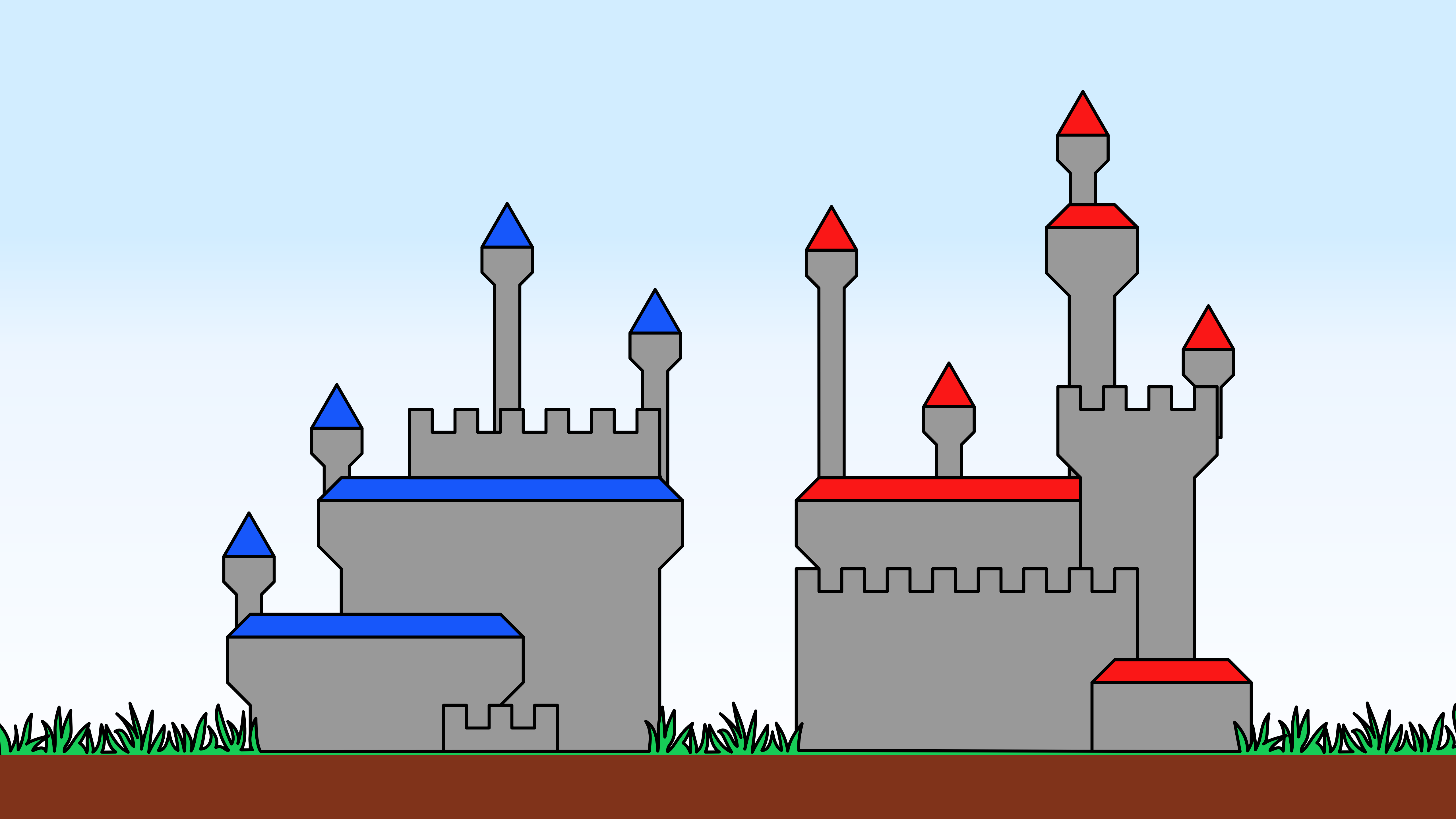
## WHO?

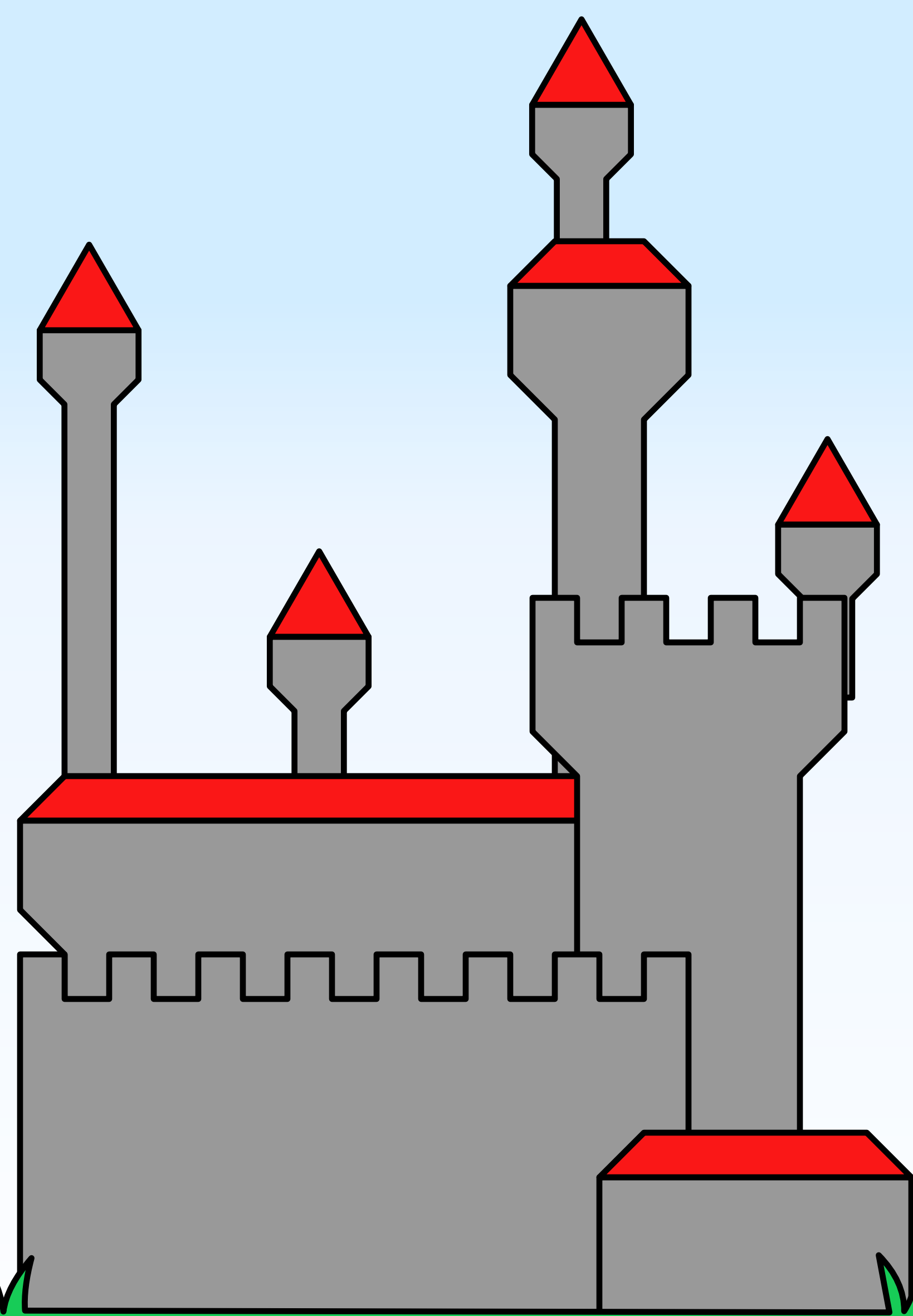
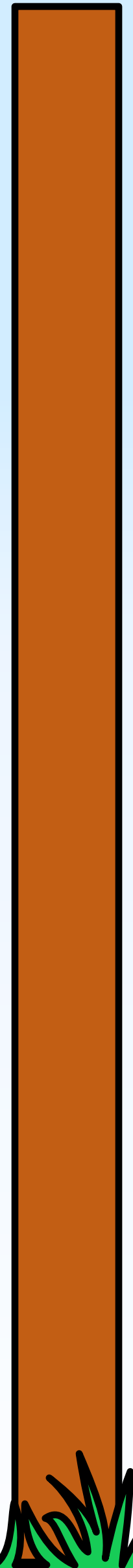
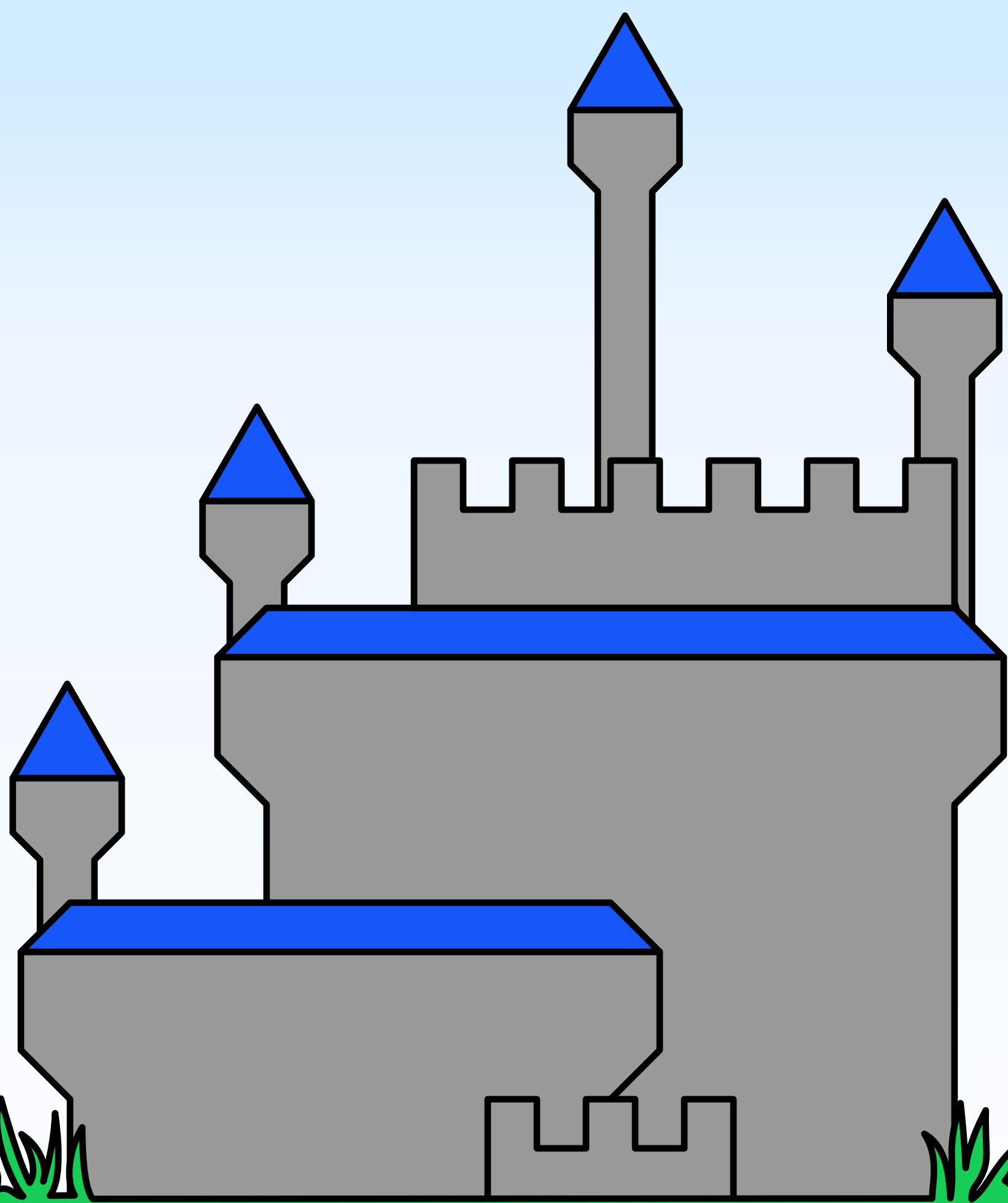
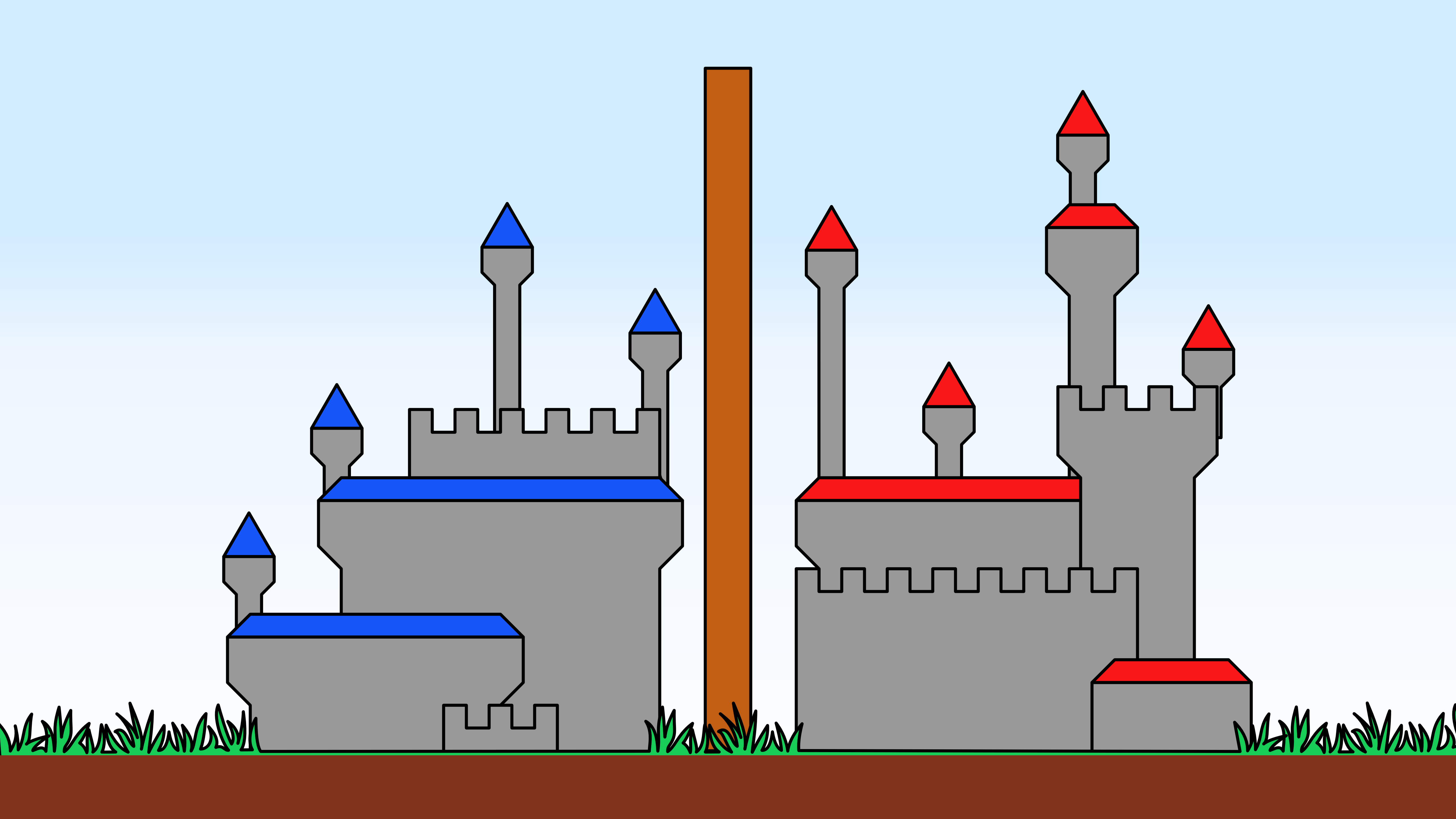
Students, teachers,  
programmers, users,  
children, ...  
Everybody!

## HOW?

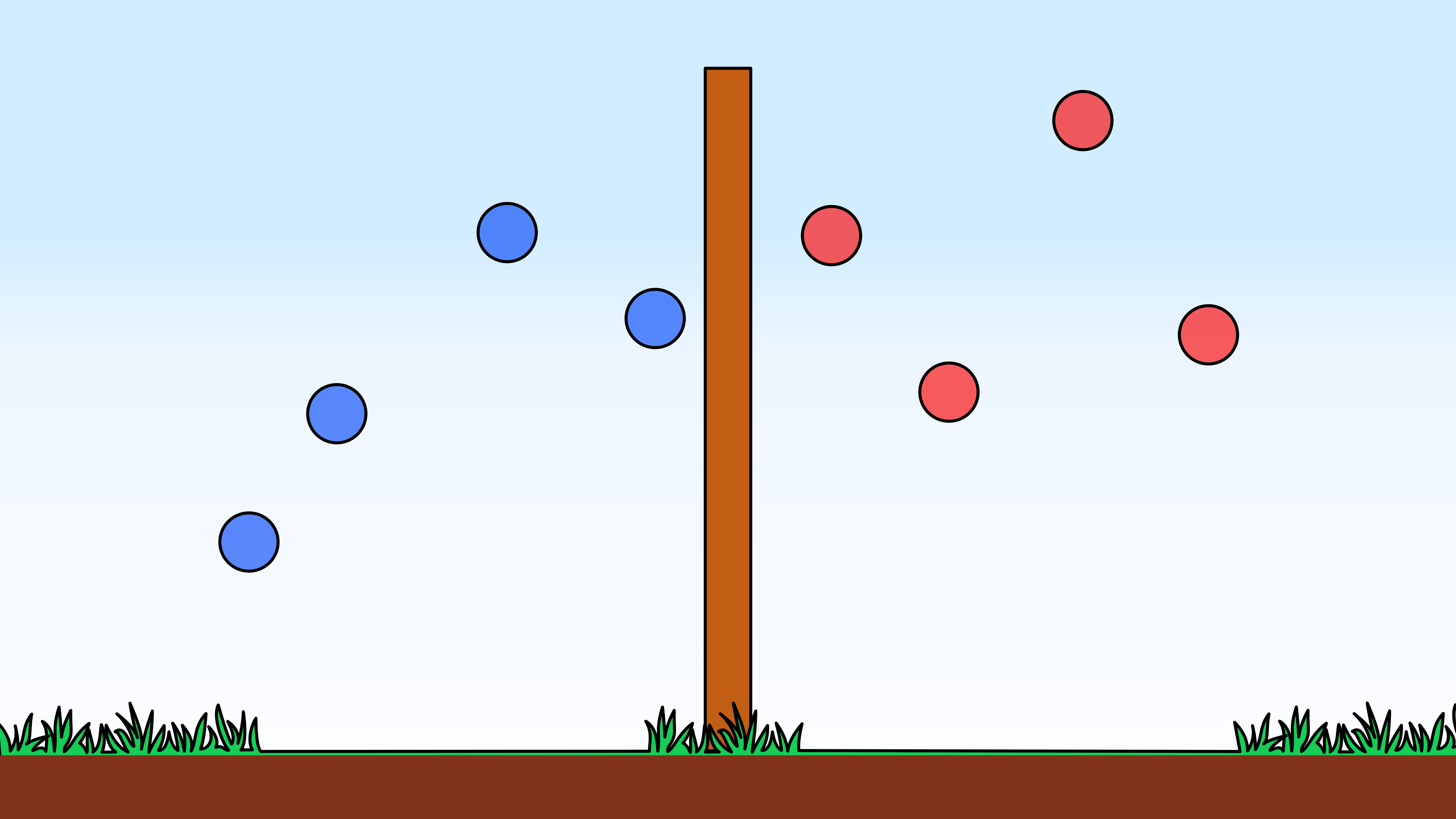
- Find out today:
- the **King's Problem**
  - the **Two Algorithms**
  - the **Paper Computer**

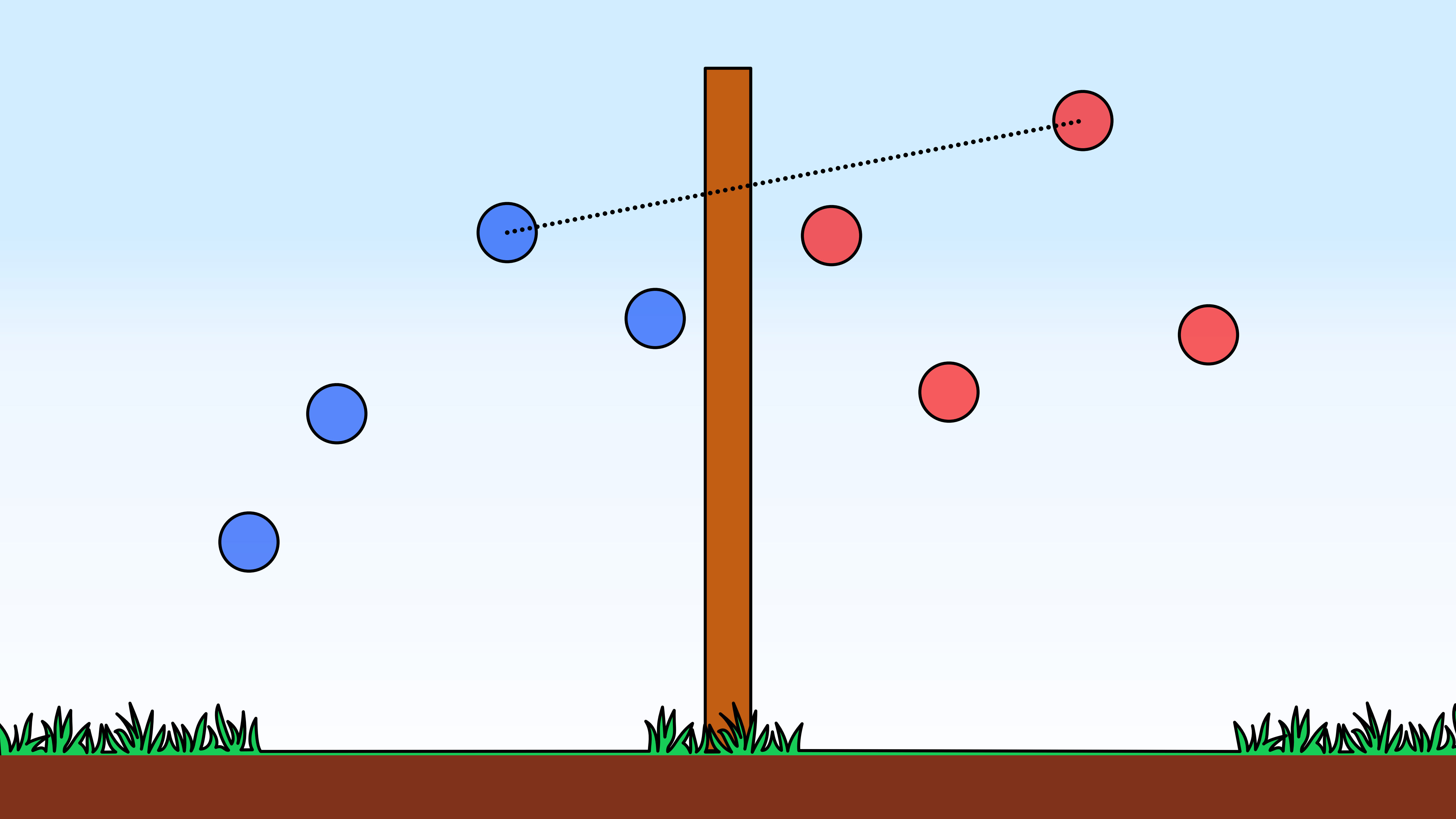
# the **King's Problem**



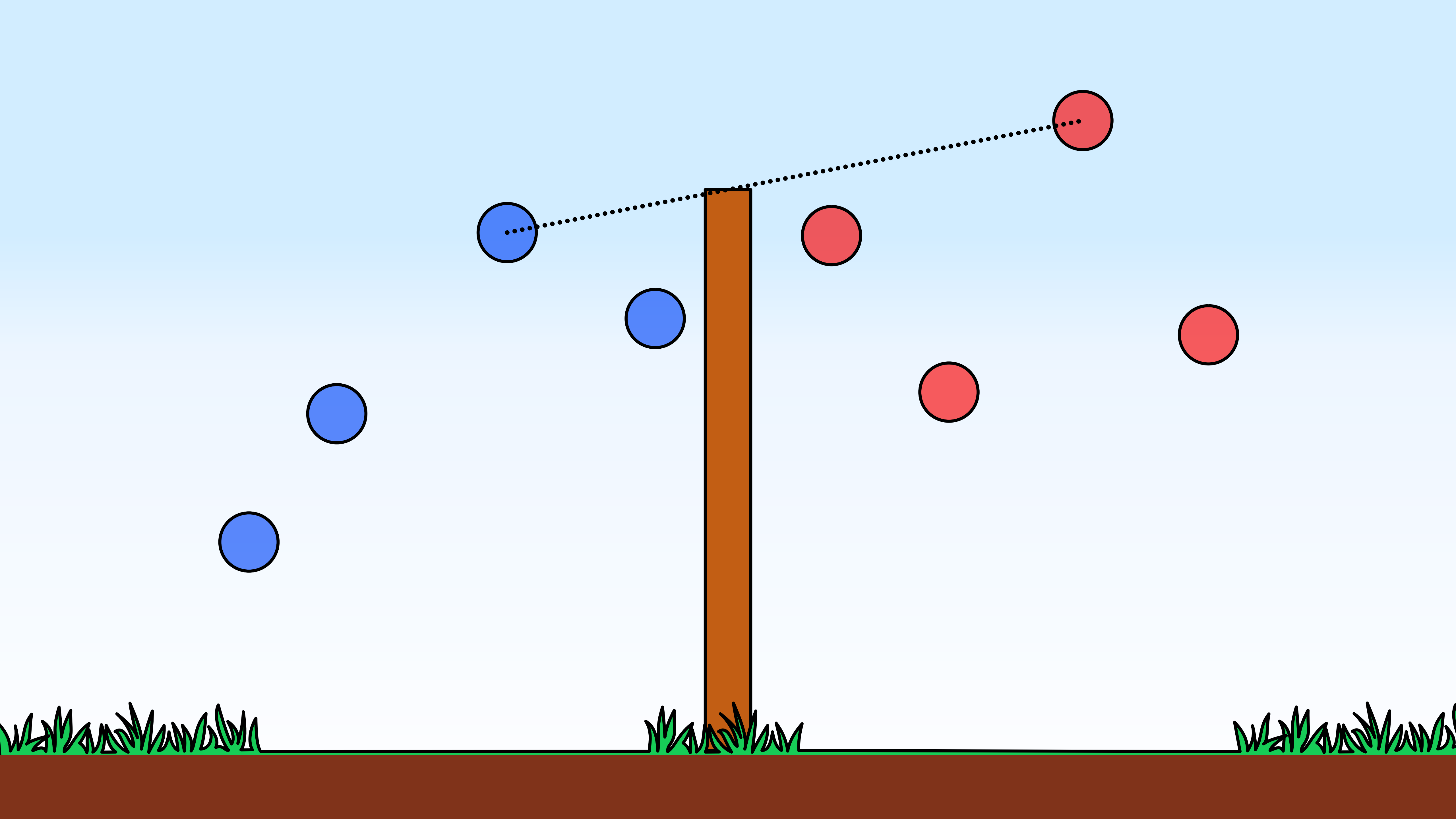


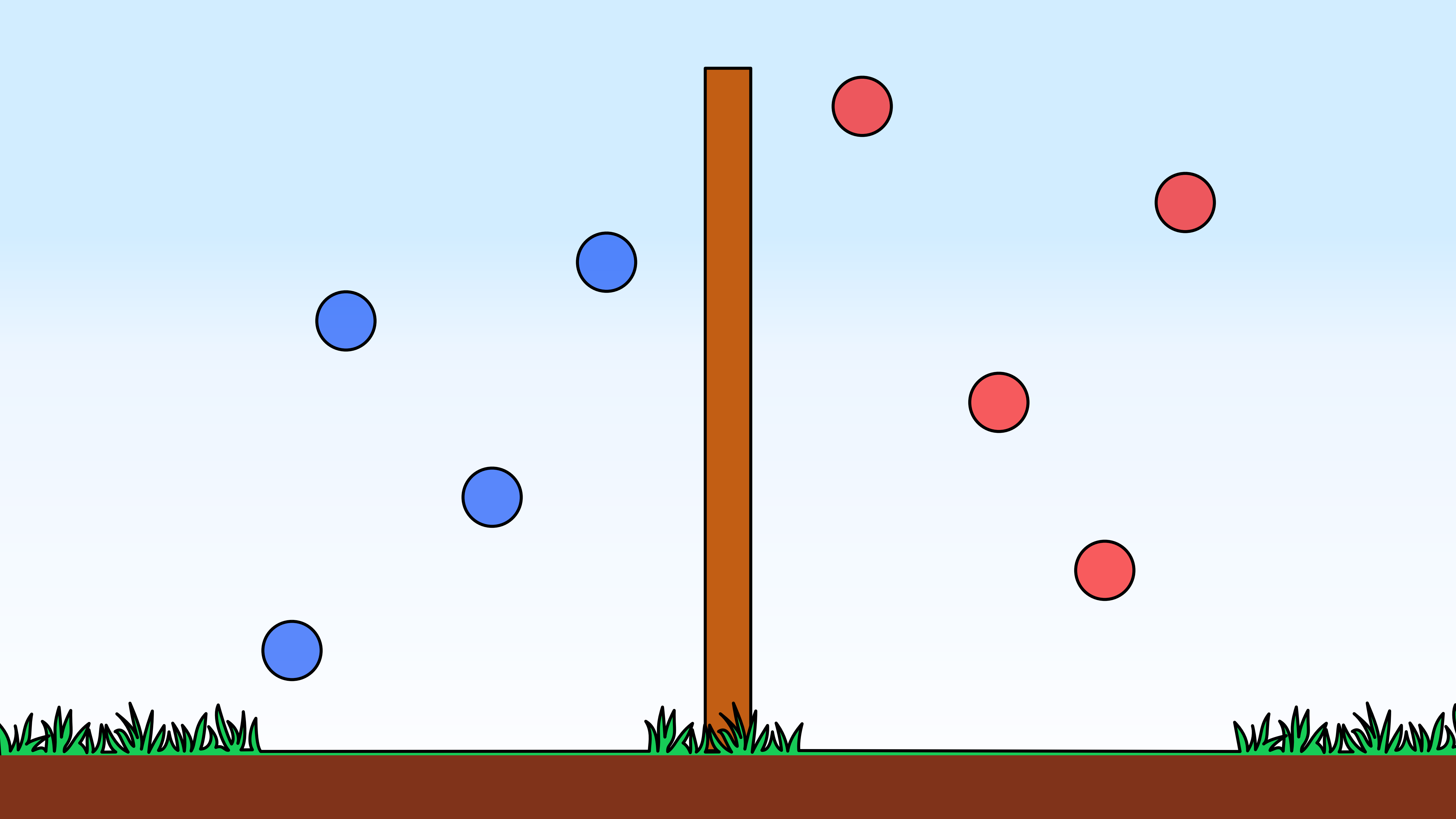




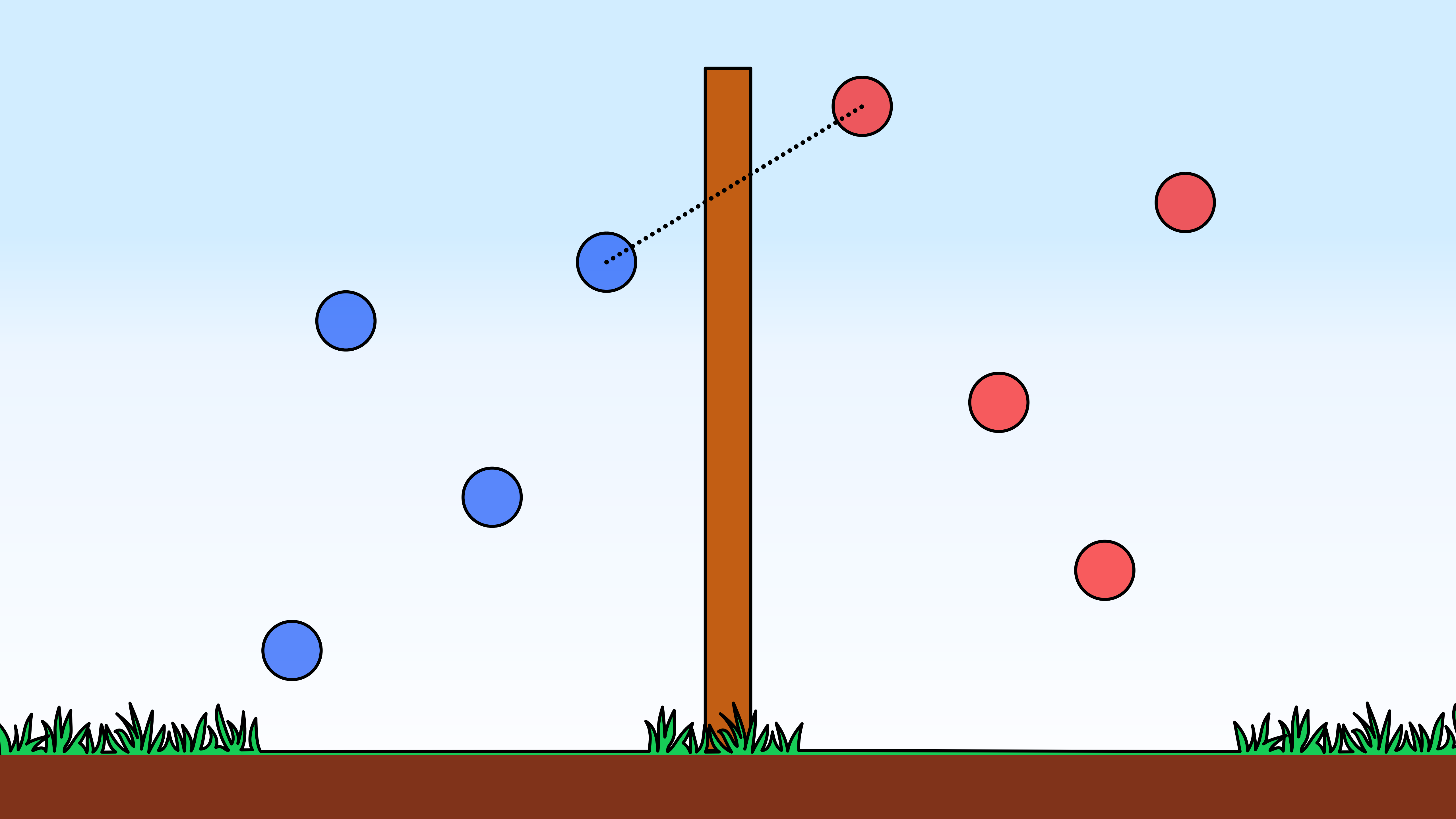


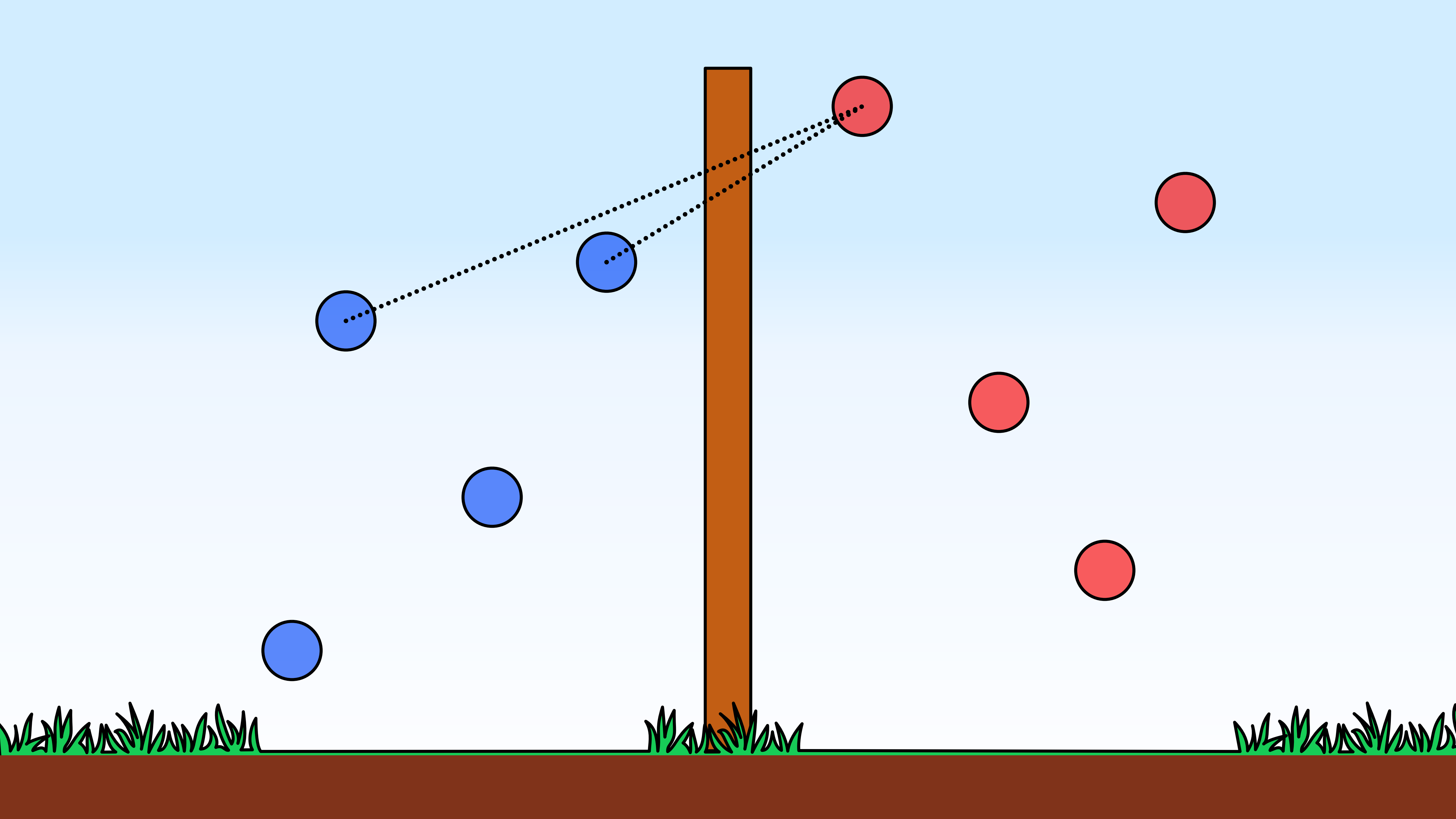




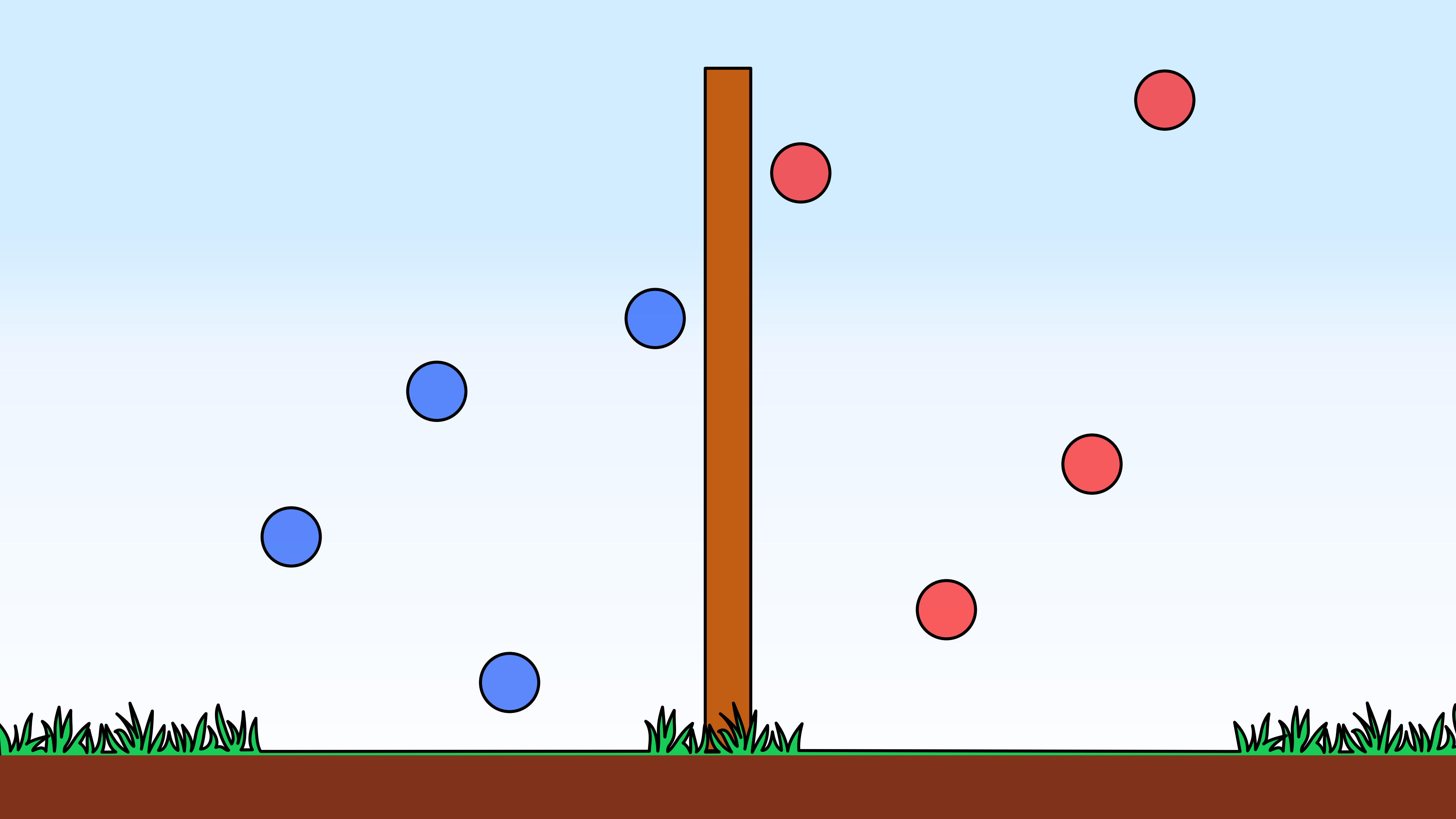


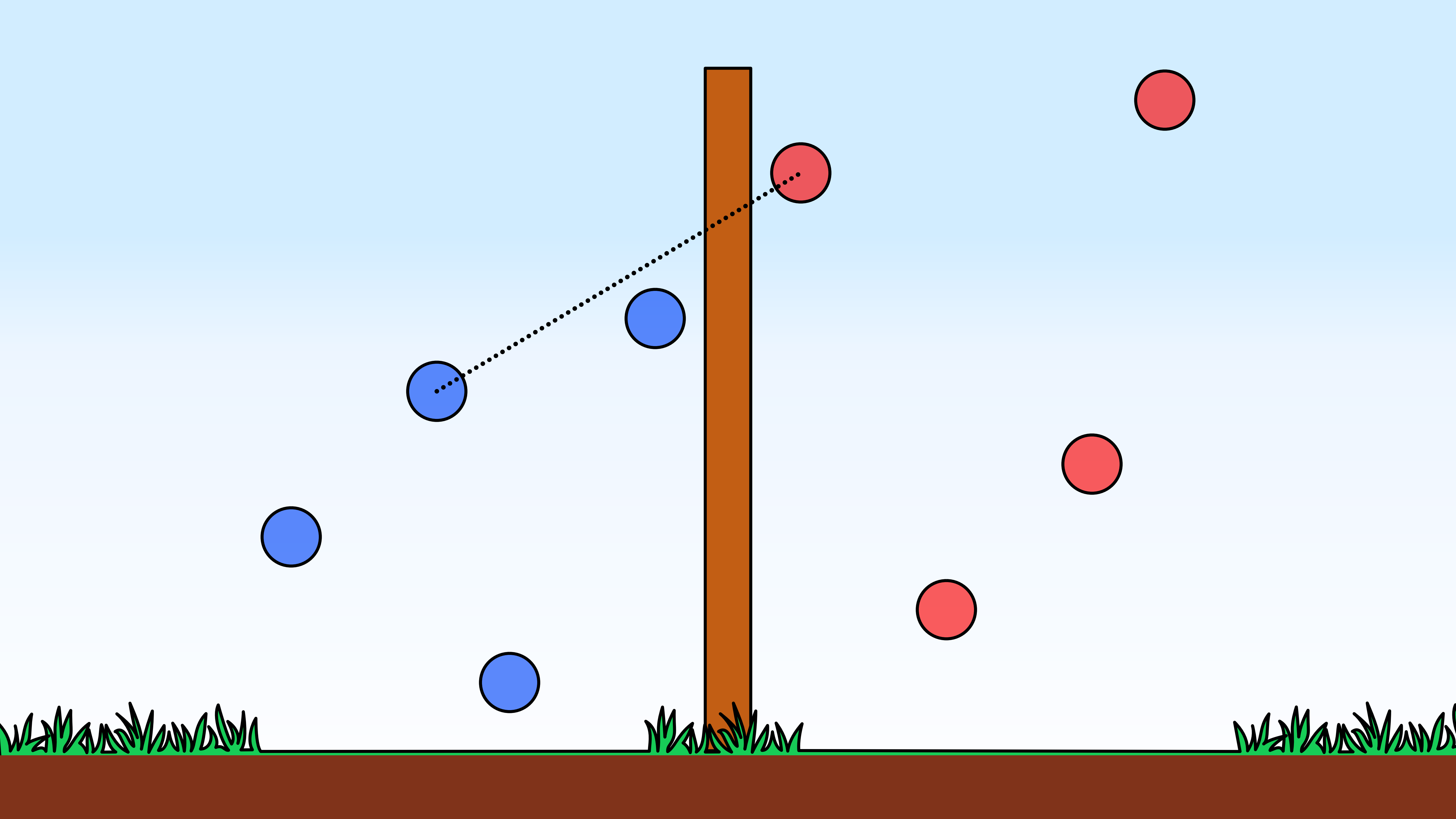




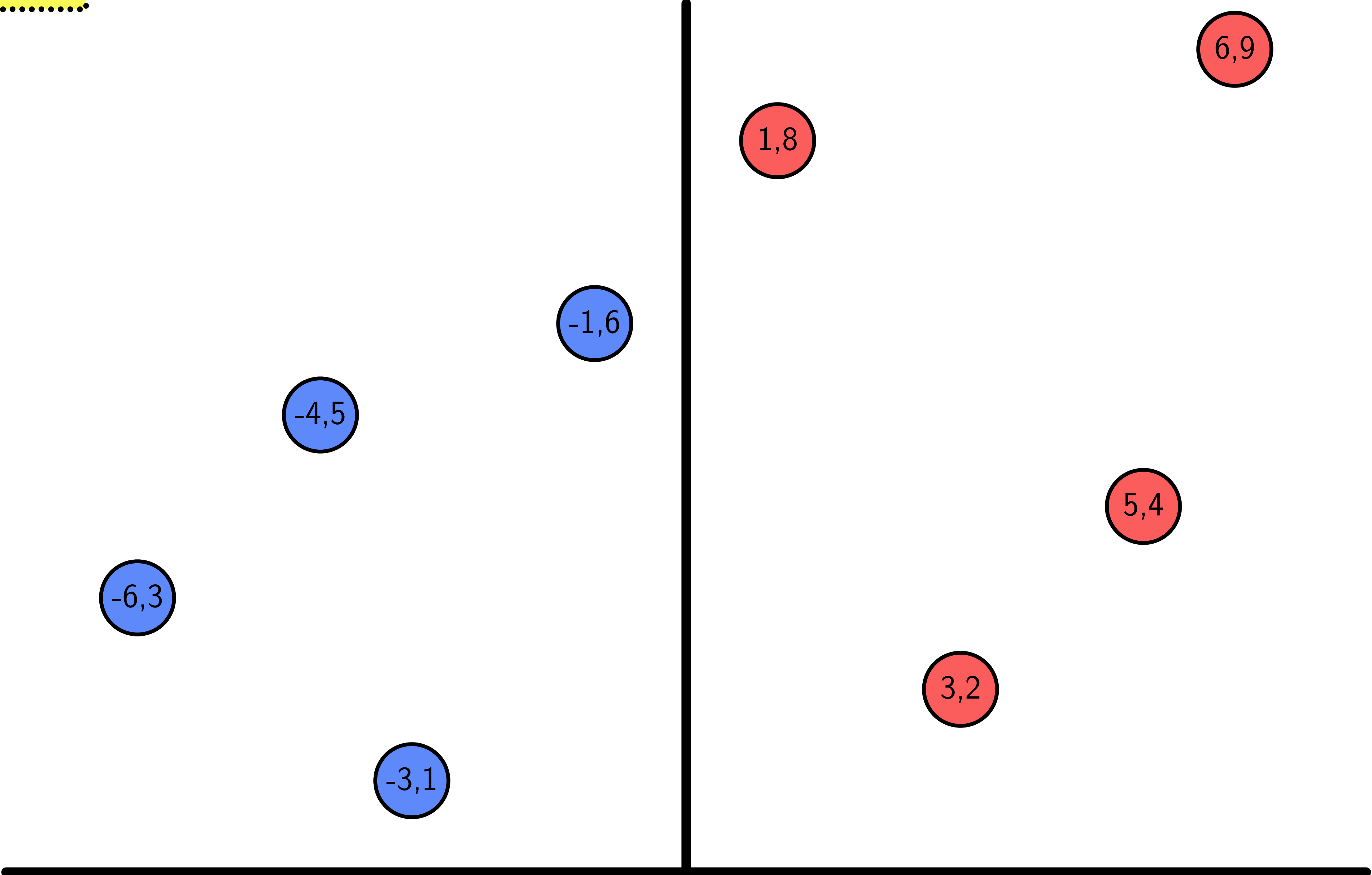








IN YOUR HANDOUT



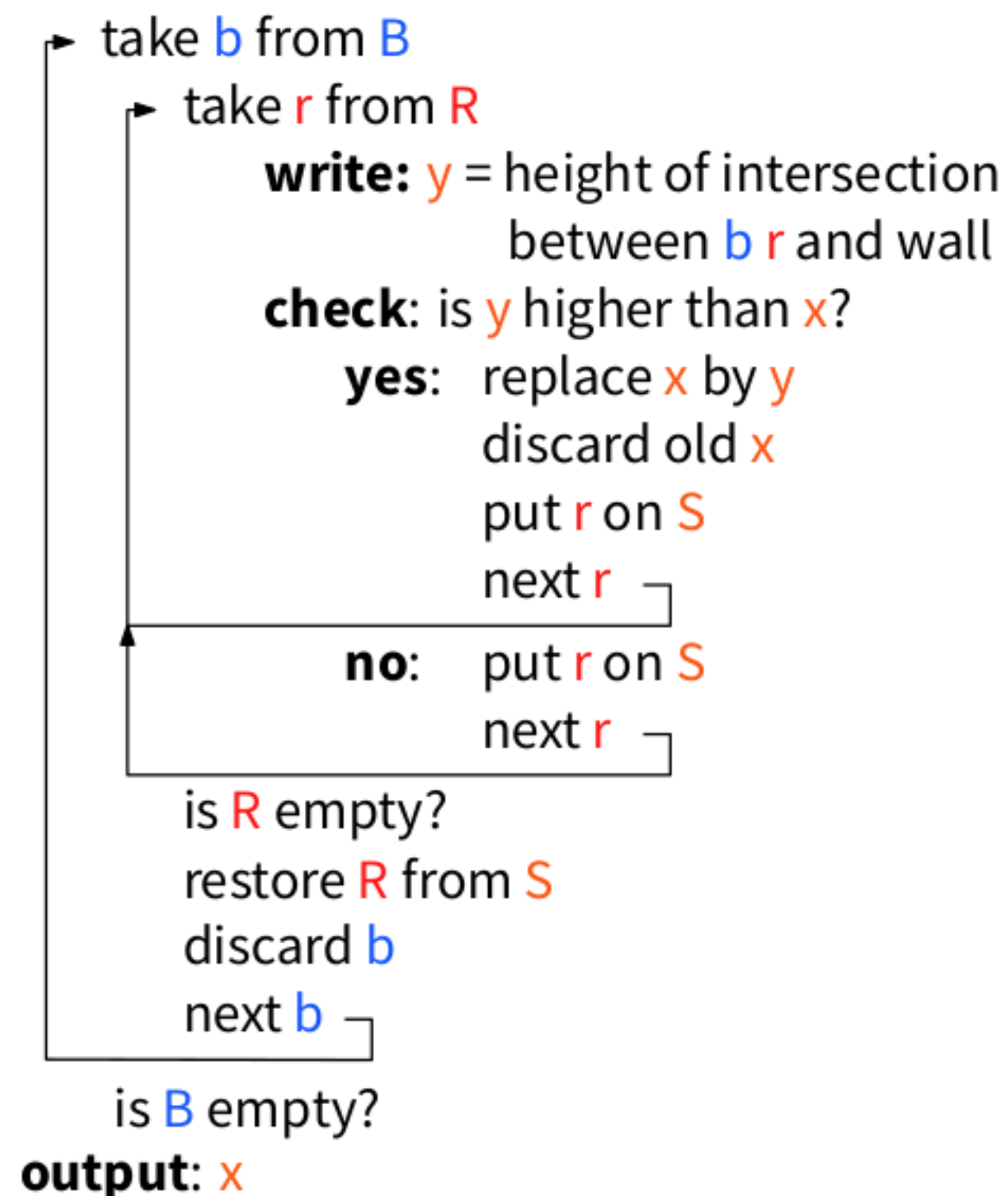
the **Two Algorithms**



## ALGORITHM 1

**input:** two stacks **B** and **R**

**write:**  $x = 0$



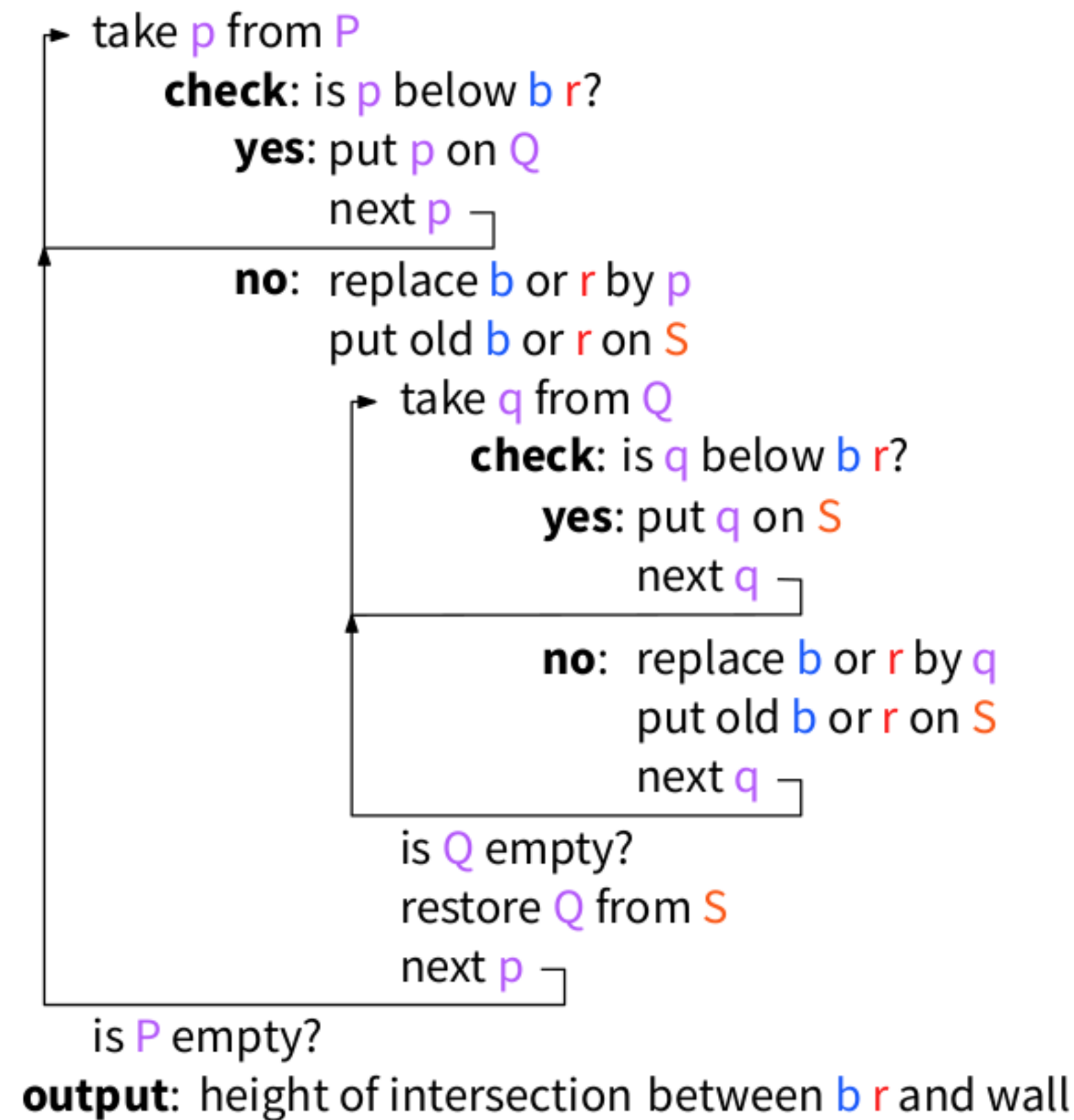
## ALGORITHM 2

**input:** two stacks **B** and **R**

take **b** from **B**

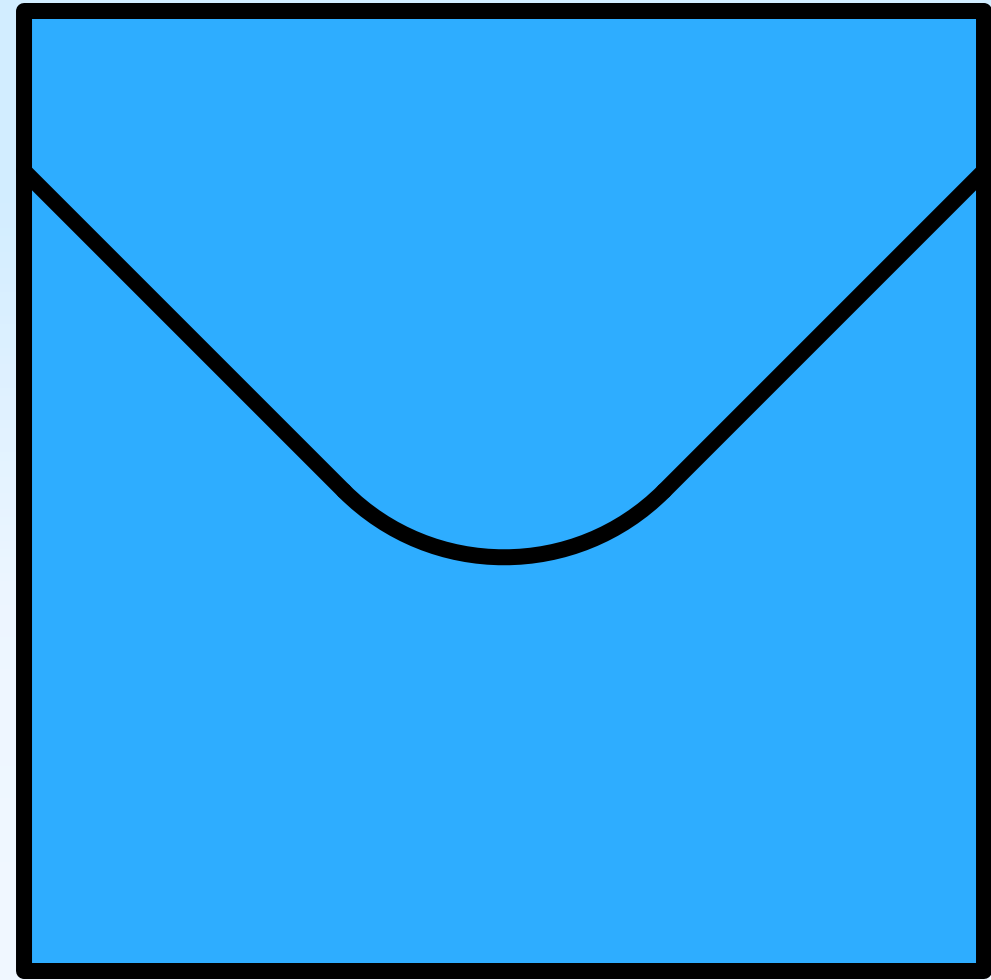
take **r** from **R**

**P** = shuffle the rest of **B** and **R** together



# the **Paper Computer**

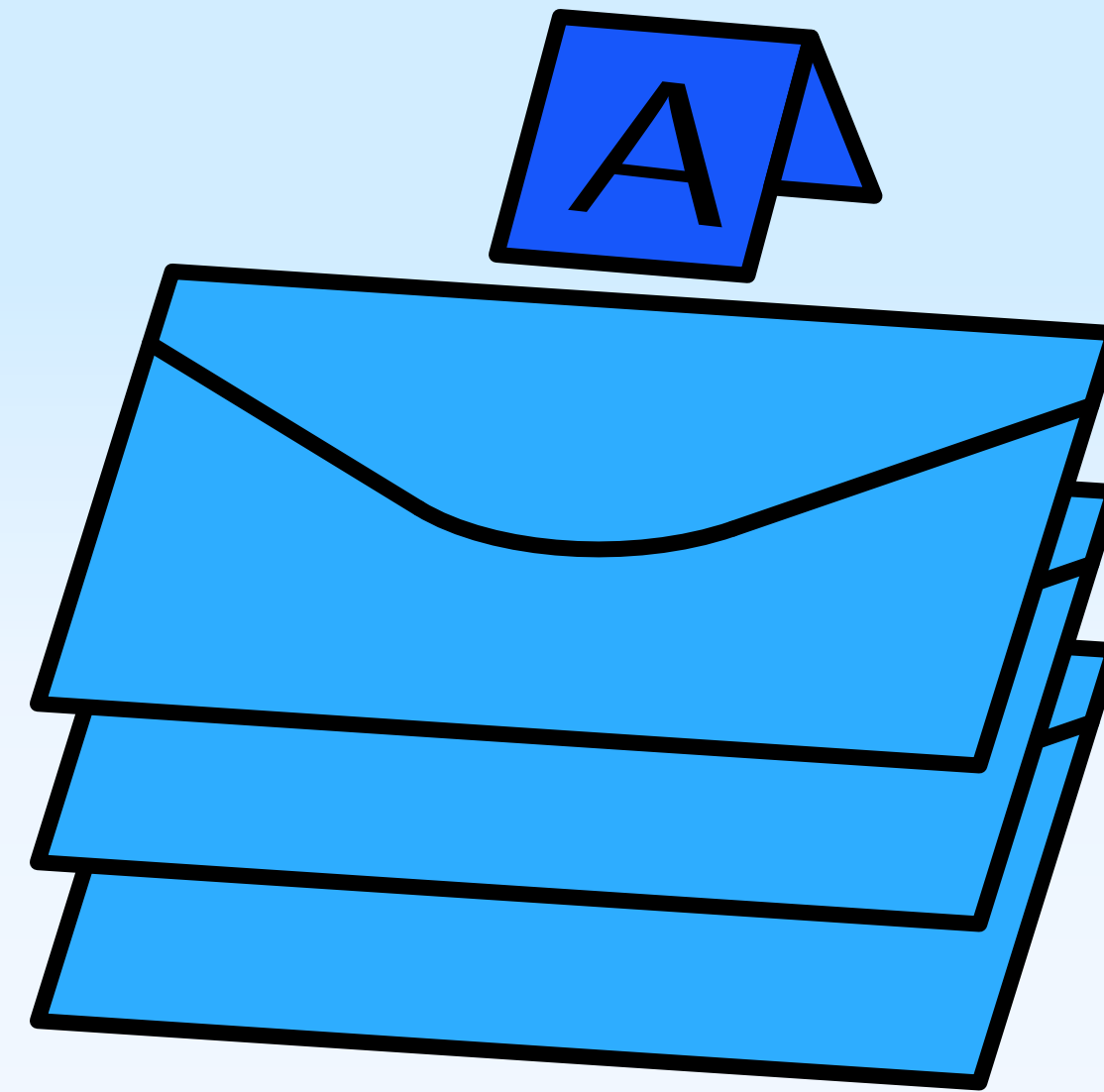
MEMORY CELLS



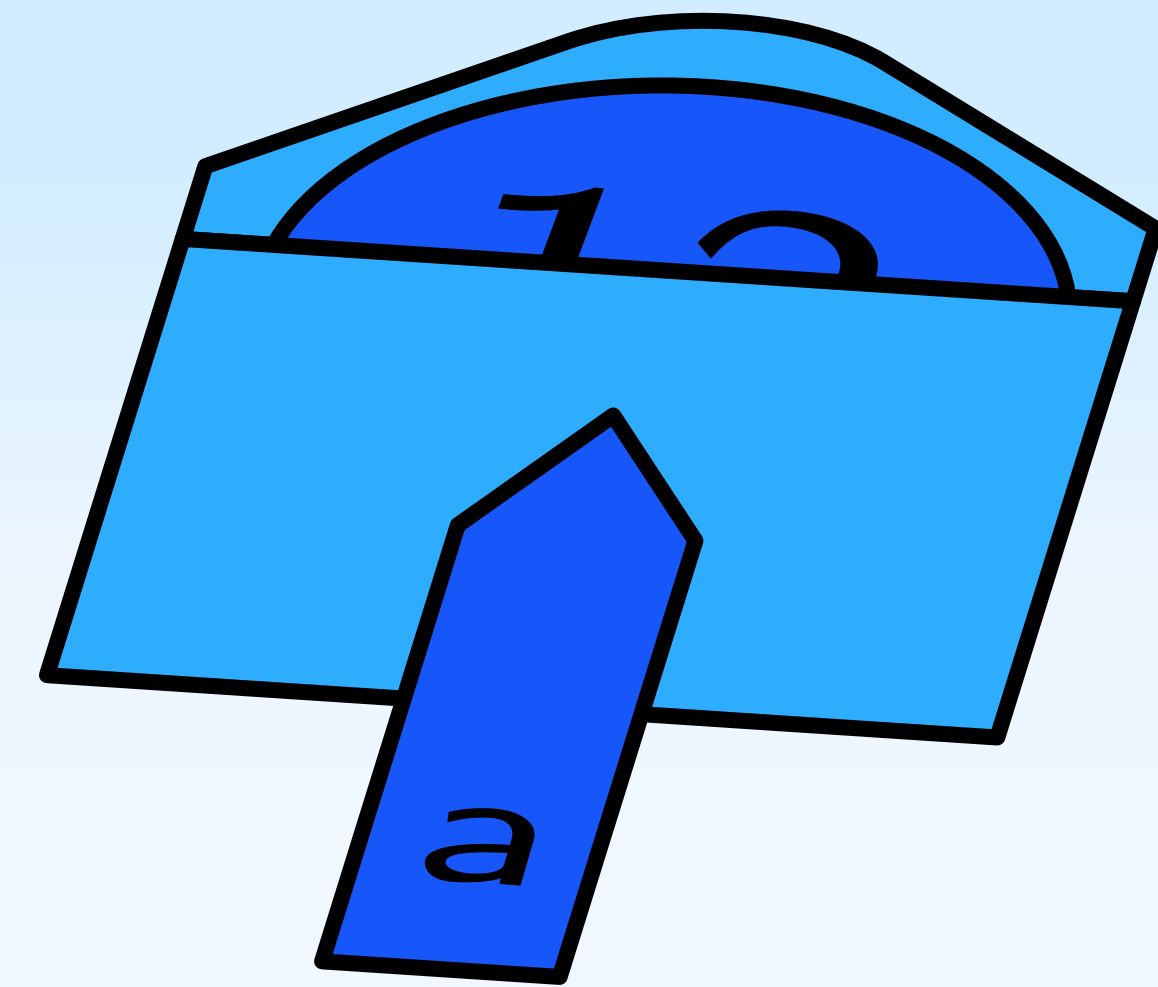
VALUES



STACKS



POINTERS



- Stacks contain multiple memory cells, but only closed ones.
- With a pointer to a cell, you may open it and look at the value.
- You can do simple checks and calculations on open values.
- You can write new values based on such calculations.

the **Experience**



- Execute Algorithm 1.
  - When you are done, write down your time!
- Execute Algorithm 2.
  - When you are down, write down your time!