

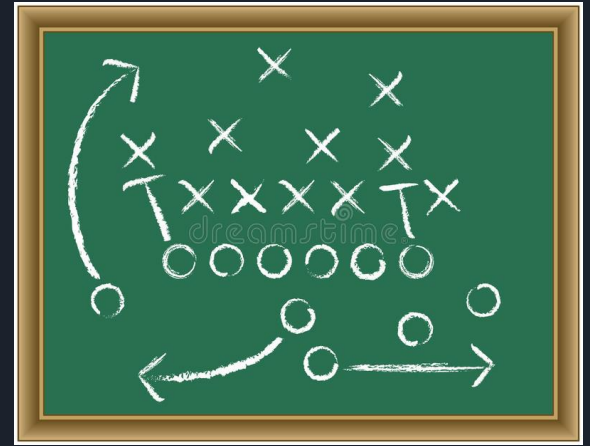
A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light green. They are positioned diagonally, with the blue one partially covering the green one.

# Our Game Presentation

Justin Dietrich and Ryan Scott

# The Game Plan

- Create a fun shooter game with a top-down view
- Dynamically generate an environment that has paths, walls, and other obstacles
- Unique enemies with their own AI and weapons
- Buttery smooth controls for the Player





# Top-Down Games: What Are They?

- A game perspective that views the action from above looking down
- Defining characteristics include being at a fixed position and/or rotation
- The Top-Down perspective is also known as “overhead perspective”

# Top-Down Examples



Bloons TD 6



Stardew Valley

## Top-Down Examples 2



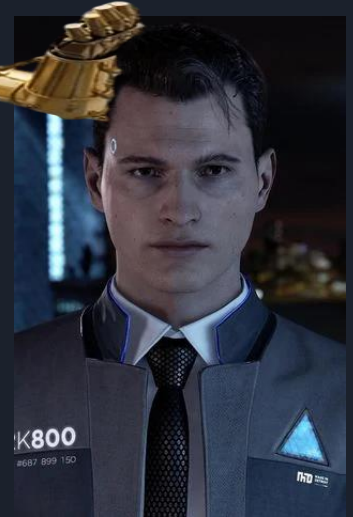
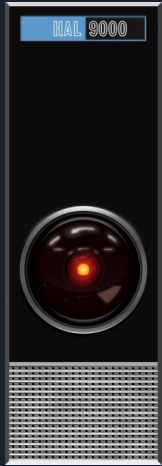
Binding of Isaac



Hotline Miami 2

# The AI

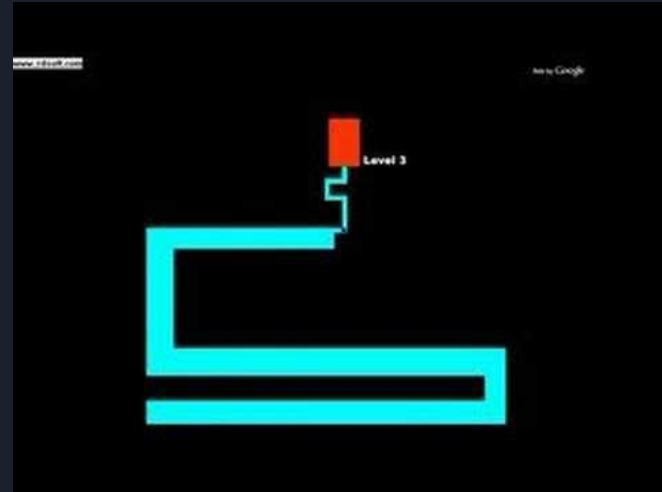
The next few slides will explain all of the AI elements that were implemented into our game from the projects.



# Pathfinding

The most important AI of any video game... pathfinding!

A\* is the best one, obviously, and we aim to make the best game so that's the one we have.





Many enemies in the game use A\* Pathfinding.

It uses a mix of trying to get closer to the player, while accounting for the cost of the position.

It is very well needed to navigate randomly generated levels.





# Constraint Satisfaction

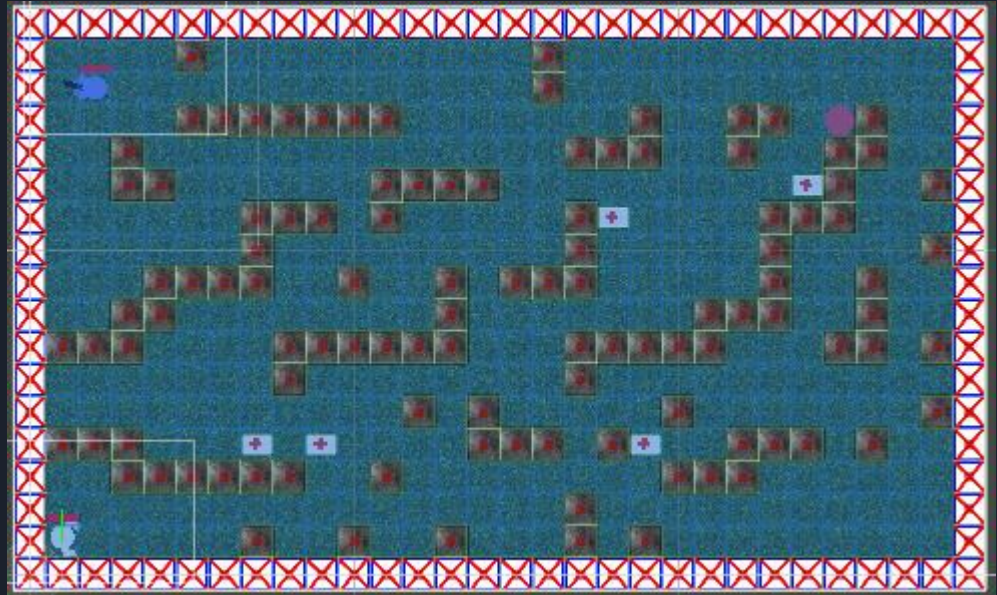
A constraint satisfaction problem is a problem where there can be any solution, so long as they follow the constraints.



Here are some of the maps generated with our constraints.

Really, the only constraint is to make every area of the level accessible while also having a good amount of walls so that more interesting interactions can take place.

The algorithm goes to every tile, decides if it should place a wall, snakes the wall in random directions until terminated by random chance or there's no possible open direction.



# Minimax Algorithm

A minimax algorithm calculates the best move a player can do, considering that both players play optimally.

One player tries to minimize the other's reward, while the other tries to maximize their reward.

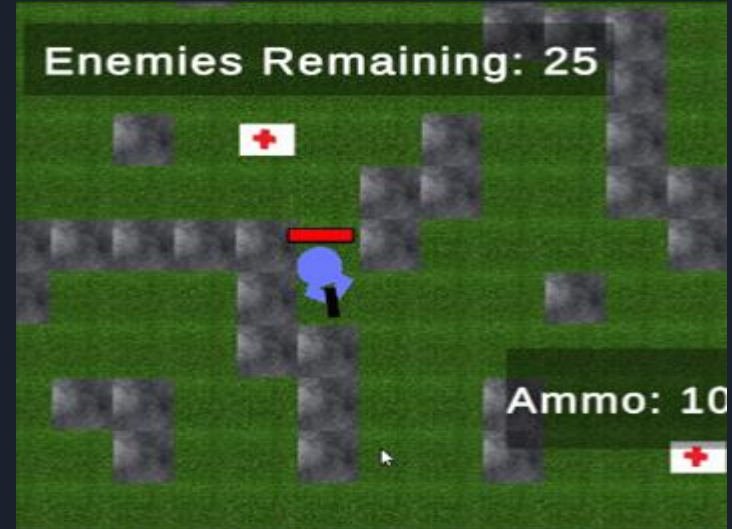
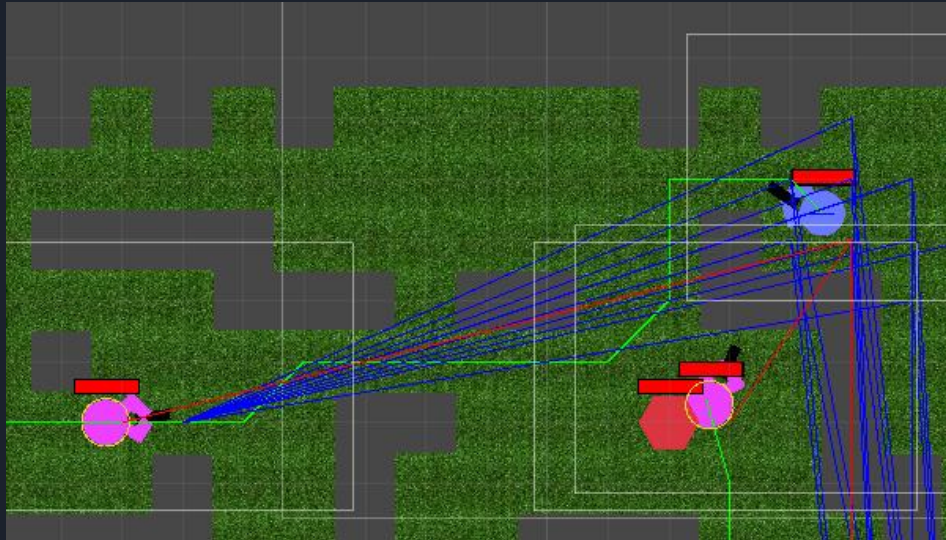




We have an enemy called the Runner that uses a minimax algorithm.

It gets points by getting close to the player while staying out of their field of view.

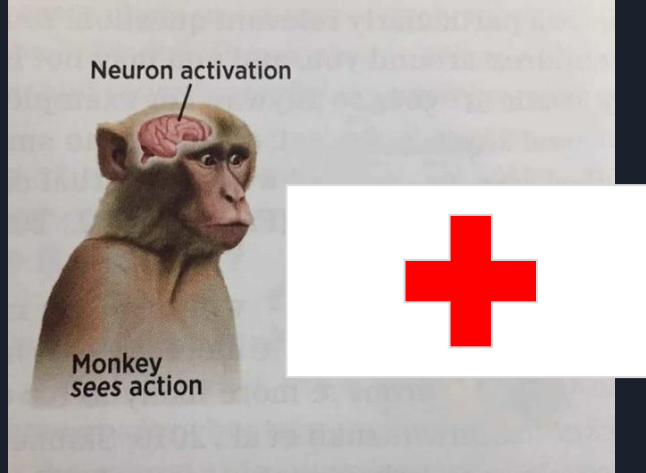
Being in FOV of the player loses it a lot of score. However, if it is close enough then being in FOV is negligible compared to the score it gains from staying close.



# Bellman Equation

The Bellman equation is an algorithm that find a path of choices with the highest score.

Score is calculated by setting a value reward for changing from one state to another.



The Brute enemy uses the Bellman equation when choosing where to move next.

It gets rewarded for getting closer to the player, going for no Line of Sight to LOS, and for stealing your precious health packs to heal himself.

Sometimes, there is no optimal option, since value for things decay so much if it would take too long.

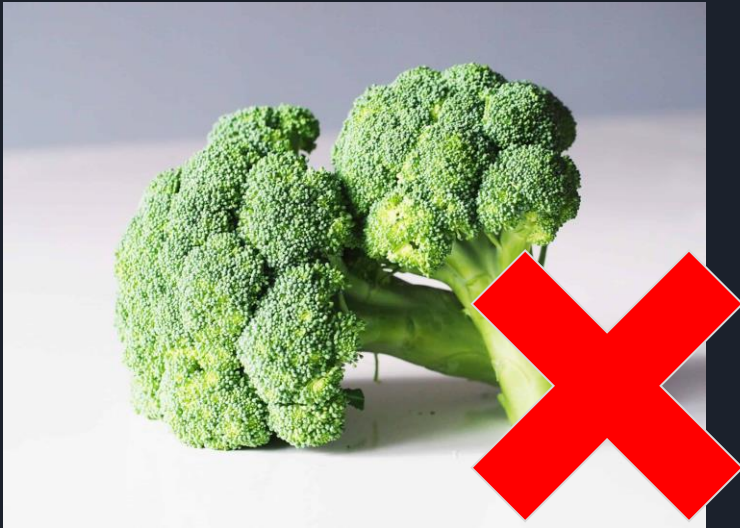




# Learning

You may not always know what is good and what is bad until you try.

This type of learning starts with an empty database and populates it with scores of what it tries.

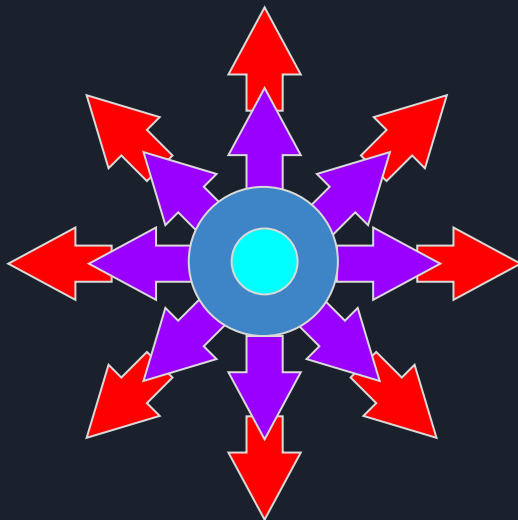
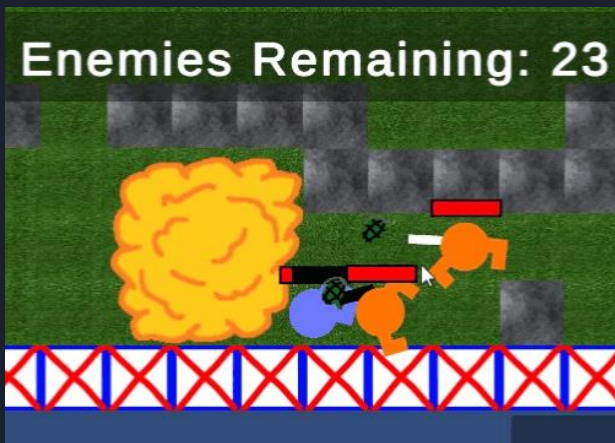


The Grenade Thrower doesn't know where to throw a grenade... and maybe you wouldn't either!

It can throw in 8 directions relative to your current direction.

It can throw in each direction 2 different distances.

Counting directly onto the player, that's 17 different ways to throw!

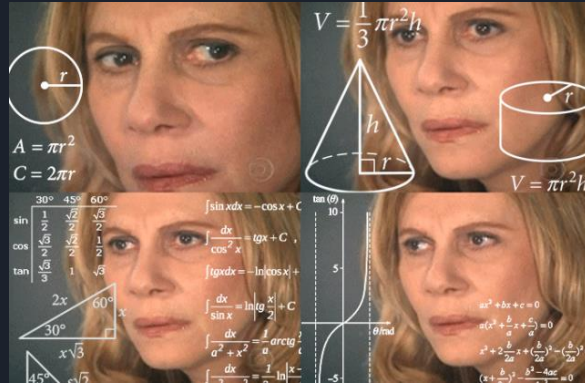


Direction	Close	Far
Copy Player	0.0	0.0
+45 degrees	0.0	0.0
+90 degrees	0.0	0.0
+135 degrees	0.0	0.0
+180 degrees	0.0	0.0
+225 degrees	0.0	0.0
+270 degrees	0.0	0.0
+315 degrees	0.0	0.0
Direct	0.0	NA

# More Statistics

If someone told you to guess what was about to happen, you would probably choose the most likely thing.

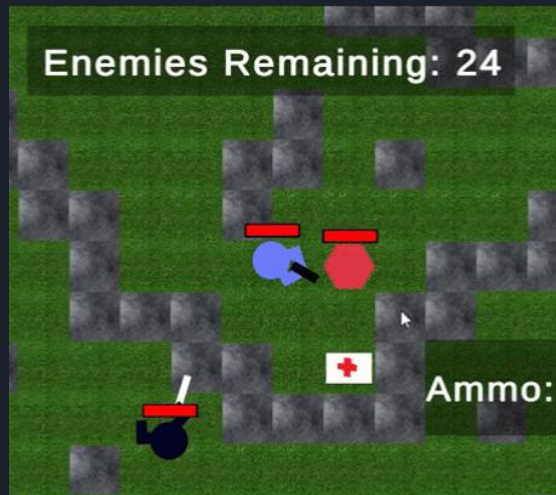
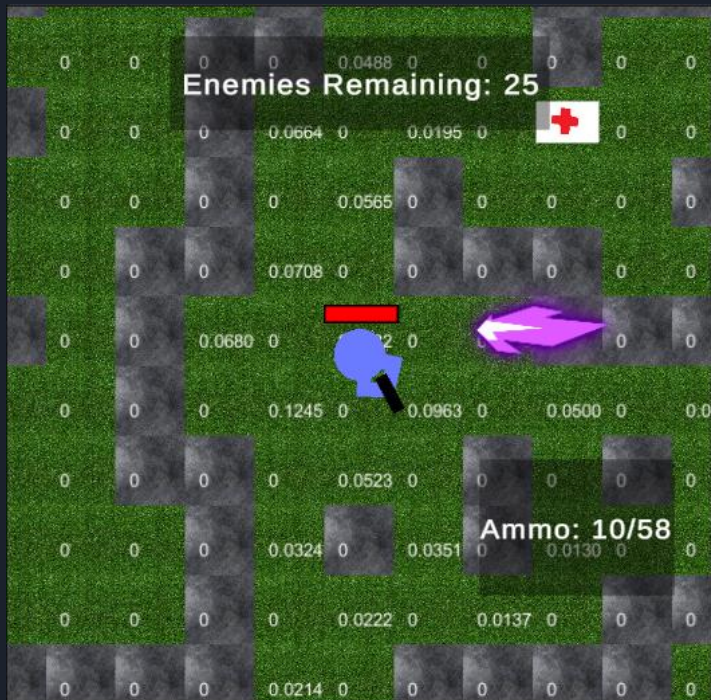
AI can also be used to calculate the odds given it has some odds to calculate off of.



The tower tries to calculate where you will be, given you move 5 tiles.

In the pictures, you can see the chance it thinks you'll be on each tile.

The tower operates off of this chance table.



Continue	0.5
Turn Right	0.2
Turn Left	0.2
Turn Around	0.1

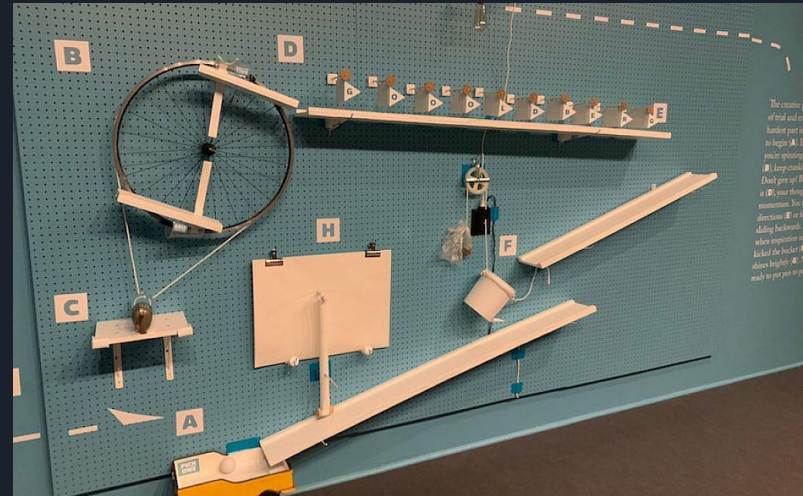


# Bayesian Network

Sometimes, things are unrelated.

BUT SOMETIMES THEY ARE.

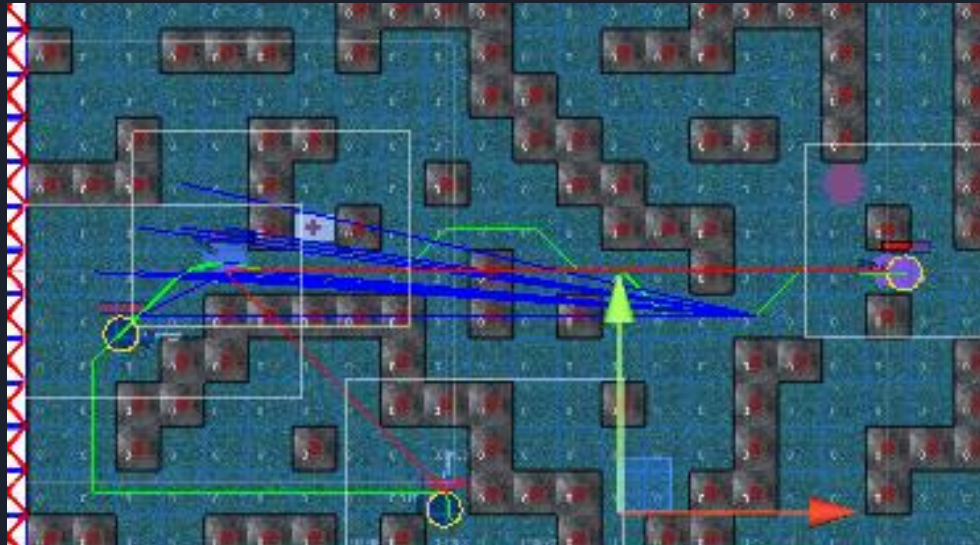
That is a Bayesian Network. It keeps track of what occurrences are related to others, and can calculate chances of things based on a network of other possibilities and occurrences.



The spawning of enemies utilizes this concept to predict the direction you will travel in.

Is the player at low hp? They may want a health kit. Is there a health kit nearby? Then they are likely to travel toward it.

In the picture, an enemy has just spawned in the direction of a nearby health kit.





# OUR GAME



You play as our hero. He has nothing to lose and a thirst for vengeance. He once only cried for 1 minute and 23 seconds after getting stung by a bee. To survive, he will have to utilize his expert skills, and use the resources available at his disposal throughout the levels.

# Bad guys



Bad Guy - Equipped with a gun and sometimes a knife, they will stop at nothing to hunt you down and make you have to restart.

Brute - He lumber slowly toward you when you're close. While not very smart, he knows when the health kit hits like the cold brew. His gun is quite huge, so he can't move while shooting.





## Bad guys



Runner - He annoyingly stays behind cover, unless he knows he's got you. He can be a nuisance that block your exits.

Grenade Guy - He throws grenades. They take a second, but cover a large area.

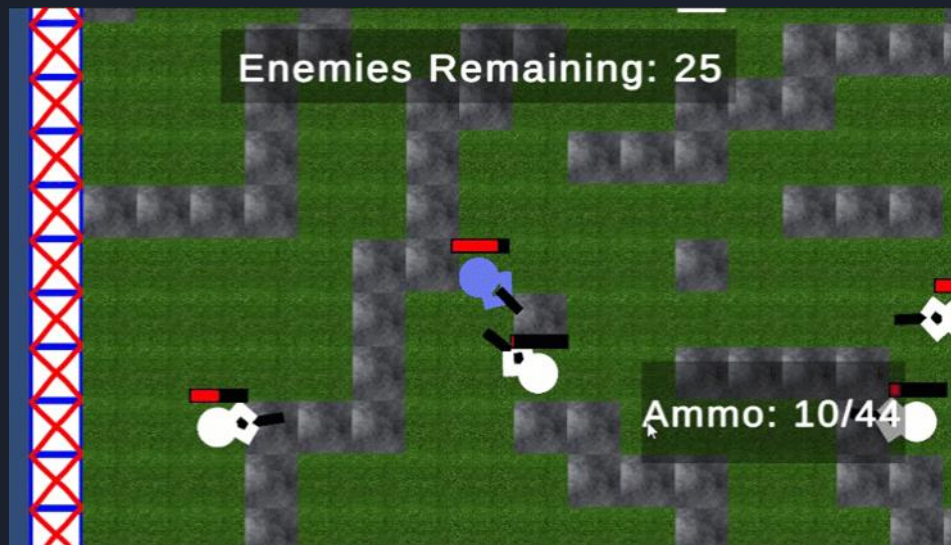




The tower - It shoots missiles at the player. It has a large travel time, but missiles fire from any range.

# Swapping Weapons

- It wouldn't be a proper top-down shooter without the ability to pick up your dead foe's much cooler weapons!



And Now... It Is Time...

