

## Inspiration:

I had recently created a 2-D game of a virtual carnival, the point of it was to give the user the enjoyment of going to a carnival without leaving his/her house in a pandemic, though it would have been better for immersion if the game was 3-D, this project helped me understand the basics of ray casting and 3-D rendering and laid the groundwork for a future 3-D virtual carnival, it may prove useful for a virtual student made fest if the campus doesn't open soon.

## Functionality:

Places the player in a virtual square area enclosed by walls, and allows said player to navigate through it.

## How it was built:

- 1) I created a class of surface (similar to line segment)
- 2) Made an algo to check whether a surface exists in a particular direction (Find the angle from reference axis of both the end points of the surface and checked if the angle that we have chosen lies in between both of them)
- 3) Calculated the normal distance between surface and player calculated angle between the normal and the line of sight and normal then using basic trigonometry to calculate the distance
- 4) This distance along with the angle was used to calculate the apparent height of the surface on screen

## Challenges Faces:

- 1) I wasn't using any online resources so I had to calculate and derive how vision at an angle works, there was a lot of experimentation involved.

## Things I intended to do but couldn't:

- 1) I was going to combine the Map Gen and Renderer to create a new maze every time, but ran out of time to do so.
- 2) There are optimizations for the renderer that I can think of but ran out of time to do so

## Team:

Netradeepak Chinchwadkar

Bhuvan Jayam