Pseudo code Cave explorer

The point of this program/game is to randomly generate cave structures and navigating through them with different difficulty settings.

Dicculty Easy:

No enemies generate and the chanse for the way out is increasing by 10% every room you pass through

Difficulty Normal:

Enemies generate and the chanse of the way out is the same

Difficulty Hard:

Enemis generate and the chance of the way out is decreased to 5% per room

Ghosts:

Has 2 hp and moves slightly slower than the player and can go through walls

Player:

Has 4hp and moves faster than ghosts, cannot go through walls

Main module:

This module controls the player and the enemy positions

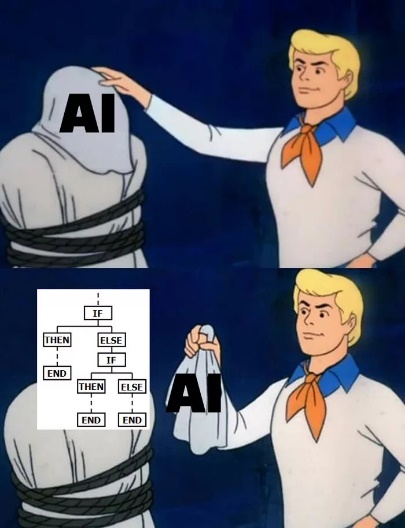
Wallhit(playerpos):

Checks if player is still on a floor tile and hasn’t moved in to a wall

If player is moving in to a wall them the player position will be reset

MoveGhost():

Basic Ai that checks the ghost position compared to the players and moves closer to the player.



Attack(Ghost)

Attack(entity)

Plays an attack animation around the entity and any other entity in that region will take damage

While gameOn =True:

If w is pressed:

Move player Up

If s is pressed:

Move player down

If d is pressed:

Move player Right

If a is pressed:

Move player left

If esc is pressed:

Draw.Menu()

If space is pressed:

Attack(player)

Wallhit(player)

Moveghost()

Draw module only draws out the map

Draw module:

Menu()

Create button start game

Creste button options

Create button exit

The button start gem leads to difficulty()

The button options opens a options tab

The button exit sets gameOn to False and terminates the application

Difficulty()

Creates button Easy

Creates button Normal

Creates button Hard

Button easy sets the difficulty to easy and leads to game()

Button normal sets the difficulty to normal and leads to game()

Button Hard sets the difficulty to Hard and leads to game()

Game():

Generates playercoord and vertexcoord

Random\_room.generate(playercoord,vertexcoord)

For row in range(0,width)

For col in range(0,length)

If tile with coordinate (col,row) =0:

Paint a rectangle

20% procent chance of enemy spawn

if tile with coordinate (col,row) =3:

paint a rectangle brown

Places a player on playercoord¨

Random room module changes tile values depending on calculations

Random\_room module:

Generate(player coord)

For x in range 0 to width:

y = graph(playercoord, vertexcoord,x)

tile with coordinates(x,y)=0

and generates some extra width for every y value

graph(playercoord, vetrtex,x):

Graph make a second degree ekvation and makes playercoords y value to C

And calculates the A and B value from the vertexcoord

Returns A\*x2+B\*x+C