Coding For Good

OPERATOR

OPERATOR

Jonah Sagers, Anthony Arcuri, Xiaoen
Ning, Baxter Hovis, Cormac McNees
[Table 4]

WHAT IS OPERATOR AND WHY IS IT CODING FOR GOOD?

Operator is a game which lets students take on graphing problems in a 3D environment.

We wanted to make sure students have access to math problems that engage them both on an educational and fun level. Many students have disdain for educational games, so we hope to break that stigma and encourage them to pursue learning.

WHAT MAKES OPERATOR UNIQUE?

- Rewards Pattern Recognition

- Targeted towards Middle School students beginning to take on algebra

- Provides a 3-Dimensional Representation of Graphs

Innovation

Procedurally Generated Levels

- Each level generates its own graphs

- Selecting between linear, parabolic, and sinusoidal equations as well as random points

Efficient Level Design

- Rather than brute force, the scene is able to adapt to different scenarios

Process / Challenges

- Unity Collaborate vs. GitHub
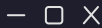

- Interpersonal Relationships

Ambitions

- Score tracking / motivation

- Multiple point graphs

- Interactive lesson style graphics, which adjust based on performance

GALLERY

Goal

Start

Input Method

Equation:
$$y = m * \sin(x - d) + b$$
values:
$$m = 0 \quad d = 0$$
$$b = 0$$

GALLERY

```csharp
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

☉ Unity Script (1 asset reference) | 0 references
public class PlayerTeleport : MonoBehaviour
{
    public string targetObject;
    public float posX = 0;
    public float posY = 0;
    public float posZ = 0;

    public Vector3 pos;
    // Start is called before the first frame update
    ☉ Unity Message | 0 references
    void Start()
    {
        pos = gameObject.transform.position;
    }

    // Update is called once per frame
    ☉ Unity Message | 0 references
    void Update()
    {
        pos = gameObject.transform.position;

        if (Input.GetMouseButtonDown(0) && pos.z <= (2.5) && pos.z >= (-2.5) && pos.x <= (2.5) && pos.x >= (-2.5))
        {
            posX = GameObject.Find(targetObject).transform.position.x;
            posY = GameObject.Find(targetObject).transform.position.y;
            posZ = GameObject.Find(targetObject).transform.position.z;

            gameObject.transform.position = new Vector3(posX, posY + 2, posZ);
        }
    }
}
```

Automatic Problem Generator

Create a graph which intersects the point (6, 8)

GALLERY

```csharp
public class RoomTP : MonoBehaviour
{

    public GameObject player;
    public Questions questions;
    // Start is called before the first frame update
    // Unity Message | 0 references
    void Start()
    {
        questions = gameObject.GetComponent<Questions>();
    }


    // Update is called once per frame
    // Unity Message | 0 references
    void Update()
    {

    }


    // Unity Message | 0 references
    private void OnCollisionEnter(Collision collision)
    {
        Debug.Log("yolo " + collision.gameObject);
        if (collision.gameObject == player)
        {
            player.transform.position = new Vector3(player.transform.position.x, 1.5f, player.transform.position.z-40);
            player.transform.rotation = (player.transform.rotation);
            questions.roomChange = true;
            //Debug.Log(player.transform);
        }
    }
}
```

```csharp
// Unity Script (2 asset references) | 2 references
public class Questions : MonoBehaviour
{
    public int roomNum = 0;
    public bool roomChange;
    public int chooseGraph;
    public Text txt;
    public GraphParticles grapher;
    public DoorMove doorMove;
    // Start is called before the first frame update
    // Unity Message | 0 references
    void Start()
    {
        grapher = GameObject.Find("Grapher").GetComponent<GraphParticles>();
        doorMove = GameObject.Find("Door").GetComponent<DoorMove>();
        txt.text = "Create a graph which intersects the point (5, 5)";
    }

    // Update is called once per frame
    // Unity Message | 0 references
    void Update()
    {
        if (roomChange)
        {
            roomNum++;
            roomChange = false;
            doorMove.isSolved = false;
            int xVal = Random.Range(1, 10);
            int yVal = Random.Range(1, 10);
            chooseGraph = Random.Range(1,5);
            if(chooseGraph == 1)
            {
                grapher.graphType = "SlopeIntercept";
            }
            if(chooseGraph == 2)
            {
                grapher.graphType = "ParabolaFunction";
            }
            if(chooseGraph == 3)
            {
                grapher.graphType = "SineFunction";
            }
            if(chooseGraph == 4)
            {
                grapher.graphType = "CosineFunction";
            }
            txt.text = "Create a graph which intersects the point (" + xVal + ", " + yVal + ")";
            grapher.targetPos.x = xVal;
            grapher.targetPos.y = yVal;
        }
    }
}
```

REFERENCES

Main Menu Image
https://stock.adobe.com/search?k=cyber+background&asset_id=298585211

Logo and Button Text Generator
https://flamingtext.com/

References
https://forum.unity.com/
https://docs.unity3d.com/Manual/index.html