Education Through the Entertainment of Video Games

Student: Lilyanna Yang, Advisor: Hubert Cecotti
Department of Computer Science, College of Science and Mathematics, California State University Fresno

Background

What is Serious Game?

- Serious Game is a branch of video game. Serious Games include an educational scenario with educational objectives for the players. This kind of game aims to allow the player to learn during the game [1].
- The game presented aims to enhance the player's cognitive abilities.
- It looks to improve the players mathematical skills in using mathematical functions such as addition, subtraction, and multiplication while also focusing on algebraic equations.

Game Engine

The Unity Engine

- This game was made using the Unity Engine 2022.30.3f.
  - Unity uses C# as its primary language.
  - All the games code was written using Microsoft Visual Studio.
  - Main Classes: Unity Engine and the MonoBehaviour class.
- Unity makes it easy to import assets, design levels using assets from the asset store and personally designed assets, UI, and Unity's own imported shapes and objects.

```
public void InitializeGrid()
{
    gridHandler = new Grid[MapWidth, MapHeight];

    for (int x = 0; x < gridHandler.GetLength(0); x++)
    {
        for (int y = 0; y < gridHandler.GetLength(1); y++)
        {
            gridHandler[x, y] = Grid.EMPTY;
        }
    }

    Walkers = new List<WalkerObject>();

    VectorInt TileCenter = new VectorInt(gridHandler.GetLength(0) / 2, gridHandler.GetLength(1) / 2, 0);

    WalkerObject curWalker = new WalkerObject(new Vector2(TileCenter.x, TileCenter.y), GetDirection(), 0.5f);
    gridHandler[TileCenter.x, TileCenter.y] = Grid.FLOOR;
    tileMap.SetTile(TileCenter, Floor);
    Walkers.Add(curWalker);

    TileCount++;

    StartCoroutine(CreateFloors());
```

```
public void mathRandomizer2()
    {
        if (isAnswers1)
        {
            rng2 = Random.Range(1, 10);
            rng3 = Random.Range(1, 10);

            randomY = Random.Range(1, Mathf.RoundToInt(bounds));

            secondValue = rng2;
            thirdValue = rng3;

            hiddenAnswer2 = int.Parse(firstNumber2.text + secondNumber2.text);

            if (quizMan.rngMath1 == 1)
            {
                answer2 = (secondValue + randomX) - (thirdValue + randomY);
                result2.text = "(" + secondValue + "X" + ") " + " + " + "(" + thirdValue + "Y" + ") " + " = " + answer2;
            }
            else if (quizMan.rngMath1 == 2)
            {
                answer2 = (secondValue + randomX) - (thirdValue + randomY);
                result2.text = "(" + secondValue + "X" + ") " + " + " + "(" + thirdValue + "Y" + ") " + " = " + answer2;
            }
            isAnswer2 = true;
        }
```

Artworks

Art

- The games Assets were made using Procreate.
- Supplies used
  - Apple Pencil
  - Ipad 10th Gen
  - Procreate

Discussion and Conclusion

Key Mechanics

- Game includes randomly generated numbers and insures players will be able to answer missing value based on the mathematical answer set in the games code.
- Game uses any controller plugged into the computer the game is running on.

Limitations

- Game only has three levels and no official end.
- Game walls, items, and decorations are placed by hand and not randomly generated.
- Game doesn't have complex math equations such as PEMDAS (Order of Operations).

Future Implementation

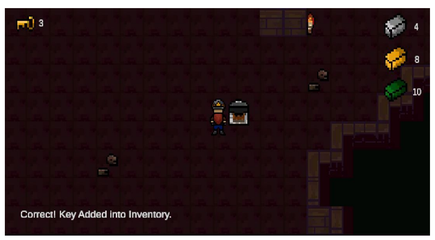
- Implement Start and End UI
  - Add an ending.
  - Add menu to main page.
- Add animations to character module.
- Add a timer
- Assess the game difficulty with human subjects

Game Mechanic

- There are 4 chests that have their own math equations. The equations compose of addition and subtraction in the format of algebra questions. X, Y, and Z uses different types of Gems as it's variable. The least chests uses a diamond instead of the answer.
  - **Chest 1** contents a simple algebra equations such as 2X = 10 for example.
  - **Chest 2** has a equations such as 3X + Y = 60 making the player remember what variable X is.
  - **Chest 3** uses the equation 2X + Y + Z = 45. The game requires you to remember the other two missing variables to solve the equation.
  - **Chest 4** uses all the gem elements for X, Y, Z to solve for the answer instead of the missing variable.
- Key will be added to the players "inventory" indicated by the text on the bottom left and on the top right.
- A diamond will be added to the players "inventory" when the last chest is solved.
- Each Chest uses Random.Range to determine a random number for the missing variable, the numbers multiplied by the gems, and the answers.

Map

- Map is randomly generated using walkers [6].
  - Walkers are similar to how people walk. Each walker has a set tile to paint on the map.
  - There are a max of 10 walkers
  - Each walker uses different X and Y coordinates randomly generated by Random.Range.
- The walls and items are placed by hand.
- Exit door uses fading dialogue to prevent the player from continuing without solving the puzzles. [7]
- Door transfers using Enter and Exit triggers to the next level [3].



Controller

- Game uses a Xbox One controller to move and interacts with things in the game [8].
  - The Joystick moves the character around in a 2D top down format [2].
  - "A" interacts with the door.
  - "X" interacts with the chest.
  - "B" exits out of the equation panel after interacting with the chest.

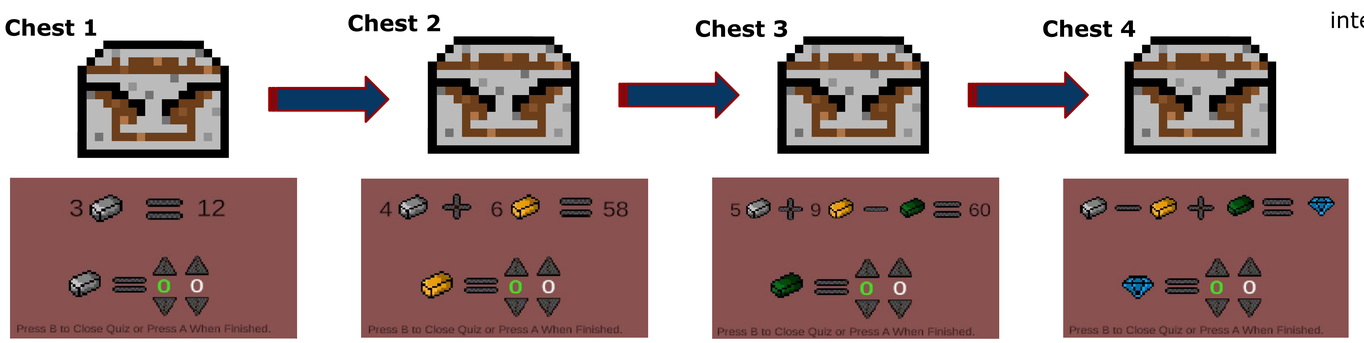


Image 1

Acknowledgement

I would like to recognize the Department of Education and Finish In Five program and team for giving me this opportunity. The Finish In Five was funded by the Department of Education.

Funding Agency:
U.S. Department of Education
Official Name of the Grant:
Title V Promoting Postbaccalaureate Opportunities for Hispanic Americans
Project Title: Finish in Five
PR Award Number: P031M210039

Reference

1. Fauquet-Alekhine, P. (2011). *Human or avatar: psychological dimensions on full scope, hybrid, and virtual reality simulators*. Ludoscience.
2. Brackeys. "Top down Movement in Unity!" *YouTube*, 11 Aug. 2019, youtu.be/whzomFgjT50?si=gh5t9PK8Zqsl7oia.
3. Chris's Tutorials. "How to Switch Levels with Trigger Area for 2D Rpgs in Unity." *YouTube*, 29 Dec. 2021, youtu.be/-7I0slJyi8q?si=3P46r3kEh9OGrGCu.
4. Garnet. "Random Map Generator in Unity! [Tutorial]." *YouTube*, 2 Jan. 2023, youtu.be/6B7vOnqpK_Y?si=GUn_1VKNxPUEX0Cd.
5. Mech Mine Games. "How to Use Unity's New Input System: Super Easy [Gamepad Joystick & Buttons!]." *YouTube*, 29 Sep. 2021, youtu.be/cWAdUrw2bXQ?si=9A9Xvqe2rP2v-5eo.
6. Coffee Cup Dev. "Unity Tutorial - How to Create a Fade Away Text Display." *YouTube*, 12 July 2023, youtu.be/WiUUW9RSa5Y?si=jxUEDnXHHunSaaGu.
7. Young, Eric. "Simple Math Game in Unity Tutorial." *YouTube*, 15 Dec. 2021, youtu.be/8NxRarssxnA?si=Uq5SxMsMfH9U65uU.
8. Image 1 - PDP Controller Wired Controller for PC and Xbox One, 2018

Unity C# math code base on reference [7]
Unity C# code for assets [2-6]