

[Battle Scene: Prototype: Debug Log]

[Things to do]

1. Fix bugs and complete battle system

[Bugs]

1. Amber cannot cast properly- is not clear wt happening inside
2. The Turn system doesn't progress properly

[Debug]

1. using Debug.Log
2. how to output array contents?: array[id].ToString()

[Video]

1. Tried {0,1,3} and should print the Text "Amber halved enemies' attack powers"
2. Input was successfully {0,1,3},but the text and output was not successful.
3. => problems on Battle.cs, probably comparing arrays / the Amber's action itself

[Improvements so far]

1. Combined cast.cs into game control.cs

[Debug Log]

1. amberInput: ok
2. turnID: ok
3. array comparing: not ok

[Solution]

1. convert int[] to int first and make the comparing simple!

[Video: Fixed ver.]

1. Success to attack an enemy

[Class GameControl]

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

public class GameControl : MonoBehaviour {
    //savedata vars
    public static int chapterID;
    public static int amberHP;
    public static int jasperHP; //when jasper is not in party,
HP == 0
    public static int ioliteHP; //when iolite is not in party,
HP == 0

    //temp vars
    public static bool casting;
    public static int[] circle;
```

```

private GameObject battle;

void Start () {
    //initialise when the game starts
    chapterID = 0;
    circle = new int[] {7,7,7};
    casting = false;
    battle = GameObject.Find ("Battle");
    amberHP = 150;
    if (battle.GetComponent<Battle> ().jasperJoined ==
true) {
        jasperHP = 150;
    } else {
        jasperHP = 0;
    }
    if (battle.GetComponent<Battle> ().ioliteJoined ==
true) {
        ioliteHP = 150;
    } else {
        ioliteHP = 0;
    }
}

void Update () {
}

void OnMouseDown() {
    if (Element.eleCount == 3) {
        circle = new int[] { Element.firstEle,
Element.secondEle, Element.thirdEle};
        Array.Sort (circle);
        Debug.Log (circle[0].ToString() +
circle[1].ToString() + circle[2].ToString());

GameObject.Find("Ele1").GetComponent<Symbol>().spriteID = 7;
GameObject.Find("Ele2").GetComponent<Symbol>().spriteID = 7;
GameObject.Find("Ele3").GetComponent<Symbol>().spriteID = 7;

        Element.EleCount (0);
        casting = true;
    }
}
}

```

```

[class Battle]
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class Battle : MonoBehaviour {

```

```

public int battleID;
public int enemiesNo;
public float craftTimeSetting; //can be longer by gathering
runes?
public string startText;

public bool jasperJoined;
public bool ioliteJoined;

public int enemy1HP;
public int enemy2HP; //input zero if the enemy doesnt
exist!
public int enemy3HP;
//i'm too lazy to make enemy class! let's just say this is
not RPG but adventure game =_=...

private int amberInput;
private int a;
private int b;
private int c;
private float amberSpeed;
private float craftTime;
private int turnID;
private int enemyHPTotal;
private int playerHPTotal;
private int jasperPW;
private int enemyPW;
private int damage;

private bool raining;
private bool damaging;

private int turn;
private int rainEnd;
private int damEnd;
private int damWho;

private bool next;
private bool craft;
private GameObject book;
private GameObject bui;
private GameObject bt;

void Start () {
    next = true; //the obj is tappble or not
    craft = false;
    turnID = 0;
    craftTime = craftTimeSetting;
    jasperPW = 1;
    enemyPW = 1;
    amberSpeed = 1;
    turn = 0;
    raining = false;
    damaging = false;

```

```

        book = GameObject.Find ("Book");
        bui = GameObject.Find ("BattleUI");
        bt = GameObject.Find ("BT");
        bt.GetComponent<Text>().text = startText;
        book.SetActive (false);
    }

    void Text (string txt) {
        bui.SetActive (true);
        bt.GetComponent<Text>().text = txt;
    }

    void Total () {
        enemyHPTotal = enemy1HP + enemy2HP + enemy3HP;
        playerHPTotal = GameControl.amberHP +
GameControl.jasperHP + GameControl.ioliteHP;
    }

    void Rain() {
        if (enemy3HP > 0) {
            enemy3HP = enemy3HP - 10;
        }
        if (enemy2HP > 0) {
            enemy2HP = enemy2HP - 10;
        }
        if (enemy1HP > 0) {
            enemy1HP = enemy1HP - 10;
        }
        if (GameControl.ioliteHP > 0) {
            GameControl.ioliteHP =
GameControl.ioliteHP - 10;
        }
        if (GameControl.jasperHP > 0) {
            GameControl.jasperHP =
GameControl.jasperHP - 10;
        }
        if (GameControl.amberHP > 0) {
            GameControl.amberHP = GameControl.amberHP
- 10;
        }

        if (rainEnd == turn) {
            raining = false;
        }
    }

    void Damage () {
        if (damWho == 3 && enemy3HP > 0) {
            enemy3HP = enemy3HP - 10;
        } else if (damWho == 2 && enemy2HP > 0) {
            enemy2HP = enemy2HP - 10;
        } else if (damWho == 1 && enemy1HP > 0) {
            enemy1HP = enemy1HP - 10;
        } else {

```

```

        damaging = false;
    }

    if (damEnd == turn) {
        damaging = false;
    }
}

void Turn () {
    next = false;
    Debug.Log ("TURN" + turnID.ToString ());

    //Amber's turn
    if (turnID == 0) {
        bui.SetActive (false);
        book.SetActive (true);
        turnID = turnID++;
        craft = true;

        //Jasper's turn
    } else if (turnID == 1) {
        Text ("Jasper attacked an enemy!");
        if (enemy3HP > 0) {
            enemy3HP = enemy3HP - 50 *
jasperPW;

        } else if (enemy2HP > 0) {
            enemy2HP = enemy2HP - 50 *
jasperPW;

        } else {
            enemy1HP = enemy1HP - 50 *
jasperPW;

        }

        jasperPW = 1;
        turnID = turnID++;
        next = true;

    } else if (turnID == 2) {
        if (playerHPTotal <= 210) {
            Text ("Iolite healed everyone!");
            //change the rate 0.5 if the game
            balance went wrong
            GameControl.amberHP = (150 -
GameControl.amberHP) / 2 + GameControl.amberHP;
            GameControl.jasperHP = (150 -
GameControl.jasperHP) / 2 + GameControl.jasperHP;
            GameControl.ioliteHP = (150 -
GameControl.ioliteHP) / 2 + GameControl.ioliteHP;
        } else if (GameControl.amberHP <= 30) {
            GameControl.amberHP = 150;
        } else if (GameControl.jasperHP <= 30) {
            GameControl.jasperHP = 150;
        } else if (GameControl.ioliteHP <= 30) {
            GameControl.ioliteHP = 150;
        }
    }
}

```

```

        } else {
            Text ("Iolite doubled Jasper's
attack.");
            jasperPW = jasperPW * 2;
        }

        turnID = turnID++;
        next = true;
    } else if (turnID >= 3) {
        if (GameControl.ioliteHP > 0) {
            Text ("An enemy attacked
Iolite!");
            GameControl.ioliteHP =
GameControl.ioliteHP - 30 / enemyPW;
        } else if (GameControl.jasperHP > 0) {
            Text ("An enemy attacked
Jasper!");
            GameControl.jasperHP =
GameControl.jasperHP - 30 / enemyPW;
        } else {
            Text ("An enemy attacked Amber!");
            GameControl.amberHP =
GameControl.amberHP - 30 / enemyPW;
        }

        enemyPW = 1;
        turnID = turnID++;
        next = true;
    }
}

void Update () {
    if (craft == true){
        craftTime = craftTime - 1 *
Time.deltaTime;

        if (craftTime <= 0) {
            Text ("Amber missed casting!");
            craftTime = craftTimeSetting;
            book.SetActive (false);
            next = true;
        } else if (GameControl.casting == true) {

            a = GameControl.circle [0];
            b = GameControl.circle [1];
            c = GameControl.circle [2];
            Debug.Log (a.ToString() +
b.ToString() + c.ToString());

            amberInput = a * 100 + b * 10 + c;

            if (amberInput == 13 || amberInput
== 125) {
                enemyPW = 2;
            }
        }
    }
}

```

```

enemies' attack powers!");
                                Text ("Amber halfed
                                } else if (amberInput == 15) {
                                if (enemy3HP > 0) {
                                enemy3HP =
                                Text ("Amber
                                } else if (enemy2HP > 0)
                                enemy2HP =
                                Text ("Amber
                                } else {
                                enemy1HP =
                                Text ("Amber
                                }
                                } else if (amberInput == 35) {
                                if (enemy3HP > 0) {
                                enemy3HP =
                                }
                                if (enemy2HP > 0) {
                                enemy2HP =
                                }
                                if (enemy1HP > 0) {
                                enemy1HP =
                                }
                                if (GameControl.ioliteHP
                                > 0) {
                                GameControl.ioliteHP = GameControl.ioliteHP - 50;
                                }
                                if (GameControl.jasperHP
                                > 0) {
                                GameControl.jasperHP = GameControl.jasperHP - 50;
                                }
                                if (GameControl.amberHP >
                                0) {
                                GameControl.amberHP = GameControl.amberHP - 50;
                                }
                                } else if (amberInput == 34 ||
                                amberInput == 45) {
                                damage = Random.Range
                                (-100, 0);

```

```

(debug.ToString ());

enemy3HP + damage;

enemy2HP + damage;

enemy1HP + damage;

enemy3HP - 50;
{
enemy2HP - 50;

enemy1HP - 50;

GameControl.amberHP - 25;

{

<= GameControl.jasperHP) {
GameControl.amberHP = 150;
(GameControl.jasperHP <= GameControl.ioliteHP) {

Debug.Log

if (enemy3HP > 0) {
    enemy3HP =
}
if (enemy2HP > 0) {
    enemy2HP =
}
if (enemy1HP > 0) {
    enemy1HP =
}
} else if (amberInput == 124) {
    amberSpeed = 2;
} else if (amberInput == 135) {
    if (enemy3HP > 0) {
        enemy3HP =
    } else if (enemy2HP > 0)
        enemy2HP =
    } else {
        enemy1HP =
    }
    enemyPW = 2;
} else if (amberInput == 235) {
    GameControl.amberHP =

    if (enemy3HP > 0) {
        enemy3HP = 0;
    } else if (enemy2HP > 0)
        enemy2HP = 0;
    } else {
        enemy1HP = 0;
    }
} else if (amberInput == 245) {
    jasperPW = 0;
    enemyPW = 3;
} else if (amberInput == 456) {
    if (GameControl.amberHP

} else if

```



```

GameControl.jasperHP = 150;
GameControl.ioliteHP = 150;
{
    } else {
    }
    } else if (amberInput == 356) {
        raining = true;
        rainEnd = turn + 3;
    } else if (amberInput == 256) {
        damaging = true;
        damEnd = turn + 5;
        if (enemy3HP > 0) {
            damWho = 3;
        } else if (enemy2HP > 0)
            damWho = 2;
        } else {
            damWho = 1;
        }
    } else if (amberInput == 26) {
        jasperPW = jasperPW * 2;
    } else {
        Text ("Amber failed to
cast!");
    }
    GameControl.circle = new int[]
    GameControl.casting = false;
    craftTime = craftTimeSetting *
    amberSpeed;
    book.SetActive (false);
    next = true;
}
}
//avoid minus HPs
if (GameControl.amberHP < 0) {
    GameControl.amberHP = 0;
}
if (GameControl.jasperHP < 0) {
    GameControl.jasperHP = 0;
}
if (GameControl.ioliteHP < 0) {
    GameControl.ioliteHP = 0;
}
if (enemy3HP < 0) {
    enemy3HP = 0;
}
if (enemy2HP < 0) {
    enemy2HP = 0;
}
if (enemy1HP < 0) {

```

```

        enemy1HP = 0;
    }
}

void OnMouseDown () {
    if (next == true) {
        Total ();

        //skipping dead or non-existing characters
        if (turnID == 1 && GameControl.jasperHP <=
0) {

            turnID = turnID++;
        }

        if (turnID == 2 && GameControl.ioliteHP <=
0) {

            turnID = turnID++;
        }

        if (turnID == 4 && enemy2HP <= 0) {
            turnID = turnID++;
        }

        if (enemyHPTotal <= 0) {
            //Winning Text
            //
Application.LoadLevel("narrativeX");
        } else if (playerHPTotal == 0 ||
GameControl.amberHP <= 0) {
            Destroy (this); //only for
prototype!

            //GameOver text
            //Application.LoadLevel("start");
        } else if (enemiesNo < turnID - 2) {
            turnID = 0;
            turn = turn++;
            if (raining == true) {
                Rain ();
            }
            if (damaging == true) {
                Damage ();
            }
            Turn ();
        } else {
            Turn ();
        }
    }
}
}
}

```