

[class Battle: Prototype 03: 16.02.17]

[Initial plan]

1. complete battle incl.Amber's circle crafting
2. try unity scene

[Problems & Solutions]

1. continuous damages – using turn count and bool

[Thoughts]

1. some of Bea's game design is impossible to realise on Unity for my skills...T T

[completed class Battle prototype]

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

public class Battle : MonoBehaviour {
    public int battleID;
    public int enemiesNo;
    public float craftTimeSetting; //can be longer by gathering
runes?

    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 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;

    void Start () {
        next = true; //the obj is tappble or not
```

```

        craft = false;
        turnID = 0;
        craftTime = craftTimeSetting;
        jasperPW = 1;
        enemyPW = 1;
        amberSpeed = 1.0;
        turn = 0;
        raining = false;
        damaging = false;
    }

    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;

    //Amber's turn
    if (turnID == 0) {
        craft = true;

        //Jasper's turn
    } else if (turnID == 1) {
        if (enemy3HP > 0) {
            //text"Jasper attacked enemy3"
            enemy3HP = enemy3HP - 50 *
jasperPW;

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

        } else {
            enemy1HP = enemy1HP - 50 *
jasperPW;

        }

        jasperPW = 1;
        next = true;

    } else if (turnID == 2) {
        if (playerHPTotal <= 210) {
            //text"Iolite used healing spell
for all"
            //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;
        }
    }
}

```

```

        next = true;
    } else if (turnID == 3) {
        if (GameControl.ioliteHP > 0) {
            GameControl.ioliteHP =
GameControl.ioliteHP - 30 / enemyPW;
        } else if (GameControl.jasperHP > 0) {
            GameControl.jasperHP =
GameControl.jasperHP - 30 / enemyPW;
        } else {
            GameControl.amberHP =
GameControl.amberHP - 30 / enemyPW;
        }

        enemyPW = 1;
        next = true;
    }
}

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

        turnID = turnID++;

        //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) {
            Destroy (this); //only for
prototype!

            //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 ();
    }
}

}

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

        if (craftTime <= 0) {
            craftTime = craftTimeSetting;
            next = true;
        } else if (GameControl.casting == true) {

            amberInput = GameControl.circle;

            if (amberInput == new int[] { 0,
1, 3 } || amberInput == new int[] { 1, 2, 5 }) {
                enemyPW = 2;
            } else if (amberInput == new int[]
{ 0, 1, 5 }) {

                if (enemy3HP > 0) {
                    enemy3HP =
enemy3HP - 50;
                }
                {
                    if (enemy2HP > 0)
                        enemy2HP =
enemy2HP - 50;
                }
                {
                    if (enemy1HP > 0)
                        enemy1HP =
enemy1HP - 50;
                }
            }
        } else if (amberInput == new int[]
{ 0, 3, 5 }) {

            if (enemy3HP > 0) {
                enemy3HP =
enemy3HP - 50;
            }
            {
                if (enemy2HP > 0) {
                    enemy2HP =
enemy2HP - 50;
                }
            }
            {
                if (enemy1HP > 0) {
                    enemy1HP =
enemy1HP - 50;
                }
            }
        }
    }
}

```

```

enemy1HP - 50;
}
> 0) {
    if (GameControl.ioliteHP
GameControl.ioliteHP = GameControl.ioliteHP - 50;
}
> 0) {
    if (GameControl.jasperHP
GameControl.jasperHP = GameControl.jasperHP - 50;
}
0) {
    if (GameControl.amberHP >
GameControl.amberHP = GameControl.amberHP - 50;
}
} else if (amberInput == new int[]
{ 0, 3, 4 } || amberInput == new int[] { 0, 4, 5 }) {
    damage = Random.Range
(-100, 0);
    if (enemy3HP > 0) {
        enemy3HP =
enemy3HP + damage;
    }
    if (enemy2HP > 0) {
        enemy2HP =
enemy2HP + damage;
    }
    if (enemy1HP > 0) {
        enemy1HP =
enemy1HP + damage;
    }
} else if (amberInput == new int[]
{ 1, 2, 4 }) {
    amberSpeed = 1.2;
} else if (amberInput == new int[]
{ 1, 3, 5 }) {
    if (enemy3HP > 0) {
        enemy3HP =
enemy3HP - 50;
    }
    {
        } else if (enemy2HP > 0)
            enemy2HP =
    } else {
        enemy1HP =
    }
    enemyPW = 2;
} else if (amberInput == new int[]
{ 2, 3, 5 }) {

```

```
GameControl.amberHP - 25;
```

```
{
```

```
{ 2, 4, 5 }) {
```

```
{ 4, 5, 6 }) {
```

```
<= GameControl.jasperHP) {
```

```
GameControl.amberHP = 150;
```

```
(GameControl.jasperHP <= GameControl.ioliteHP) {
```

```
GameControl.jasperHP = 150;
```

```
GameControl.ioliteHP = 150;
```

```
{ 3, 5, 6 }) {
```

```
{ 2, 5, 6 }) {
```

```
{
```

```
{ 0, 2, 6 }) {
```

```
attack"
```

```
GameControl.amberHP =
```

```
if (enemy3HP > 0) {  
    enemy3HP = 0;
```

```
} else if (enemy2HP > 0)
```

```
    enemy2HP = 0;
```

```
} else {  
    enemy1HP = 0;
```

```
}
```

```
} else if (amberInput == new int[]
```

```
    jasperPW = 0;
```

```
    enemyPW = 3;
```

```
} else if (amberInput == new int[]
```

```
    if (GameControl.amberHP
```

```
    } else if
```

```
    } else {
```

```
    }
```

```
} else if (amberInput == new int[]
```

```
    raining = true;
```

```
    rainEnd = turn + 3;
```

```
} else if (amberInput == new int[]
```

```
    damaging = true;
```

```
    damEnd = turn + 5;
```

```
    if (enemy3HP > 0) {
```

```
        damWho = 3;
```

```
    } else if (enemy2HP > 0)
```

```
        damWho = 2;
```

```
    } else {  
        damWho = 1;
```

```
    }
```

```
} else if (amberInput == new int[]
```

```
    jasperPW = jasperPW * 2;
```

```
} else {  
    //text "Amber missed her
```

```
}
```

```

    GameControl.circle = new int[]
{ 7, 7, 7 };
    GameControl.casting = false;
    craftTime = craftTimeSetting *
    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;
}
}
}

```

[next to do]

1. combine with graphics on Unity