MyRANCHSTORY

NOTE

BY---Yamil-Z

**Asset** ：

PolygonFantasyCharacters From 淘宝

PolygonFarm From 淘宝

StarterAssets From UnityStore

Suntail Village From 淘宝

**Tool** ：

PolyBrush From UnityStore

**<--------------------------------------------------------->**

10.27

**DAY 1:**

**解决了第三人称人物移动问题**

采用了StarterAssets的第三人称解决方案，查看并了解该方案的解决原理和解决过程：

解决算法为标准第三人称人物移动和镜头跟随算法，算法思路：通过InputSystem获得玩家输入的horizontal和vertical向量，通过Mathf.Atan获得两者角度，然后运用该角度与相机角度进行运算，得出玩家相对相机角度，然后进行旋转，该过程运用了Mathf.Lerp进行非线性插值，使玩家转身更加的圆润

相机采用Cinemachine的虚拟相机，采用3d环绕，可以通过玩家鼠标输入旋转值

尝试了解了unity2020的InputSystem

**解决了人物移动和跳跃的Animator**

采用BlendTree对跳跃过程和运动切换进行了动作切换的圆滑

**Summary：**

Unity2020封装了新了InputSystem，使得新的unity2020资源不再对2020以下的版本适用

对Unity2020缺乏了解，新的功能，新的界面

在源码理解中，有了更多的心得，对下次阅读他人代码提供了宝贵经验

**<--------------------------------------------------------->**

10.28

**DAY 2:**

对昨天的人物移动和动画进行了进一步的调试

**场景搭建：**

进行了布局图的查找和搭建的艺术的学习

使用polybrush对地形进行了简单的绘制

对农场进行了搭建，并把Tag的框架提前做好，给搭建的物体进行Tag分类

**Summary：**

场景搭建很难，比打代码还难！

使用polybrush的顶点绘制和贴图绘制，必须使用polybrush的material，并且在安装polybrush的时候，如果项目使用了urp渲染管道，必须安装polybrush的upr material

**<--------------------------------------------------------->**

10.29

**DAY 3:**

**熟悉unity2020InputSystem**

从2019新加的输入系统，在2020会默认使用

新系统的好处就是帮你整合和包装了很多用户输入的方式，这样的好处就是，更加的系统，在加入新的用户输入时，不用翻查代码，就可以很快的完成你想完成的工作，这样也有利于旧项目想对玩家操作优化的工作减轻了负担

坏处就是，这是一个新系统，熟悉起来比较慢，新的事物难以让一些在常在旧版本的人认可。

**开关门事件的解决**

完成了开关门交互事件的代码

在这个代码中，加入了新的InputSystem操作，并用C#自带event进行了事件的注册添加，这样便于后面新加入玩家交互后，代码更加的简洁易懂

开关门逻辑处理，采用了玩家发信号，具体操作门上的单独的代码自己执行

在门自己执行开关门操作时，因为模型旋转轴在模型中央，所以基于模型设置了子物体旋转点，并通过GetComponentsInChildren获得其旋转点，减少手动拖拽的重复工作。

**玩家交互的单例模式**

在本项目中，把负责玩家交互的代码运用了单例模式管理，到后面新加入时间系统等等，会添加成泛型单例，节约人工重复的工作。

**进一步完善人物移动细节和场景模型顶点网格优化处理**

完善了人物移动中和模型网格的碰撞，减少了玩家移动时的卡Bug，同时减小了地形模型的凹凸，让玩家行动更加的丝滑

**Summary：**

因为今天都是课，所以没有做什么东西，主要去看了新的InputSystem

在代码书写上，进行了简化，尽可能提高代码效率

管理模式非常好用，但现在会的太少

不要局限于你知道的，要多接触你不知道的，要学会学习和接纳新的事物

**<--------------------------------------------------------->**

10.30

**DAY 4:**

**解决了游戏玩法核心的基础系统：时间系统**

解决了以Time类为基础的游戏时间系统，后期会以此为基础，创建我们的节日活动系统，和作物动物成长系统，和游戏里的日夜交替和季节系统，同样后面会加入到泛型单例。

**交互提示**

增加了玩家的交互提示，在可以交互的地方，会提示玩家交互，目前只有开门会有提示，后面会加上种植，浇水，砍伐，收获等交互和交互提示。

**玩家界面的UI**

构建了玩家主界面的UI框架，目前显示时间和交互提示，后期会加上季节，天气，主界面物品栏等。

**Summary：**

目前的交互方式是从屏幕中间射出射线（有一定的距离），判断检测到的物体是否可以交互，但这个方式对第三人称的操作来说，选取到可以交互的物体的条件有点苛刻，想尝试新的方法（如给物体新加一个trigger，但是如果两个可交互物体离得比较近，玩家站在了两个物体中间，就很难判断哪个先触发），但目前没有什么好想法。

出现了莫名其妙的报错，是unity自己的报错，大概意思是：某某某没有实例化，不懂是啥，捣鼓了半天还是报错。

**<--------------------------------------------------------->**

10.31

**DAY 5:**

**解决了多个交互提示的问题**

今天在项目中新加了种植的交互提示，但和开门的交互提示的判断冲突了，改用了int型判断做一个类似并联的功能，这种模式可以应用在以后更多交互提示的出现

**种植的脚本模板**

完成了种植的脚本模板，在完成背包系统后，会进一步完善这个脚本，现在已经做好了框架

**Summary：**

今日划水，大部分时间花在了改昨天说的报错上面，应该是unity自己的bug

下一个功能是背包系统，为种植和砍伐和浇水做基础

今天复盘了一遍这个项目前几天做的东西，顺手完善了一下之前的代码

**<--------------------------------------------------------->**

11.1-2

**DAY 6,7:**

**背包系统解决方案**

第六天没做什么，所以和第七天一起写了

这两天把背包系统做了出来，但是目前在inputsystem上还是出了问题，还是没有去弄清楚他的机制

背包系统采用了MVC框架，这个框架，其实在背包系统之前，我已经用了无数次了，就是一个框架和逻辑的问题

背包系统采用了Grid Layout Group进行UI布局的操作，采用ScriptableObject进行数据的存储。

**Summary：**

目前背包系统还没有完全完工，基本功能做了出来，但还是再inputsystem上出了错误，感觉还是没有把这个弄清楚，对后来的交互操作会埋下非常大的隐患，所以计划明天弄清楚这个inputsystem，并把玩家的交互设计的更完善些。

**<--------------------------------------------------------->**

11.3

**DAY 8:**

**Input System**

首先来更正一下之前的错误，这并不是一个新系统，而是在2016年就开始有准备的系统，当时各种游戏媒介如，ps4，Xbox，switch等等，输入系统都不同，每个写一遍非常麻烦，更改起来也麻烦，为了更加简便和容易的进行各个系统之间操作的更换，unity在2016推出了试验版的Input System，在2020进行了正式版的发布，这个input system 会和老版本的input 进行互相排斥，你只能用一个。

其次，这个输入系统，采用的是添加和订阅事件的方式，你可以通过添加事件的方式，来操作你的输入。可以使用unityEvent，也可以使用C#自带的Event，也可以只让他Send Messages，然后像旧版本的一样，操作的你的输入。

总的来说，这个系统，操作起来更方便，也更容易，在各个系统切换，代码的效率会提高很多相比于旧版本

**完善背包系统**

完善了昨天的背包系统，让背包内的物品的更新看起来更加丝滑

**一些Bug修复**

修复了在拾取物品时UI提示未能正常关闭

修复了背包在一次按键一帧内打开多次的情况

**作物生长的逻辑**

完成了作物生长了逻辑，目前只做了3个形态的作物，到后期会根据不同的作物，制作多种形态。

**Summary：**

这个正式版的输入系统其实并没有我看起来的那么难，之前觉得难是因为不了解，对于陌生的恐惧。

接下来的计划是，完成种菜交互，和UI交互提示的升级，对于不同交互的不同提示。

在今天的进程中，再一次深入了解了StarterAssets，通过这个外部资源包去了解新的输入系统，并通过修改这个资源包的代码，达成自己想要的效果。可以说，这个资源包是一个教程资源包，带你了解新的输入系统，里面的代码也很简单。

**<--------------------------------------------------------->**

11.4

**DAY 9:**

**种植的实现**

完成了种植的实现，但是目前浇水交互还没有实现，所以在耕地的干湿属性上还没有具体实现，打算在工具类完成后，再进行实现

种植交互中，发出命令的为主角，实行种植交互的为耕地本身，种子类自带他的各个阶段的形态和他的生长周期

在耕地类中进行种子的成长，到达不同阶段，获取种子类的不同形态，达成切换形态的目的

**主界面背包的实现**

完成了一个非常重要的游戏方式，在主界面，无须开启背包，通过Q和E切换主角手上的工具或者种子等等，以此达成不同种类物品的不同交互。比如花洒浇水，种子种植，斧头砍伐等等  
后面找到了工具的模型，会把工具显示在主角手上

**Summary：**

下一个目标是，完成工具类代码，和找到相应的素材UI

目前项目中的UI很少，很单调，并不能和模型联系在一起，所以还需要找到更多形式丰富的UI，或者自己找别人画

找到竟可能多的交互动画，目前项目中交互动画稀缺

**<--------------------------------------------------------->**

11.5

**DAY 10:**

**完善种植系统**

项目有了更完善的种植系统，更加多样化的工具，同时代码方面简短了不少，让不同的交互，逻辑清晰的缩短在一个方法里，减少了代码的重复性

经过测试，目前的作物成长系统功能良好

目前种植和浇水，收获的交互动画已经就位，近期有打算把这些交互的动画完成

在完善种植系统的过程中，发现了几个bug

背包物品在种植减少后变成负数依然能用（忘记给他加数量的限制了）

在加上数量限制后，出现了频率不高，忽高忽低的列表下标越界的提醒，当时判断是在切换界面背包时，一个周期的sub在到底负数，或者超过list上限时没能及时归零导致的，最后在把此属性传递给主角的方法上加上if限制后解决

**完善物品UI**

给物品换上了更精美的UI，同时发现，界面UI和背包物品UI缺少背景，看起来有点单调，预计近期完善

**Summary：**

今日主要是修复bug和完善之前的功能，找到了UI图片的资源网站，许多交互动作也找到了（又给自己挖了个坑）

近期打算完成的：

UI的全面升级

缺失模型的寻找

交互动作的完善

时间系统的时间流逝速度的调整

商城系统的制作

**<--------------------------------------------------------->**

11.5

**DAY 10:**

**交互动画系统的解决**

今日完成了整个交互动画系统的框架，和浇水交互动画的具体实现，并在这个过程中，禁止人物的移动。禁止人物移动的具体解决是：

在浇水动画上，挂载脚本，但这个脚本属性赋值并不能像之前的脚本一样直接拖拽，所以我在人物移动的脚本上，进行了该脚本的注册，从而在外部脚本进行赋值

在这个交互动画系统的完成中，发现了一些我无法挽回的问题

1：受限制于人物骨骼，我本身的人物骨骼缺少一些下载来的动画的骨骼的点，会导致我人物在执行外部来的动画时，有些部位的动画不太自然，解决办法是重新找一个骨骼点更加完善的骨骼，但目前没有这种想法，大局影响不大

2：因为是个人创作，所以在整个项目的策划上，没有精细的计划，导致目前项目的功能和代码，是走一步看一步，代码没有整体性

**Summary：**

今日大部分时间都在弄这个动画系统，在这个过程中，发现自己面向对象四象越来越好了，感觉自己有进步，代码逻辑也越来越好，很少能出现写出来的功能实现不了的情况

一些工具的模型还是找不到

下个功能是做商城

**<--------------------------------------------------------->**

11.6

**DAY 11:**

啥都没做……

**<--------------------------------------------------------->**

11.7

**DAY 12:**

啥都没做……

**<--------------------------------------------------------->**

11.8

**DAY 13:**

**BUG修复**

修复了种子在消耗到0时，图标消失，list也删除，但是实际的角色手中还有残存的种子信息。解决办法是，在角色交互控制脚本中，当种子消耗到0时，更新界面背包的sub，并且更新一次背包内容

**<--------------------------------------------------------->**

11.9

**DAY 14:**

**完善角色属性背包面板**

完善了属性背包面板的逻辑操作和UI显示

完成了角色信息类的创建和显示

完善了属性面板

**Summary：**

这几天有点偷懒，没做什么。打算先不做商城系统，因为商城系统你不光有买东西，还有卖东西，打算先把收获给做了

**<--------------------------------------------------------->**

11.10

**DAY 15:**

**收获交互的完成**

完成了收获的交互

在完成收获的交互时，发现收获完后，背包主背包并没有及时更新背包，反倒是界面背包及时的更新了。

所以在每次打开背包时，我都调用了一次背包更新的函数，让他能够更新及时，不过，如果你背包东西多，有可能打开就会有卡顿，不过是个游戏都有点卡顿啊，不管了

**背包界面的内部界面的交换逻辑**

完成了内部界面的交互逻辑，目前只有2个分栏，但是框架支持无数个分栏，为以后添加更多的分栏（地图，好感度，拥有的动物等等……）创造了便捷方法

**Summary：**

目前发现的几个问题：

1：目前只有浇水的交互动作，收获的交互动作我也是打算做的，但是种植的交互动作就不做了，感觉这样会拖慢游戏的节奏

2：感觉代码有点乱，往会看会觉得我自己的代码很“高大上”

3：场景有点丑，东西有点少

接下来打算做的功能：

商城的买卖系统

畜牧业

NPC

新的场景（小镇，集市，朋友的家）

**<--------------------------------------------------------->**

11.11

**DAY 16:**

校运会划水

**<--------------------------------------------------------->**

11.12

**DAY 17:**

继续校运会划水

**<--------------------------------------------------------->**

11.13

**DAY 18:**

**商城买卖系统框架的搭建**

完成了商城买卖系统和商人系统的框架搭建

在这个完成过程中，突发奇想，想让我的商人们，不一定天天都在上班，可以休息的时候，让他们在小镇上休息，随便走，所以，他们只有在商店里，是商人，下班了，就不是商人了

商人的物品，不设置数量上限，无限制的卖给玩家（反正玩家也不需要这么多），商人的物品和玩家的物品的数据库不同，商人的物品的数据库里有玩家的物品数据库，以方便买卖时，数据和玩家数据互通。

目前的商人系统比较单调，货物全年都是一样的，目前不考虑增加多样性，因为感觉游戏玩法重点不在商人这里，如果做的话会拖延我的进度

**Summary：**

准备考试了，有可能这个游戏的进度要变慢了。

**<--------------------------------------------------------->**

11.14

**DAY 19:**

**买东西功能完成**

完成了向商人买东西的功能

**Summary：**

在完成这个功能的过程中，把之前的代码重新整合了一遍，开启了GameManager脚本，控制在某些UI打开时，玩家失去对角色的控制权

在之前的F按键交互事件上，重新进行了逻辑更改，更具有多样性。同时把打开商城的事件也放进这个Even中。

下个是做卖东西的功能，然后把畜牧业做了，然后NPC，然后好好的布置场景，给之前做的功能增加实际的游戏内容。

**<--------------------------------------------------------->**

11.15

**DAY 20:**

**探索**

这是自学的一个重大并且浪费时间的环节，没有人引导或者教你，探索非常的浪费时间，但是比别人教你更加的令你印象深刻

探索NavMesh的使用：

因为要做畜牧，所以要做动物的AI，包括之后的NPC AI，所以了解一下NavMesh

AI框架的搭建，初步探索了动物AI框架的搭建

**Summary：**

目前通过了解其他的案例，通过其代码和说明书了解用法，和具体的思路

**<--------------------------------------------------------->**

11.16

**DAY 21:**

**目前畜牧业的成果**

通过昨天案例的分析和使用手册的阅读，思路已经有了

NavMesh也了解的差不多了，已经做出了动物区域内的自动寻路，但效果不太理想，目前正在改善

**遮罩剔除**

主角在被其他图层遮挡后，会有特殊的sharder显示给玩家，以此达到玩家不会看不到主角的效果

本来是想弄相机绕过障碍物的（思路是：在摄像机和玩家之间生成几个点，在这几个点中，选取摄像机能看到玩家的点，否则选取最近的点），但是上学期做过了，换个新鲜的。

**Summary：**

目前进度不太理想，有点慢，预计2个月做完的，目前看来不太可能，一个人做，资源和美术从网上找，加上很多功能需要自己摸索，所以进度有点慢

**<--------------------------------------------------------->**

11.17

**DAY 22:**

**AI随机移动的探索**

本来是想用一下AStar算法的，但是鉴于这个游戏并不是策略性的简单寻路，AStar算法基于格子来遍历计算行走权重，算最优路径很好用，但是算法复杂度较高，这个游戏的牧场比较平整，所以打算用NavMesh拐角点算法。NavMesh通过计算烘焙的地图的边缘顶点，当到达边缘顶点时，绕过该点，所以叫他拐角点算法

**AI随机移动的完成**

完成了AI随机移动的功能（但是在某些地方还是会卡住不动，先不管），采用了NavMesh的方法，状态机切换框架也搭好了，明天再弄上去

随机移动的方法：

通过获得已经烘焙的地图的随机网格，再在网格中随机获得一个点得到随机网格点

但是地图的烘焙不是连续的，一旦他随机到另外一个区域，他就会卡住，采用了SamplePathPosition来判断该条路径的前方是否可以到达，否的话回到原点。

**Summary：**

接下里完成状态机后，做玩家与动物的交互

在本次探索中，学会了NavMesh的使用和其他一些没用到的属性和方法，为下次使用奠定了基础

**<--------------------------------------------------------->**

11.18

**DAY 23:**

没做

**<--------------------------------------------------------->**

11.19

**DAY 24:**

**对目前功能的思考**

昨天达成了随机移动的功能，今天用一只猫试了一下，发现，如果随机获取已经烘焙的网格上的一个随机点的话，他很可能会在这个烘焙区域长距离的跑，这不符合动物的风格，一般动物休闲的时候，都是小范围的走动和休息，不会大范围的跑动，所以取消了之前随机取网格点的办法， 换成动物本身小范围内的随机移动（以动物自身为圆心随机取周围的点，判断是否是可以走的的范围），这个可以让动物更加的灵活，不会长距离的跑动（虽然这个功能会让我的代码B格下降）

**<--------------------------------------------------------->**

11.20

**DAY 24:**

**动物随机移动的完善**

昨天做动物的分区域随机移动的时候，出现了自定义区域无法移动的bug，经过这两天的排查终于找到了原因:

在随机移动判定的时候，为了让随机点选到可以移动的区域，采用了NavMesh.SamplePosition，这个方法可以判定你选的点是否可移动。但在这个方法，要选择你认为可以移动的区域，但目前我还不知道我自定义的区域的INT值，所以把全部区域都选上了

**其他**

进行了一些场景的布置

**修复了一些BUG**

完善了一些地方的NavMesh的烘焙

修复了关掉商城后无法移动的问题

**Summary：**

2个星期才做这点东西，我好菜啊

**<--------------------------------------------------------->**

11.21~26

**DAY 30:**

**完成了玩家岛屿的场景布置**

渲染管线中的关于阴影：

Shadows 中的 Distance选项，数值代表靠近多少距离才会渲染出阴影，减少会有效的降低性能消耗

Cascades 分层渲染 可以通过修改层数，来达到近处真实渲染，远处虚拟

迷雾：

Lighting里的fog选项，选择linear选项，可以调节能见度和颜色

：

[RequrieComponent(typeof( T ))]:自动为物体添加 T组件