**第二次作业**

1. 简述NavMesh的设置方法

首先给他添加一个Nav Mesh Agent组件，设置Radius、Height和Stopping Distance。在Windows菜单下找到Navigation窗口。计算导航网格，设置假设的物体大小，物体能上的台阶高度，物体能爬的坡度，物体能跳下的高度，物体能跳的距离，点击Bake烘焙导航网格。最后写一个脚本，在Start或者Awake里面找到Nav Mesh Agent组件，在Update里面运用SetDestination设置目标地点即可。

2. 简述Nav Mesh Agent属性参数的使用方法

1. Radius：导航代理的半径
2. Speed ：这个属性代表这个导航网格代理寻路时可以达到的最大速率
3. Acceleration ：加速度，表示代理的速度从0加速到Speed时的最大的加速度
4. Angular Speed ：最高的角速度
5. Stopping distance：制动距离，当代理据目的地的距离小于这个值时开始减速
6. Auto Traverse OffMesh Link ：自动移动并关闭OffMeshLinks
7. Auto Repath：自动重新寻路，若发现有路径已失效，那么它将获得新的路径
8. Height：导航代理的高度
9. Base Offset：基本偏移，我们可以通过调整这个变量来调整代理自身的包围盒
10. Obstacle Avoidace Type：代理躲避的水平，一般选默认的High Quality
11. NavMesh Walkable：导航网格代理可以通过的网格层类型

3.简述寻路过程中路网烘焙过程

首先选择需要行走的路块进行烘焙（点击最上方的Windows下拉菜单的 Navigation），选择完成后，点击右下角的Bake，即可烘焙完成。在需要寻路的人物上，加上自动寻路Nav Mesh Agent组件，选中需要烘焙的路面以及物体，打上红勾，可以行走的区域选择Walkable，不可行走的选择Not Walkable。接下来再写个简单的小脚本，就可以让我们的人物动起来了。

4.对于寻路过程中的障碍物绕行应该怎样处理？

通过 Navmesh 插件实现主角遇到障碍物时自动绕行效果。执行 File→Save Scene as 命令，将 Navigation 场景另存为 Obstacle 场景新建一个障碍物，将其放置在主角的前方，执行 Component→Navigation→Nav Mesh Obstacle 命令添加 Nav Mesh Obstacle 组件，则主角就会绕过障碍物并到达终点

5. Nav Mesh Agent组件的代理器移动到给定目标点需要利用哪几个函数？该函数有几个参数？其含义分别是什么？

函数：Transform.Translate

参数：position 位置

Rotation 旋转

Scale 缩放