问题：

AI的 move to一直不在地上到底什么几把玩意？

已解决：UNavMovementComponent的GetActorFeetLocation()导致，默认返回updatecomponent的location-actor bound的Z，如果Actor的mesh component的原点不在中心，就会产生偏差导致初始位置偏低，移动时往天上飞。解决方法，重载GetActorFeetLocation函数

Move to到了位置一直不结束。

已解决：没有触发OnActorBump，改Actor的collision；改完后没移动到就停止：取消勾选moveto节点的重叠半径选项

C++类细节面板是空的

已解决：重新reparent一下，或者试试重新编译

Enhanced input 输入值为0

已解决：不要加其他触发器，默认的就是1

函数参数有原生C++型的时候函数用UFUNCTION标记，编译报错Unable to find 'class', 'delegate', 'enum', or 'struct' with name …

已解决：

开始以为是头文件的问题，后来发现把UFUNCTION去掉就能过了，是反射不支持，以下是GPT的回答：

在 Unreal Engine C++ 中，你可以使用 UFUNCTION 标记一个函数，但它的参数和返回类型需要遵循一定的规则。特别是，Unreal Engine 反射系统（Reflection System）对函数参数类型有严格的限制。基本原则是，函数参数应该是引擎能够识别和处理的类型。

支持的参数类型

Unreal Engine 反射系统支持以下类型的参数：

基本类型：如 int32, float, bool 等。

FString：字符串类型。

引擎的结构体类型：如 FVector, FRotator 等。

引擎的对象类型：如 AActor\*, UObject\* 及其子类。

TArray：引擎的数组类型。

不支持的原生C++类型

原生C++类型（如 std::string, std::vector 等）是不能直接用在 UFUNCTION 参数中的，因为这些类型不在引擎的反射系统支持范围内。

打包没问题，运行报错LowLevelFatalError。。。LowLevelFatalError

已解决：打包前是livecoding编译的，需要退了用vs编译再打包

打包后连不上steam

已解决：改配置后没有重新generate

笔记：

meta = (WorldContext = "WorldContextObject")

将调用此函数的object赋值给WorldContextObject参数

PRAGMA\_DISABLE\_OPTIMIZATION

PRAGMA\_ENABLE\_OPTIMIZATION 可以关闭优化！！避免调试时变量被优化！！

编译报错无法识别的外部链接，可能是build.cs里没有添加相应的模块

非 dynamic delegate声明不需要加参数名称