接下来我将给各位评委老师讲解我们模型的代码实现。

要用代码实现我们的物流园模型，有两个问题是必须要解决的。只要解决了这两个关键问题，我们就能轻松地模拟出我们的物流园模型

首先我们来考虑第一个问题。

这个问题有两个关键词，第一个关键词是同时。同时意味着园区内的每辆货车都是并行的，这就要求我们必须使用java多线程编程的技术

第二个关键词是抢占。抢占意味着饥饿问题。什么是饥饿问题呢？

各位老师试想，当多辆货车同时对一个仓位发起抢占的请求，总是只能有一辆货车成功实现抢占，其余货车只能进入等待并在抢占结束后再次尝试发出请求，在这种情况下，难免有运气不好的货车每一次都无法抢占到该仓位，从而只能一直无限地等待下去，最终导致饥饿。

为了解决饥饿问题，我们在程序中使用了优先级调度技术，为每一辆货车赋予优先级，当一辆货车经过多次等待后，它的优先级就会越来越高，从而更容易抢占到这个仓位，这样我们就避免了饥饿问题

可以看到，我们第一个问题的解决就是为了还原真实情况下物流园中的堵塞现象

接下来我们来考虑第二个问题

经过分析，在我们的物流园模型中，最优路径是符合最优子结构的性质的，所以我们决定使用动态规划算法来得到最优路径。

所谓最优子结构，其实就是全局问题的最优解是由子问题的最优解组成的。

我们来看物流园模型中货车出库的一个例子。

现在我们有一辆货车，要从入口进入园区，经过A1、A8、B4仓位，再从出口离开园区。

这样一个全局的路径优化问题其实可以被分解为，从入口到A1仓位，从A1仓位到A8仓位，从A8仓位到B4仓位，从B4仓位到出口这样的四个路径优化的子问题。每个子问题的最优解组合在一起其实就是全局问题的最优解。

这样的性质就是最优子结构的性质。而这种先求解出许多子问题的最优解再组合在一起得到全局最优解的算法就是动态规划算法。

通过使用动态规划算法，我们就可以快速准确地得到我们想要的最优路径，从而最终实现我们的物流园模型。