2-2level（即完整售货机）的设计思路和知识点（包含部分新学）

一、售货机工作流程

1. 售货机的初始化
2. 上货：

售货机的上货

重复。。。

最后展示售货机

1. 购买：

选择商品

重复。。。

算账找零

1. 继续选择是否开始下一次购买

（其中可以回退最多三步操作）

二、进一步分析代码

1. 使用宏定义定义通道数、通道最大货物量、最大回退步骤
2. 定义通道、操作结构体及枚举操作类型
3. 各类函数

售货机初始化

显示售货机状态

回退（一个储存步骤，一个判断当前状态并进行回退）

摆放货物

选购货物（需要返回总价，所以函数用int而不用void）

主循环

注：由于题目要求一次上货后再循环购买 由一可知在int main（）中进行一次上货后重复进行购买操作，即主循环（包括选购货物函数加投币找零过程）。为实现主函数的循环，使用if条件语句在用户同意后再次应用该主循环函数直到用户选择跳出。

1. 知识点（与二对应）
2. #define 变量名 值

用于定义确定的值，如本次题目中提到的通道数、通道最大货物量、最大回退步骤

这样可以极大提高代码可读性

1. 当一个事物有多个性质时使用结构体struct将不同类型数据封装形成单一数据类型，更方便、有条理的访问这些数据，便于管理和操作以及复用，并且有效节省内存。同样使用enum枚举操作类型也有相似的好处。

其中使用typedef struct、typedef enum更加便捷，在后续定义结构体数组、变量时省去部分代码。

需要注意的是 在操作结构体中应该有一个属性是操作类型即是由enum操作类型定义type变量。

通道的结构体数组较为关键用于存储该通道货物类型、数量、价格。

1. 函数

售货机初始化：

运用for循环将channels[].属性全部赋值为0

注：对于.item字符串数组应赋为'\0'

显示售货机状态：

运用for循环与if条件语句结合对每个通道数组进行判断，当货物数>0，则输出此时通道内货物及其单价、数量等

回退：

储存步骤pushundo函数

运用栈的思想，操作结构体数组长度为3

if条件语句（并声明一个变量undoCount用于计数 存+1退-1），当该变量值小于3时直接存入，当变量值达到三时需要将其中第一步丢出（即运用for循环将数组内值前移覆盖）再将新的步骤放入。

回退步骤undo函数

当数组为空时显示无法撤回并结束函数运行。

当数组中有操作时将该操作导出给新的操作结构体变量

判断此时的操作类型并进行相应的回退。

摆放货物：

if条件语句对无效通道号、数量的判断

引用pushundo函数将操作放入栈

购买货物：

关键：指针

定义指针指向通道结构体数组，方便修改结构体内容，且能够使代码更加简介

pushundo同上

主循环：

按照前分析的流程编写即可