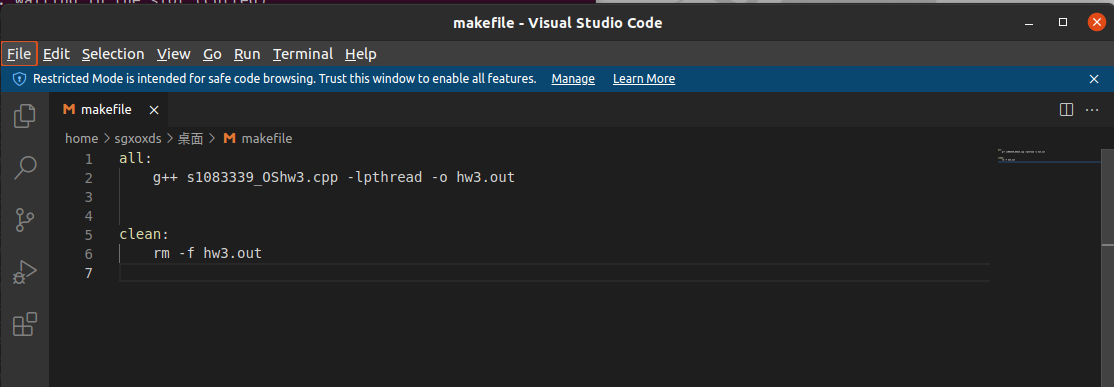
**完成的功能列表**

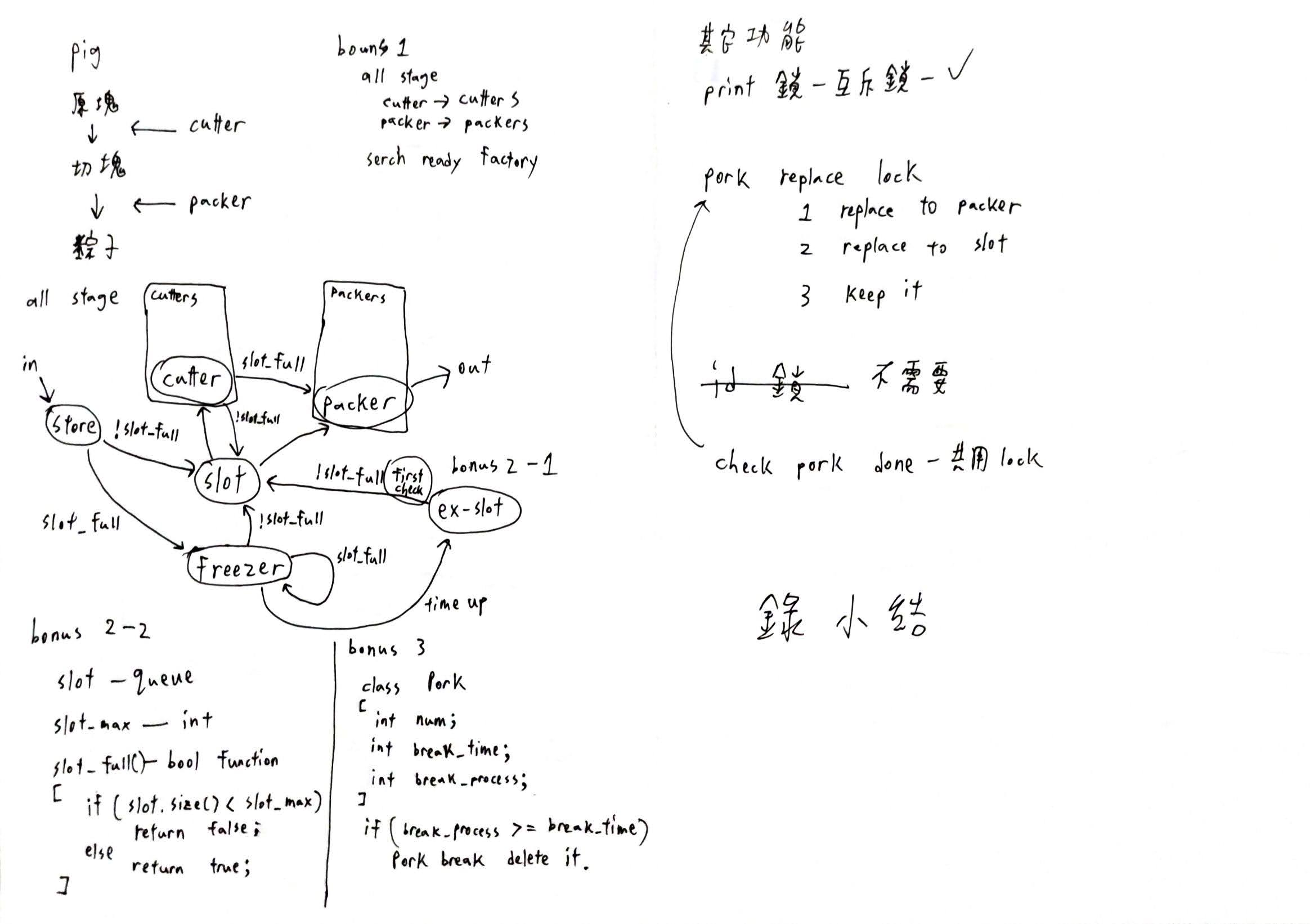
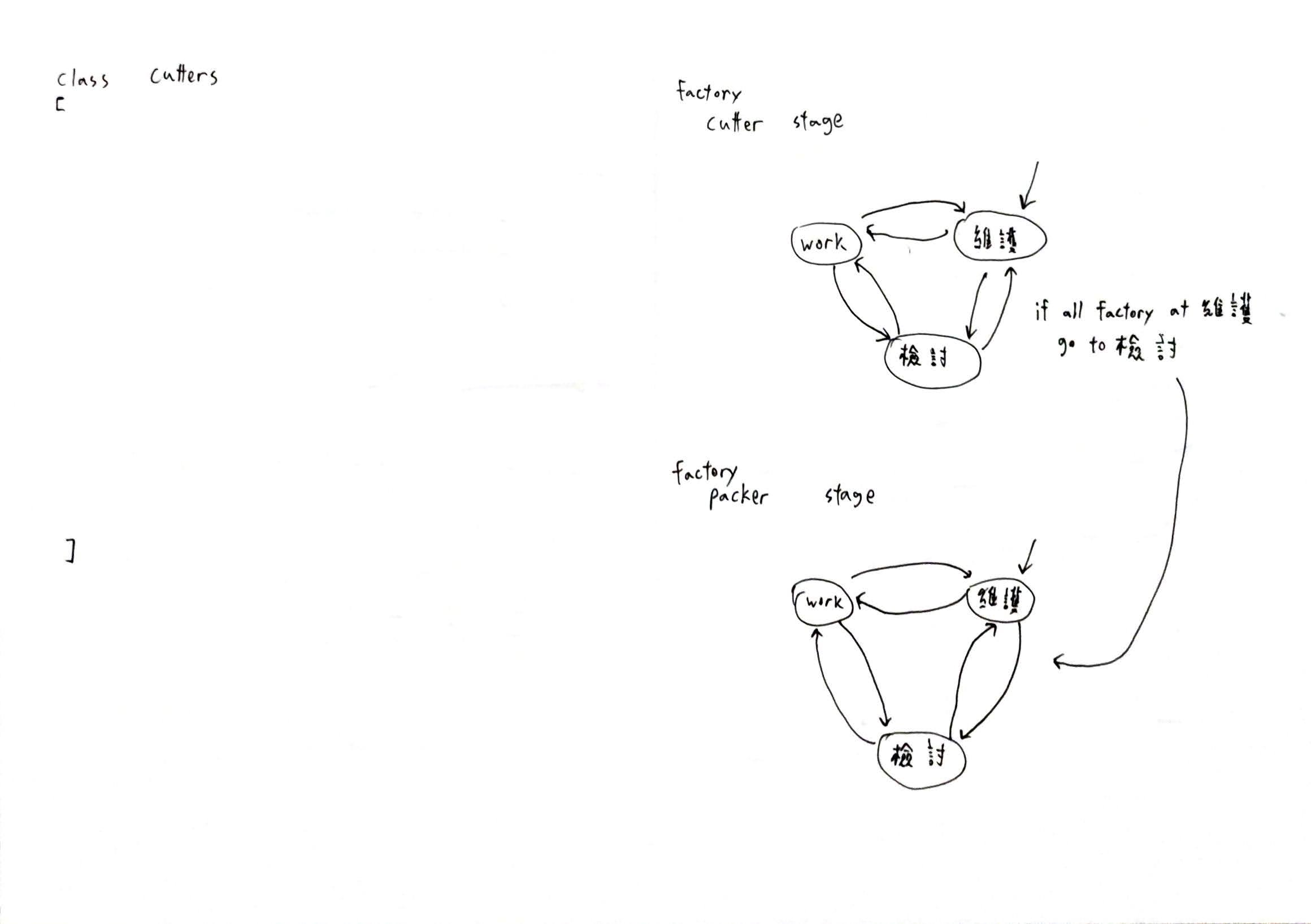
1. 使同個程式可以在不同數目的指令參數下皆可運作
2. 從store運送豬肉到slot或freezer
3. 從freezer運送豬肉到slot或回freezer冷凍
4. 檢查在slot的豬肉是否壞掉 (BONUS 功能)
5. 從slot運送豬肉到cutter或packer加工
6. 豬肉全部處理完成的檢查機制
7. 時間流逝系統
8. 檢查是否還有子程序還在執行
9. 不可搶先的輸出系統
10. 是否達到最大冷凍時間 (BONUS 功能)
11. 可以調整的工廠數目 (BONUS 功能)
12. 用thread 來模擬cutter與packer
13. 工廠(cutter與packer)是否在工作的檢查機制
14. 工廠詳細資料的struct
15. 每塊豬肉資料的struct
16. Moniter機制
17. 作業中故事敘述到的諸多功能等

**如何編譯我的程式**



g++ s1083339\_OShw3.cpp -lpthread -o hw3.out

**程式是如何運作**

設計草圖：

設計技巧：

先找出豬肉的處理流程

再去找出cutter與packer的工作流程

然後轉換流程圖為狀態機

定義好每種狀態轉換時的條件與事件 再去寫程式

如何運用 Pthread/ mutex/ semaphore 機制：

透過Moniter技巧 簡單的設計一個class去混用mutex與semaphore來達成

**BONUS 功能**

BONUS 1、2、3 我都設計出來了

BONUS 1：

就…把工廠物件化即可

透過vector去儲存工廠群的資料

再一個一個呼叫

BONUS 2：

在豬肉上新增冷凍時間的標記

在冷凍庫的豬肉的冷凍時間會不斷流逝

達到最大冷凍時間時 強制放到slot

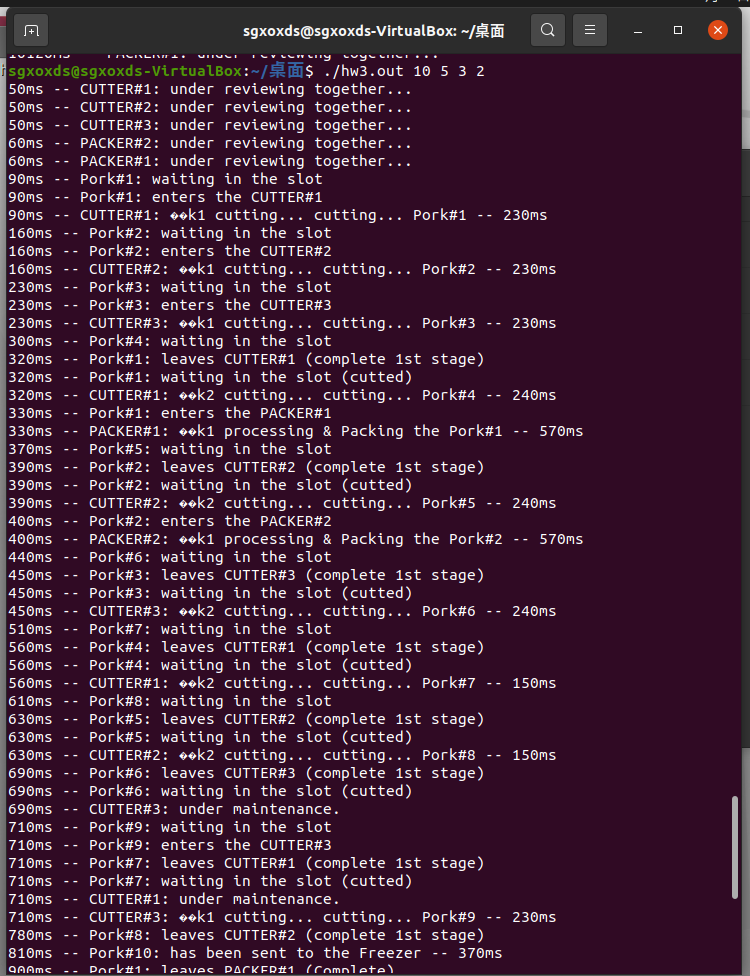
BONUS 3：

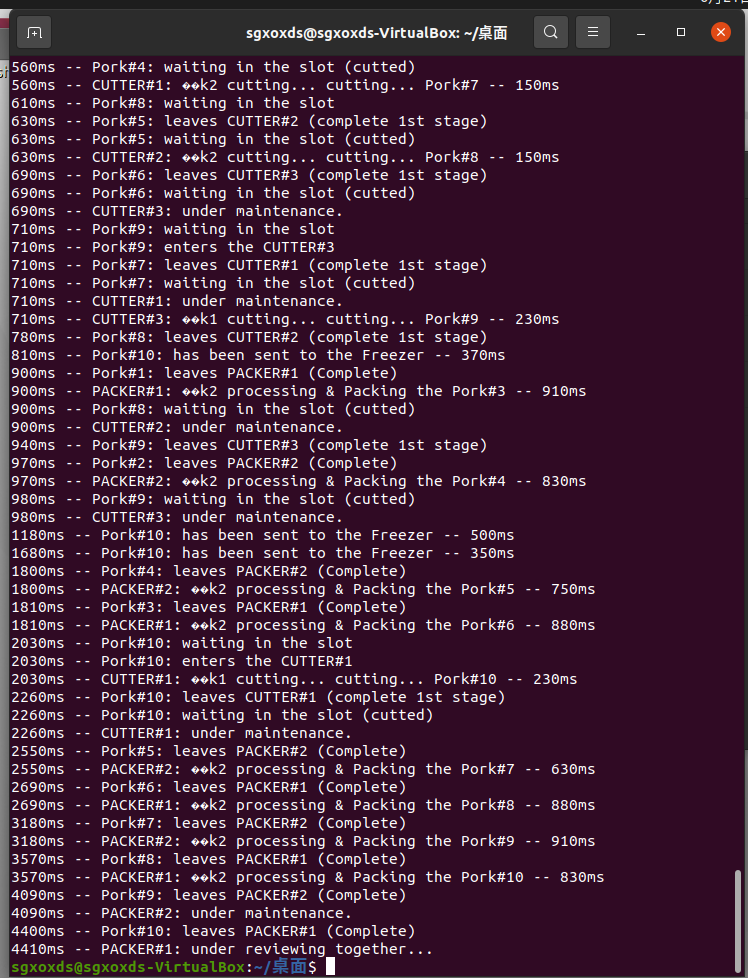
在豬肉上新增壞掉時間的標記

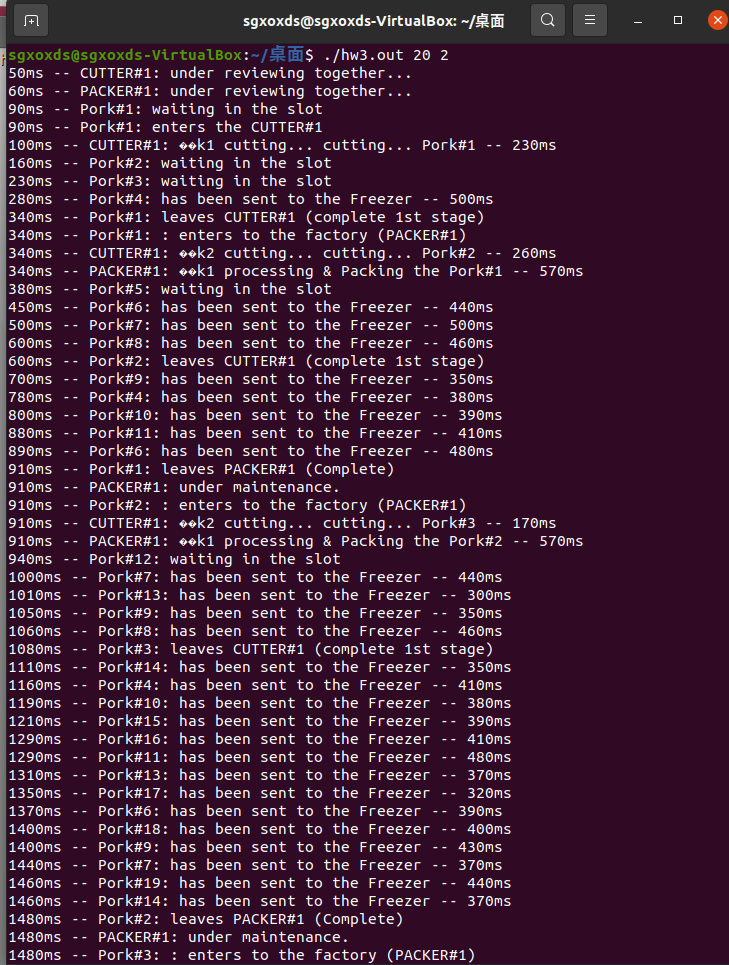
在slot的豬肉的壞掉時間會不斷流逝

達到最大壞掉時間時 移除該豬肉

測試範例 （過程與截圖）：

BONUS 1：



BONUS 2、3：

