

作業系統_第八章_Deadlock

組員: 112213063 張廷伃、112213065 張詠筑

影片網址: <https://youtu.be/sUQVWjulOpw?si=FWq6mpd3GP1xql9I>

(1) Mutual Exclusion

在影片中，電梯按鈕與電梯入口屬於不可共享的資源，一次只能由一位使用者操作或占用。因此，當其中一人正在占用電梯入口或按鈕時，另一人無法同時使用，符合「互斥」條件。

在作業系統中，互斥代表某些資源在同一時間內只能被一個行程(process)或執行緒(thread)使用。

影片中的電梯按鈕與入口即扮演臨界資源(critical resource)的角色，無法被多個使用者同時存取。

(2) Hold and Wait

角色 A 已經「持有」行李(無法放下)，但仍需要電梯按鈕來完成搭乘；

角色 B 已經「持有」電梯入口位置，卻仍在等待角色 A 先進入電梯才能按方向鍵。

兩位角色皆在持有一項資源的同時，等待另一項資源，符合「持有並等待」條件。

持有並等待表示行程在保有目前已取得的資源時，仍持續請求其他尚未取得的資源。

在影片中，A 與 B 皆未釋放手中資源，卻持續等待對方釋放所需資源，進而增加系統進入死結的風險。

(3) No Preemption

在影片中，角色 A 與角色 B 皆無法強制奪取對方所持有的資源，

例如無法強行將對方推開、奪走電梯入口或按鈕。

資源只能由持有者自行釋放，符合「不可搶奪」條件。

不可搶奪代表系統無法強制回收已分配給行程的資源，只能等待行程自行釋放。

此特性使得當行程發生相互等待時，系統無法主動介入，導致死結狀態持續存在。

(4) Circular Wait

角色 A 需要角色 B 讓出電梯入口, 才能按下「上」鍵;
角色 B 則需要角色 A 先進入電梯, 才能按下「下」鍵。
因此形成「A 等 B、B 等 A」的封閉循環等待關係, 造成系統無法繼續運作, 正式形成 Deadlock。

循環等待是 Deadlock 發生的最後一個必要條件, 表示資源請求形成封閉的等待環。
只要此循環存在, 且其他三個條件同時成立, 系統便會陷入無法前進的死結狀態。