**OShw2報告**

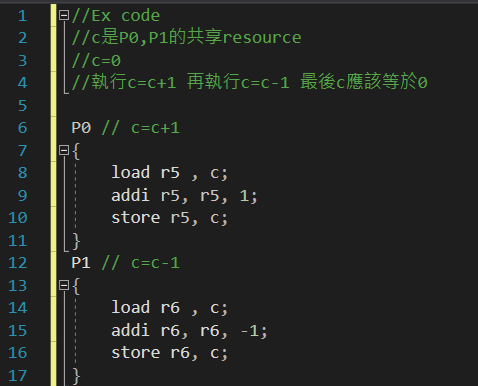
學號：1083339 姓名：楊侑承

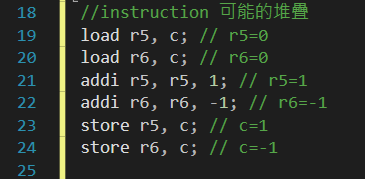
**什麼是CS？**

Critical section，指的是process中存取與其他process共享resource的程式片段。

**什麼是CS問題？**

CS問題，是指有些共享resource有無法同時被多個process存取的特性，為了解決這些共享resource的同步問題，因此對process的CS設計一個對存取共享resource的權限。



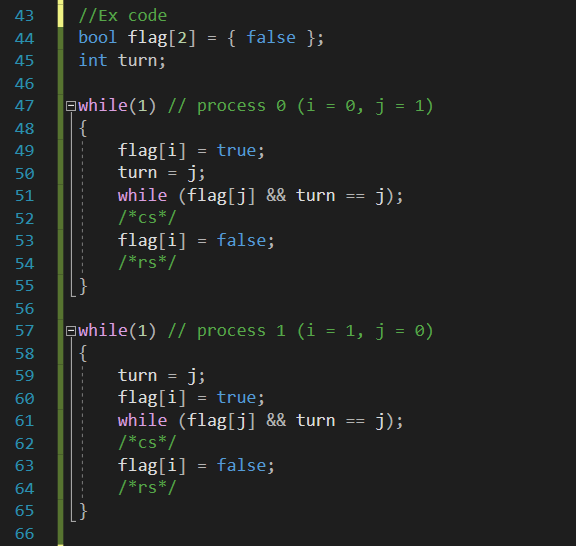


**CS問題解法的必要條件**

1. 排他性
2. 只有在CS中的process能阻止其他process進入
3. 有限的等待

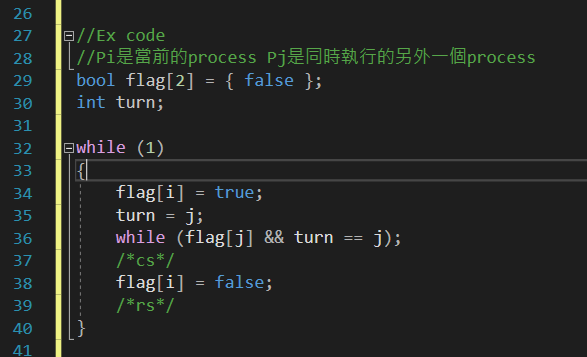
**Peterson's Solution**

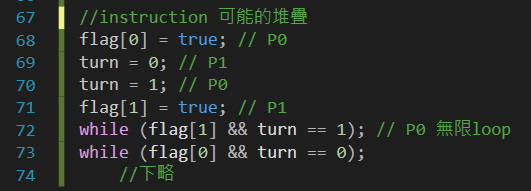
它提供了一個好的方式去描述CS問題應該如何解決的邏輯，並說明處理CS問題時的三個必要條件該如何處理。這解法僅限於2個process的CS問題處理。



**Peterson's Solution的問題**

如果兩個process的turn跟flag順序寫不一樣時 會卡死





而在現代計算機組織中，硬體設計所依據的框架是Tomasulo's algorithm。它在硬體時會優先處理已經ready的instruction，因此它的運算順序是out of order，所以會導致Peterson's Solution的執行順序並非原本在理論上的順序。

**解決方案**

軟體的方案因為硬體設計而無法被正確執行，硬體的方案則是過於複雜而且高階語言難以利用。然而OS工程師可以使用軟硬體結合來簡化硬體方案複雜的關鍵方案，例如Mutex Locks與Semaphores。但這兩套方案在不當的混合使用以處裡CS問題時，容易發生許多錯誤。因此，現行的OS偏向使用一個高階的同步結構，monitor。

Monitor由三個部分構成，共享resource宣告區、operations區、初始化區。並且在monitor中，定義了一個新的型態，通常叫condition。這個型態中定義對共享resource執行存取時一個可靠的CS問題處理方案。此外monitor還保障在monitor中的共享resource與operations的互斥性質，並且monitor在同一時間內只為一個process提供服務。

