

# 106 學年度作業系統學期群組計畫

題目：Tobacco Smokers (TS) Problem

組別：第二組

組員：賴晨和、朱家佐

GitHub 連結：

<https://github.com/NCCU-Software-Engineering/tobacco-smokers-problem>

使用語言：JAVA

題目說明：

Tobacco Smokers (TS) Problem 的問題情境為，有三種材料 Tobacco(菸草)、Cigarette papers(捲菸紙)、Matches(火柴)，一個 Agent 會固定發放其中兩種材料到 Table(桌子)上，三個 Smokers 各持有一種無限的材料，之後會到 Table(桌子)上搶奪資源，其中一個 Smoker 順利搶奪自己所沒有的其他兩種材料之後，會製作 Cigarette(香菸)，一個 Smoker 製菸完成會馬上抽菸，之後叫醒 Agent 繼續發放三種材料中的其中兩種材料於桌上，如此循環。

Deadlock 時機：

如果兩個 Smokers 擅自各搶奪桌上一種材料，如此一來，沒有 Smoker 可以完成製菸，也沒有 Smoker 可以抽菸後叫醒 Agent 繼續發放材料，至此階段發生 Deadlock，沒有後續，動作完全停止。

Deadlock 解法：

首先按照題目要求使用六個 Threads(三個 Agent、三個 Smoker)

```
public static void main(String[] args) {  
    // smokerSemaphore set 0 to prevent the Deadlock.  
    for (int i = 0; i < smokerSemaphore.length; i++) {  
        smokerSemaphore[i] = new Semaphore(0);  
    }  
  
    Agent paper_matches_agent = new Agent(PAPER, MATCHES, 0);  
    Agent matches_tobacco_agent = new Agent(MATCHES, TOBACCO, 1);  
    Agent tobacco_paper_agent = new Agent(TOBACCO, PAPER, 2);  
  
    Smoker tobacco_smoker = new Smoker(PAPER, MATCHES, 0);  
    Smoker paper_smoker = new Smoker(MATCHES, TOBACCO, 1);  
    Smoker matches_smoker = new Smoker(TOBACCO, PAPER, 2);  
}
```

圖一：六個 Threads 的宣告

之後使用三個 Semaphore 控管 Smokers，Agent 確定發放的兩種材料後叫醒 Smoker 於 Table 取材料。再一個 Semaphore 讓 Smoker 叫醒 Agent 繼續發放材料。

```
public class TobaccoSmokersProblem {  
    static int count;  
  
    static String[] resources_name = new String[] { ("tobacco"), ("paper"), ("matches") };  
    static final int TOBACCO = 0;  
    static final int PAPER = 1;  
    static final int MATCHES = 2;  
  
    static Semaphore agentSemaphore = new Semaphore(1);  
    static Semaphore[] smokerSemaphore = new Semaphore[3];  
  
    public static MyGUI mygui = new MyGUI("Tobacco Smokers Problem");  
  
    public static void main(String[] args) {  
        // smokerSemaphore set 0 to prevent the Deadlock
```

圖二：兩種 Semaphore 的宣告

亂數產生：

使用 Exponential Distribution 的 CDF 公式求出隨機項數  $x$ ， $x$  作為亂數使用於程式中。

```
private static double myRandom() {  
    double u = random.nextDouble();  
  
    int x = 0;  
    double cdf = 0;  
    while (u >= cdf) {  
        x++;  
        cdf = 1 - Math.exp(-1.0 * 0.1 * x);  
    }  
    return (double) (x) / 100;  
}
```

圖三：Exponential Distribution CDF 產生隨機亂數  $x$  ( $\lambda=0.1$ )

GUI：

使用 java swing 製作，提供 5 個 api 供主程式使用。

1. public void put(String item1, String item2)

將兩種材料放入桌上

Parameters:

item1, item2 - "tobacco", "matches" or "paper"

2. public void get(int ID, String item)

Smokers 從桌上取的材料

Parameters:

ID - Smokers ID

Item - "tobacco", "matches" or "paper"

3. public void smoke(int ID)

Smokers 顯示抽菸圖示

Parameters:

ID - Smokers ID

4. public void anger(int ID)

Deadlock 發生，Smokers 顯示生氣圖示。

Parameters:

ID - Smokers ID

5. public void restart()

抽完菸後呼叫，重置 GUI

圖示：



Agent



Smokers



Deadlock



抽菸



Matches



Tobacco



Cigarette papers