

# 2018 高等程式語言期末上機模擬考

6/7/2018

Note: 嚴禁作弊，作弊除考試 0 分外，另扣總分 15 分，嚴重者送校處理。

1. (20%) 我們要記錄處理有關教授(Professor)，研究助理(RA)，教學助理(TA)等相關資料。規格說明如下：

(a) Professor 之屬性資料如下：

name(姓名)，phone(電話)，salary(薪水)，  
deptName(所屬系所)，course(教授課程)，  
rank(等級) -- Full(正教授)，Associate(副教授)，Assistant(助理教授)

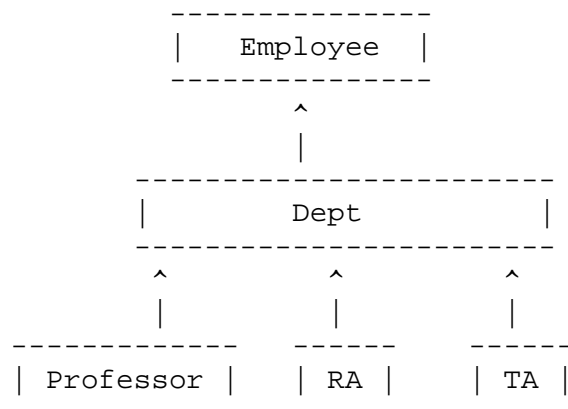
(b) RA(研究助理)之屬性資料如下：

name(姓名)，phone(電話)，salary(薪水)，  
deptName(所屬系所)，researchID(研究計畫之 ID)，  
grad(研究生) -- Master(碩士班)，Doctoral(博士班)

(c) TA(教學助理)之屬性資料如下：

name(姓名)，phone(電話)，salary(薪水)，  
deptName(所屬系所)，course(教授課程)，  
grad(研究生) -- Master(碩士班)，Doctoral(博士班)

(d) 請將上述屬性拆解成如下之繼承關係，請注意屬性不要重複出現。  
透過如下的繼承關係，class Professor 會含有上述所有 professor  
該有的資料，class RA, TA 亦同。



(e) 請根據所給的測試程式，實作所有的 class，並能產生所要的輸出。  
測試程式中有的建構子都要實作。所有的 set, get methods 必須配對提供。亦即有 set method，就有 get method，反之亦同。

檔名：包含 T1\_ID.java, 其他 java 檔, 置於 t1 package 內

// 測試程式 T5\_ID.java (除非必要，請不要修改測試程式)

```

public class T1_ID {
    public static void main(String[] args) {
        Professor prof1 = new Professor("Jack Smith","0968123456",90000);
        prof1.setCourse("Data Structure"); // 教授「資料結構」課程
        prof1.setRank("Full");             // 正教授
        prof1.setDeptName("Computer Science"); // 資訊科學系
        Professor prof2 = new Professor("Steve Bush","0921345678",85000);
        prof2.setCourse("Operating Systems"); // 教授「作業系統」課程
        prof2.setRank("Associate");           // 副教授
        prof2.setDeptName("Electrical Engineering");
        TA ta1 = new TA("John McDonold","0988987654",21000); // TA:教學助理
        ta1.setGrad("Master");                // 碩士班
        ta1.setCourse("Basic Computer Concepts"); // 「資訊概論」助教
        ta1.setDeptName("Computer Science");
        TA ta2 = new TA("Amy Miller","0938246801",25000,"Doctoral"); // 博士班
        ta2.setCourse("C Programming");        // 「C 程式語言」助教
        ta2.setDeptName("Civil Engineering");
        RA ra1 = new RA("Alex Jones","0939876543",20000); // RA:研究助理
        ra1.setGrad("Doctoral");
        ra1.setResearchID("NSC-2008-10");      //研究計畫編號
        ra1.setDeptName("Computer Science");
        RA ra2 = new RA("Mary Adams","0941678954",23000,"Master");
        ra2.setResearchID("NSC-2008-20");
        ra2.setDeptName("Mathematics");
        System.out.print(prof1+"\n\n");
        System.out.print(prof2+"\n\n");
        System.out.print(ta1+"\n\n");
        System.out.print(ta2+"\n\n");
        System.out.print(ra1+"\n\n");
        System.out.print(ra2+"\n\n");
    }
}

```

// 螢幕輸出如下：

```

Jack Smith: Full Professor of Computer Science.
==> Course: Data Structure
==> Salary: 90000, Phone#: 0968123456

```

```

Steve Bush: Associate Professor of Electrical Engineering.
==> Course: Operating Systems
==> Salary: 85000, Phone#: 0921345678

```

```

John McDonold: Teaching Assistant, Master program of Computer Science.
==> Course: Basic Computer Concepts
==> Salary: 21000, Phone#: 0988987654

```

```

Amy Miller: Teaching Assistant, Doctoral program of Civil Engineering.
==> Course: C Programming
==> Salary: 25000, Phone#: 0938246801

```

```

Alex Jones: Research Assistant of Computer Science.
==> Research ID: NSC-2008-10
==> Salary: 20000, Phone#: 0939876543

```

```

Mary Adams: Research Assistant of Mathematics.
==> Research ID: NSC-2008-20
==> Salary: 23000, Phone#: 0941678954

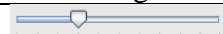
```

Coding by 1B-ID

2. (10%) 請改寫第 1 題，設定一陣列 `Employee emp[] = new Employee[MAX];` 然後將所有員工資料存在 `emp` 陣列中，利用下列多型概念產生同樣的輸出。請將原始程式碼放在 package `p2` 中。

```
for(int i=0;i<numEmployees;i++){  
    System.out.println(emp[i]);  
}
```

3. (50%) 請參考老師上課範例，將原始程式碼放在 package `finalexam` 中，整個程式放在 `JTabbedPane` 之 `tab1` 中，名稱為 `Final Exam`。請實作如下功能：

Color Reset	C1	C2	C3	<input type="checkbox"/> filled	Rect	Oval	Sudoku	Randgen
				<input type="checkbox"/> gradient	Square3	Circle3	Flag3	

- (10%) 上圖所有按鈕請放在一個 `JPanel` 中，配置使用 `GridBagLayout`，放在北方。
- (10%) 完成三個顏色按鈕(`C1`, `C2`, `C3`)及 `Color Reset` 按鈕，預設顏色 `C1` 為藍色，`C2` 為紫色，`C3` 為綠色，`C1`, `C2`, `C3` 一開始背景顏色就必須用預設顏色顯示，當設定不同顏色後可以改變按鈕之背景顏色。`Color Reset` 按鈕會將 `C1`, `C2`, `C3` 之背景設為預設顏色。
- (10%) `Flag3` 繼承 `Rect`，可以用滑鼠畫三色國旗，垂直等分，顏色由左到右由 `C1`, `C2`, `C3` 決定。
- (10%) `Randgen` - 可以亂數畫出 `n` 個圖形 (包含 `Rect`, `Oval`, `Square3`, `Circle3`, `Flag3`)，`n` 值由 slider 決定，`n` 值範圍落在 1~30。
- (10%) `Sudoku` -- 可用滑鼠產生一組 9x9 數獨 (正方形)，如下圖，請填入一組成功數字 (如圖所示即可，不必真的下)。數獨 9 個大框框，線條比較粗(5px)，每個框框內還有 9 格，線條比較細(2px)。

8	3	5	4	1	6	9	2	7
2	9	6	8	5	7	4	3	1
4	1	7	2	9	3	6	5	8
5	6	9	1	3	4	7	8	2
1	2	3	6	7	8	5	4	9
7	4	8	5	2	9	1	6	3
6	5	2	7	8	1	3	9	4
9	8	1	3	4	5	2	7	6
3	7	4	9	6	2	8	1	5

filename: `Final-ID.zip` //請將所有檔案壓縮成 `zip` 檔

- (10%) 請將本次考試程式轉成可執行的 `Final_ID.jar` 檔，並放入 `src` 中，點選 `jar` 檔可以直接執行。如果考試現場有產生 `jar`，但無法執行，請告知助教。
- (10%) 請將 `Ball Bouncing` 程式放入 `JTabbedPane` `tab 2` 中，名稱為 `Ball Bouncing`，要可以正常執行。將原始程式碼放在 package `ballbouncing` 中。