

# 中央大學 112 學年度計算機概論 II 作業 05

## Problem 05:組隊拳擊機排行

### 問題描述

遊戲《楓葉物語》是一款 MMORPG(大型多人線上角色扮演遊戲)，在《楓葉物語》的世界中總共有五種職業，分別為劍士(Warrior)、弓箭手(Bowmen)、法師(Magician)、盜賊(Thief)、海盜(Pirate)，每個職業除了在操作方式不同之外，最主要的差別為基礎攻擊力不同，但同時成長率也不同，在等級趨近於滿等的狀態下，每個職業的攻擊力也會趨於相同。

每個職業的基礎攻擊力如下所示：

- 劍士(Warrior) : 40
- 弓箭手(Bowmen) : 25
- 法師(Magician) : 30
- 盜賊(Thief) : 50
- 海盜(Pirate) : 34

遊戲的官方這次舉辦了一項限時活動，需要透過組隊的方式來挑戰活動的 BOSS，除了透過組隊成員當下的攻擊力來對 BOSS 造成傷害之外，這次活動添加了一個 BUFF(增益效果)是基於隊伍中每個成員的職業基礎攻擊力的總和，來對 BOSS 造成的額外傷害，且該 BUFF 還會根據隊伍中的職業組成來進一步獲得提升，若隊伍中的某個職業有 2 名(含)以上的成員，每多一名則額外傷害會再增加該職業的基礎攻擊力，例如：隊伍中有 3 名劍士的成員，則額外傷害會增加劍士職業的基礎攻擊力的 2 倍，以此類推。

在活動結束後，官方會根據報名參加隊伍的 BUFF 的高低做出一個排行榜，請你寫出一支程式來計算每個隊伍的 BUFF，並輸出最終的排行榜。

## 實作限制

本次作業需要實作 Inheritance(繼承)以及 Polymorphism(多型)，請在程式碼中定義以下的 class：

```
public static abstract class Character {  
  
    // return player's or team's attack  
    public abstract int attack();  
  
    // return all teammate in String type  
    public abstract String teammate();  
}
```

此 class 為之後要定義的 class 的父類別，class 內的東西不可做刪減，但允許做增加，其中的兩個函式 attack() 會回傳該職業的基礎攻擊力或是隊伍內的攻擊力總和，teammate() 會以字串形式回傳該職業成員的名稱或是隊伍的所有成員，兩個 function 都是 abstract 的，因此繼承了此類別的 class 都需要去實作 attack() 以及 teammate() 的內容。

接著定義以下的 6 個 class：

```
public static class Warrior extends Character {  
    // Implement function attack() and teammate()  
}  
  
public static class Bowmen extends Character {  
    // Implement function attack() and teammate()  
}  
  
public static class Magician extends Character {  
    // Implement function attack() and teammate()  
}  
  
public static class Thief extends Character {  
    // Implement function attack() and teammate()  
}  
  
public static class Pirate extends Character {  
    // Implement function attack() and teammate()  
}  
  
public static class Team extends Character {  
    // Implement aggregation to Character class  
    // Implement function attack() and teammate()  
}
```

6 個 class 都繼承了父類別 Character，除了必須實作的 attack()以及 teammate()之外，皆可額外增加需要的內容，前 5 個 class 為五個職業的類別，而第 6 個 class Team 是隊伍的類別，與前 5 個不同的是除了實作 attack()和 teammate()外，還需要建立對 Character class 的 Aggregation 關係。

請實作以下的 function，並在 main 中呼叫：

```
// print the ranking list
// the incoming parameter must have ArrayList or array of Character type
public static void PrintRank(ArrayList<Character> arr)
{
    /*
     * You can add other code here
     */

    // use for loop or foreach to access the element in parameter
    // Do not remove the following two instructions in for loop or foreach
    // But you can add other code in it
    for(int i = 0; i < arr.size(); i++){
        // If the parameter is array, you can use arr.length and arr[i]
        System.out.println(arr.get(i).teammate());

        System.out.println(arr.get(i).attack());
    }
}
```

PrintRank 會將最後的排行榜輸出出來，傳入的參數必須要有 Character 類別的 ArrayList 或 array，可依照自己的需求修改參數名，亦可新增計算所需的傳入參數。PrintRank 的內容可自行定義，但必須包含圖片中下方的內容：使用迴圈或 foreach 的方式訪問傳入的 ArrayList 或 array 的元素，將每個元素的 teammate()以及 attack()輸出出來，為了實作繼承及多型，因此不可移除這兩個指令，但可在迴圈或 foreach 中新增內容來符合輸出格式。

以上的內容請寫在同一個 java 檔中，上述的 class 和 function 可直接放在 class A05\_XXXXXXXXX 內，上述 class 中的變數請使用 private。

未按規定者及未完成實作內容者將斟酌扣分。

## 輸入說明

- 第一行輸入一個正整數  $n$ ，代表隊伍的數量，接著會以報名順序輸入  $n$  組隊伍的資訊
  - 輸入一個正整數  $m$ ，代表隊伍成員的人數，接著輸入  $m$  位成員的資訊
  - 每位成員會先輸入職業類別，接著再輸入成員的暱稱

其中：

- $1 \leq n \leq 100$
- $1 \leq m \leq 48$
- 同個隊伍中，同個職業不會出現相同的暱稱

## 輸出說明

依據每個隊伍的 **BUFF** 由高到低做排行榜，接著將每個隊伍的組隊成員及 **BUFF** 輸出，組隊成員需做排序，以職業做遞增排序優先，接著對暱稱做遞增排序後才輸出，每個隊伍之間需用換行隔開。

若出現 **BUFF** 相同的隊伍，則先報名的隊伍排名較高。若出現編組完全相同(即成員一致)的隊伍重複報名，仍要將相同的隊伍輸出，排名也會根據報名順序而定。

詳細的輸出格式可參考範例。

## 範例

Sample Input 1	Sample Output 1
1 5 Warrior Mikhail Bowmen	Tier 1 team:  Bowmen : Irena  Magician : Oz  Pirate : Hawkeye  Thief : Eckhart

Irena Magician Oz Thief Eckhart Pirate Hawkeye	Warrior : Mikhail Team BUFF : 179
Sample Input 2	Sample Output 2
2 5 Warrior Blaster Bowmen WildHunter Magician BattleMage Thief Xenon Pirate Mechanic 6 Warrior Aran Magician Freed Bowmen Mercedes Thief Phantom Magician Luminous Pirate Eunwol	Tier 1 team: Bowmen : Mercedes Magician : Freed Magician : Luminous Pirate : Eunwol Thief : Phantom Warrior : Aran Team BUFF : 239  Tier 2 team: Bowmen : WildHunter Magician : BattleMage Pirate : Mechanic Thief : Xenon Warrior : Blaster Team BUFF : 179

## 繳交格式

### 1. 上傳內容須為 java 檔

- 程式碼(請繳交.java 檔)，不需要壓縮。
- 檔名為 AX\_學號(EX : A05\_112000000)

### 2. 程式碼開頭請標示

```
/*  
Assignment 05  
Name: 你的姓名  
Student Number: 你的學號  
Course 2024-CE1002-你的班級  
*/
```

### 3. 檔案編碼請統一使用 : utf-8

### 4. 請針對程式內的重要功能進行「適當註解」，簡單說明該段程式碼的作用，註解的部分會納入評分考量，請勿隨意註解

### 5. 統一縮排格式，鋸齒縮排列入扣分項目。

以上繳交方式、內容錯誤的斟酌扣分、完整作業格式的解說請見新 ee-class 公告中的附檔「作業格式及繳交說明.pdf」。若對作業還有問題，請依照格式寄信到助教信箱詢問：[ncuics112@gmail.com](mailto:ncuics112@gmail.com)。

## 解題概念

Class、Aggregation、Inheritance、Polymorphism。