0819823陳子祈 演算法導論 HW5

1. 試驗成果

今天有一根87及一根14單位長的rod，請大家利用老師上課所講的**top-down和bottom-up**的方法找出要切割成幾根rod才能賣出多少最高的價格，也請在報告中告訴我們最好的切法。

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

二、 討論

如果今天不是要找最高的價格的組合，而是最低的價格組合，你覺

得演算法應該做怎樣的修正呢?(在結報中描述你的想法就可以了)

Ans: 很簡單，只要調整一些初始設定及後續的假設條件即可:

1. 初始設定

Memoized\_Cut\_Rod函式裡面原本r設為除了第0項為0，其餘1~n都是負無限大(INT\_MIN)。改成除了第0項為0，其餘1~n都是正無限大(INT\_MAX)。

Memoized\_Cut\_Rod\_Aux 函式裡面max變數名稱改成叫做min，初始值改成正無限大(INT\_MAX)。Extended\_Bottom\_up\_Cut\_Rod 函式裡面max變數名稱也改成叫做min。

1. 假設條件

Memoized\_Cut\_Rod\_Aux 函式裡面if(max < d) 改成if(min>d)、Extended\_Bottom\_up\_Cut\_Rod 函式裡面if(max < p[i] + r[j-i]) 及else if(max < r[i] + r[j-i]) 分別改成if(min > p[i] + r[j-i]) 及else if(min > r[i] + r[j-i])

三、 心得

這次作業實作動態規劃的經典Cut rod問題，我覺得很像玩益智遊戲，很需要動腦思考整個解法的設計。幸好講義上與網路上都有很多資源可以參考，若要從頭想演算法可能要花很多時間。不過這份作業出了一個講義沒有寫的問題，就是使用top down要怎麼記住每一刀切在哪裡。我參考了bottom up記住每一刀切在哪裡的方法，驚覺只要在max要被更新的時候記住當下for loop執行到第幾個iteration就好了，因此我成功寫出來top down的部分。另外，我使用top down執行長度為87的rod會因為recursive function執行時間很長而跑不出來，我後來想到是因為r在top down時不能用pass by value，會因為function call而刷新r，所以我就在class裡面新增一個r，這樣r就不會被洗掉。