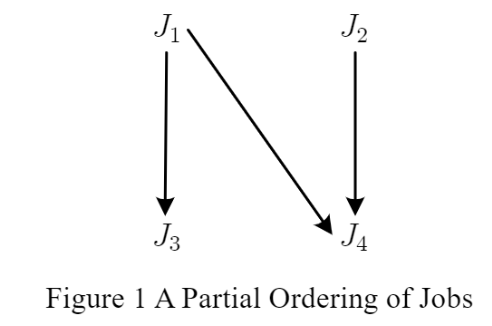
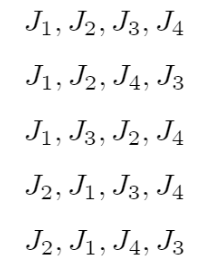
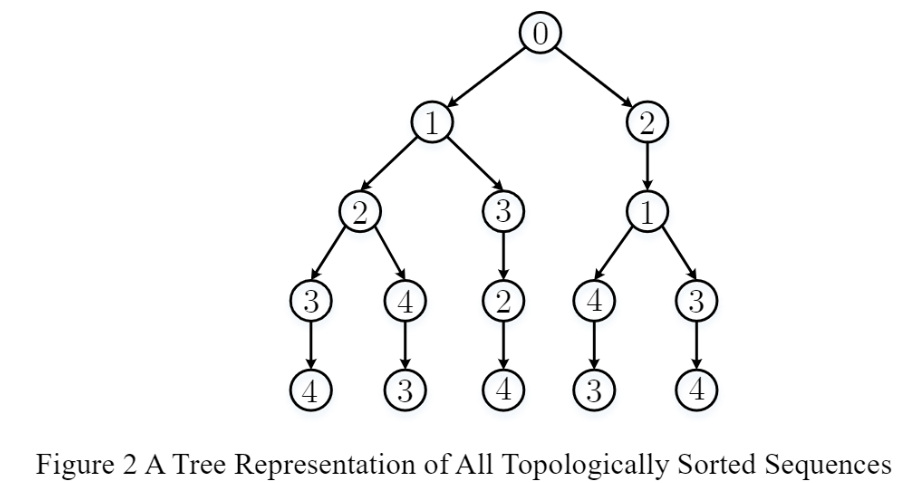
Reading Report:

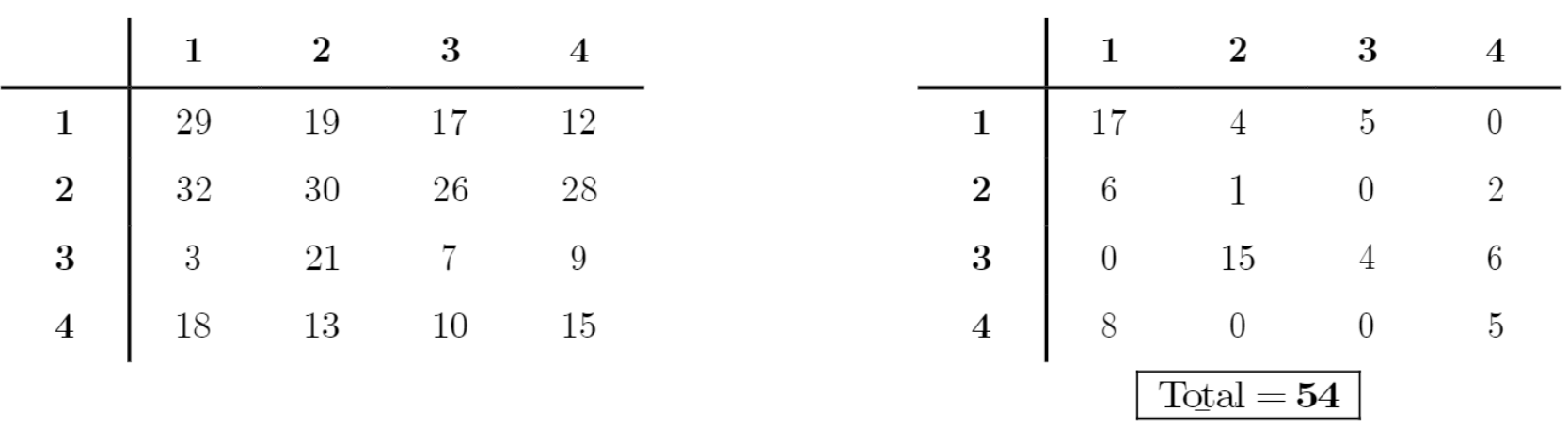
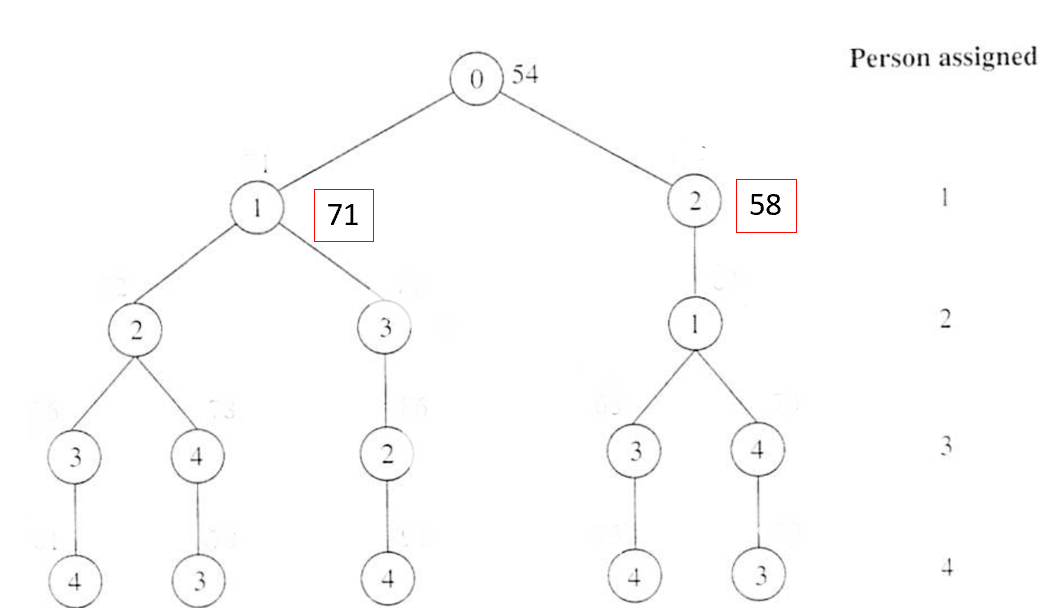
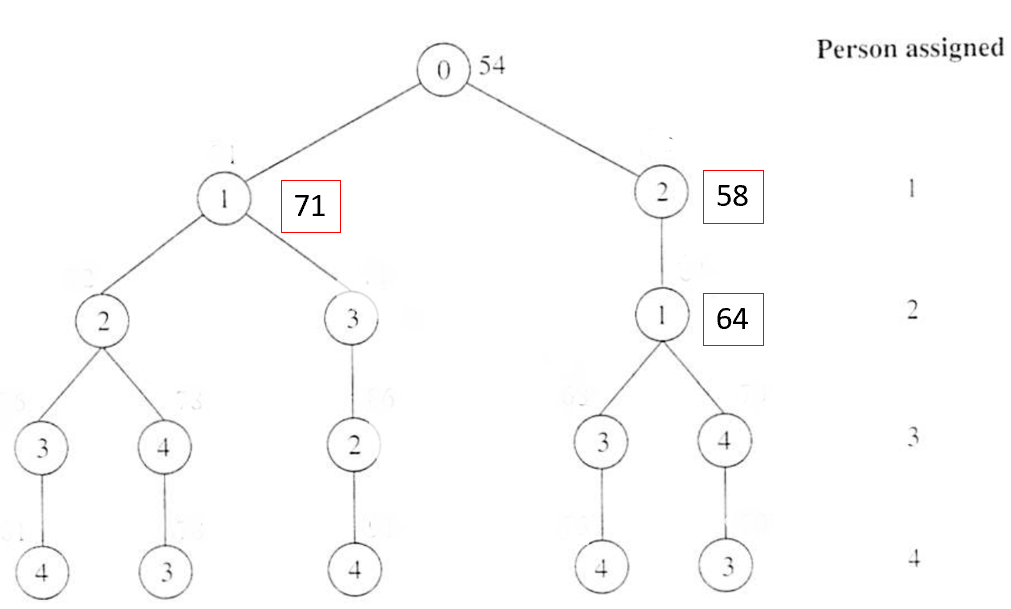
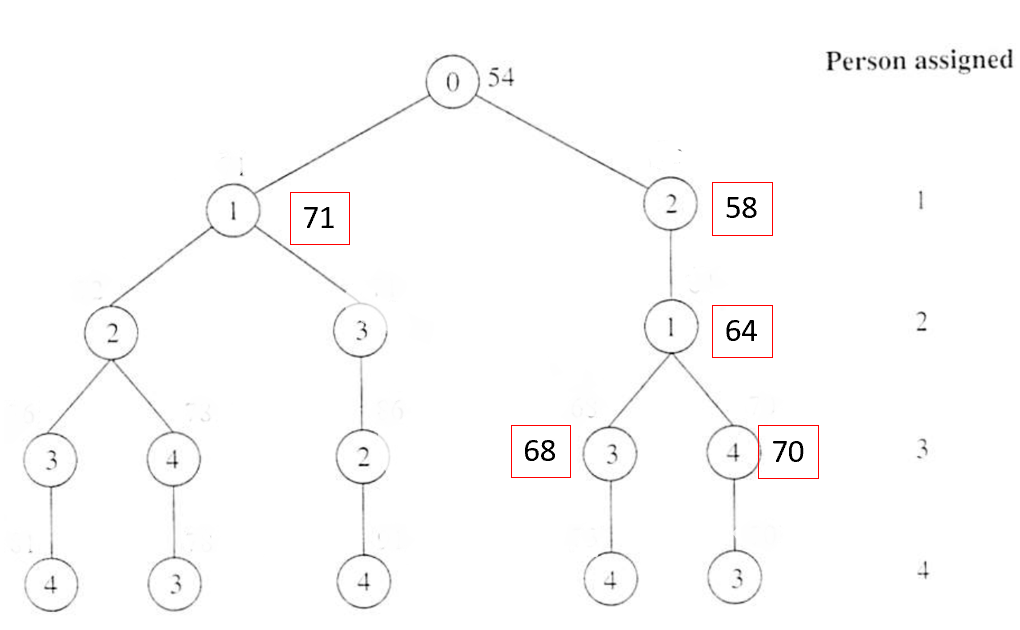
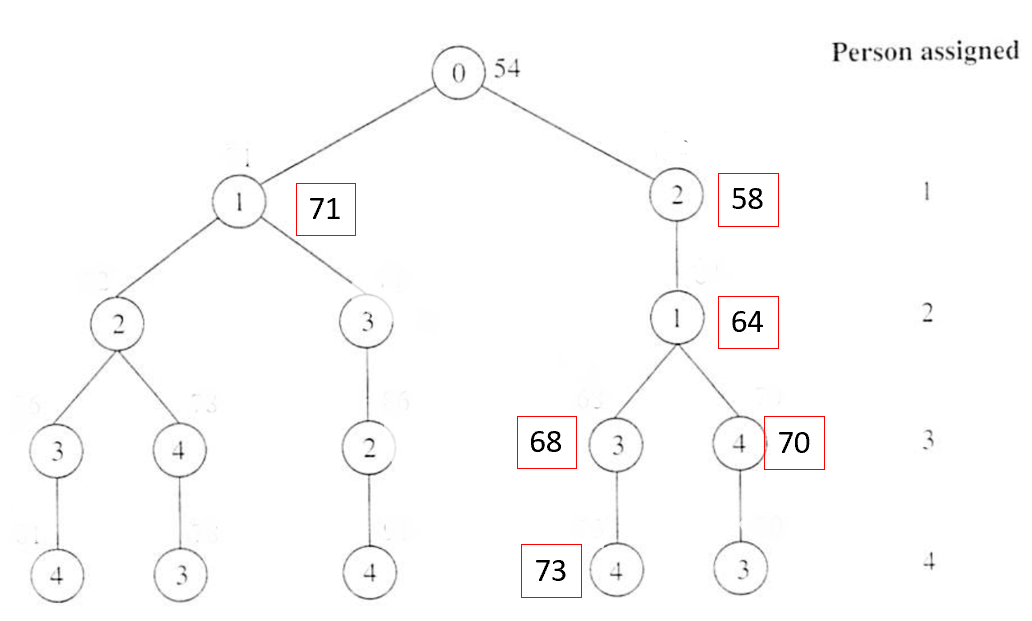
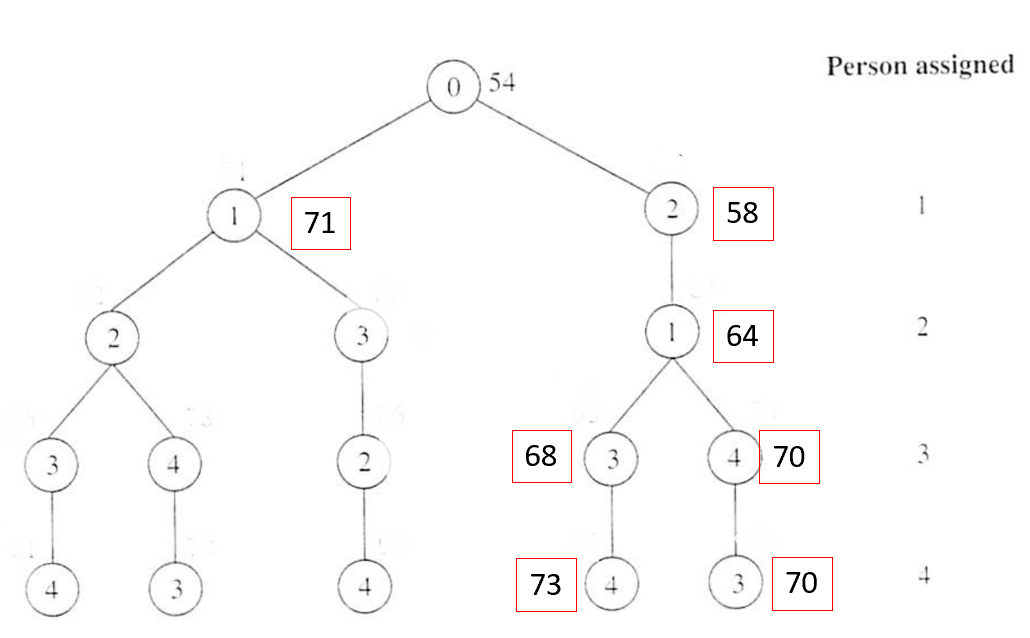
A Personnel Assignment Problem Solved By The Branch-And-Bound Strategy

學號: r08922136

姓名: 黃新予

1. 問題定義  
   此篇論文要探討的問題是personnel Assignment Problem (PAP, 指派工作問題)，此應用情境常見於公司指派任務或分組分工，一群人要分配工作，有些工作有先後順序，有些工作則可以平行做，而每個人根據不同能力分配不同工作讓整體最好等等。  
   以數學方法描述此問題則
   1. 給定一個線性有序集合代表人員(Person): P = {P1, P2, P3, P4, …, Pn}, 其中P1 < P2 < P3 < P4…< Pn，此順序可以依照能力、身高、年齡等指標皆可。
   2. 給定一個偏序集合代表工作(Job): J = {J1, J2, J3, J4,…, Jn}，其中偏序代表任務的性質，例如有些任務具備先後順序、有些任務則沒有先後順序。有先後順序的工作在J集合中一定呈現 Ji < Jj < Jk, i < j < k,這樣的關係。
   3. 每人分配工作之函數f(Pi)必須為一對一映射且符合其能力，即
      1. 若i ǂ j 時，f(Pi) ǂ f(Pj) 🡺 不同人分配不同的工作
      2. 若f(Pi) <= f(Pj), Pi < Pj 🡺 可以理解成比較能幹的人負責比較難的任務。
   4. 每個人被分配某工作都會耗費成本Cij，引入Xij = {0, 1} 代表人員i 和j 工作是否匹配，匹配則為1, 不匹配則為0。則最終分配情形產生的成本為
   5. 此問題所需要的解為最小化最終分配所需的成本，即為。
2. 解法描述  
   原本的PAP為NP-Hard問題，但利用Branch-and-Bound可以較有效率地處理。理由如下。
   1. 找出拓樸排序  
      一個偏序的任務集，皆可以化成一個有向無循環圖(DAG, Directed Acyclic Graph)，因此我們可以利用拓樸排序找出每個任務的優先次序。  
      例如一個數量為4的任務集，J = {J1, J2, J3, J4}，且其中任務的關係為  
      可以理解箭頭代表任務難易程度，其中J1比J3, J4 簡單，J2比J4簡單，但{J1, J2}, {J3, J4}的難易程度沒有先後次序之分。以上圖而言所有可能的拓樸排序可列舉為下圖:   
        
      因此拓墣排序的順序就是一種可行的分配任務情形。  
      我們可以根據以下三步驟，將所有拓樸排序表示成一樹狀結構
      1. 選取一個沒有前繼工作的任務。例如J1, J2
      2. 將上述任務定為節點，並在任務集中移除
      3. 剩下任務仍為偏序狀態，重複I繼續選取下一個節點

根據上述步驟可將此任務集繪製成樹狀結構，每一條由根到葉的路徑都是可能的拓樸排序。

* 1. 接著將成本矩陣進行成本約化，也就是將每一行每一列的數值都減去該行該列的最小值，使得每一行每一列都至少有一個元素為0。並記錄此時的成本，此成本即定義為工作分配的下界(lower bound)。例如  
       
     上圖中，左側為原本的成本矩陣，右側為約化後的成本矩陣(reduced cost matrix)，54表示要成功分配任務最少要耗費54的成本。
  2. 根據拓墣排序以及成本下界54，則可利用branch and bound展開樹狀結構。論文中將所有點都展開，但其實大可不必。
     1. 首先展開J1, J2, 可得兩節點的預期最小cost為71, 58，則優先展開預期成本下界小者58  
        
     2. 可得目前預期最小成本{64, 71}, 繼續展開64節點  
        
     3. 可得目前預期最小成本{68, 70, 71}, 繼續展開68節點  
        
     4. 可得目前預期最小成本{ 70, 71}, 且73為一可行解之成本。繼續展開70節點  
        
     5. 可得目前預期最小成本為{71}，但70為一可行解之成本，且70 < 71，表示左側所有節點都不需再展開了，此問題的分配為 P1-J2, P2-J1, P3-J4, P4-J3，成本為70。  
        

1. 讀後心得  
   個人覺得branch and bound真的是一個很優雅的想法，雖然在worst case並沒有辦法減少問題的複雜度，但卻能夠在多數的情況下，將問題所需展開的步驟減少。最近因為許多課程以及個人研究project都需要多人合作，對於要如何分配合適的任務給合適的人這類的問題算是心有戚戚焉，這篇論文剛好打中我最近在思考的問題，我也在想如何將論文的內容實際應用在工作分配，總體來說我覺得這篇很平易近人，讀起來很流暢，而且用的方法步驟也是清楚易懂。