**Bellman-ford**

**程式說明:**

程式一開始先讓使用者輸入節點數和路徑數，接著再輸入路徑(ex:3 5 3 節點3到節點5 cost 3)，再來將節點1設為起點(cost 0)，節點1到其他節點的距離設為inf(999999)，接著進入迴圈後先將現在起點到各節點的cost狀態copy下來，再對每條路徑進行比較(ex:3 5 3 ,如果起點到節點3再加cost 3到5的距離小於起點到節點5的話就更新路線)，之後再檢查更新過後的路徑是否與一開始copy的狀態相同，如果一樣就跳出迴圈不一樣就繼續，最後再顯示最短路徑結果。

**執行結果與說明:**

輸出結果為設節點1為起點(cost 0)再顯示出起點到各節點的最少cost(最短路徑)。

**遭遇的困難與解決辦法:**

一開始不太清楚Bellman-ford跟Dijkstra的差別，上網查詢才明白前者是每次都從起點開始演算，後者是從起點向外擴散。