1. 可建立項目：

p-graph

all p to p length

(p to all-p’s total cost)

p’s neighbor num

all p to p BFS route

q’s all gate-ID

1. 目標
2. p-q 配對
3. 依shortest path判斷如何swap
   1. 看路徑上兩兩之間的max( gate-ID )
4. 思考
5. p-q 配對
   1. at most p-link pairs
   2. see min( neighbor’s neighbor num )
   3. loop from “max n-n-num’s p” (BFS) – V
   4. gate-ID
      1. 兩者皆第一次出現，優先最高? <- 在沒有重複p出現前
      2. 出現重複p後，中間可隔重複p(?)
6. 步驟
   1. 統計p-link pairs組的q在gate-ID裡的出現次數(ex-num)，取max放入max n-n-num’s p (?)
   2. 以一圈min(n-num中未填者, ex-num中剩餘n)次數 跑類BFS 模式
      1. N-num: (前n-num個) p’s peer排序：<n-n-num對應ex-num>
      2. ex-num: (前ex-num個) p’s n排序：<n-n-num對應ex-num>

~~\*問題：優先級最高者可能無法直接執行~~

~~> 判斷所有優先級最高者，若pair間length>1，(swap至=1 ?)~~

1. 架構
   1. Class
      1. Physical q-bit (PQB): 到各點距離+到各點路徑+鄰居數
      2. Logical q-bit (LQB): PQB位置+gateID