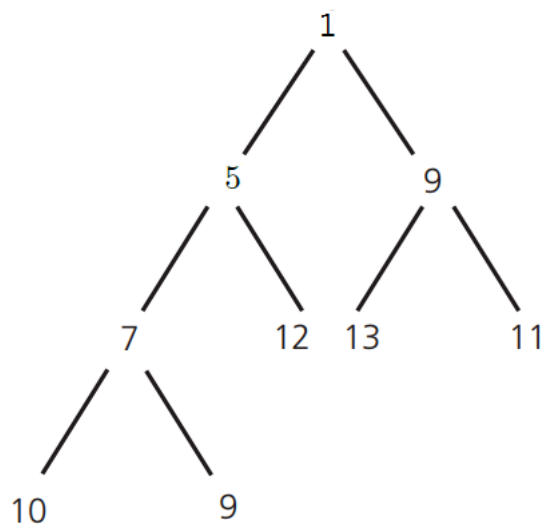


(a)



(b)

因為 heap 的插入跟刪除的時間複雜度較低。

(c)

如果 hash 時常造成 collision，會需要用 open address 或 chaining 處理，但不論用何種處理方式都會增加時間複雜度，也會造成空間的浪費。

(d)

By Prim's algorithm

$$a = 1(d) + 2(e) + 1(f) + 3(i) + 4(g) + 4(h) + 4(a) + 2(b) = 21$$

$$b = 3(e) + 1(f) + 2(c) + 1(d) + 4(g) + 4(h) + 4(a) + 2(b) = 21$$

$$a+b = 42$$

(e)

因為連接到 i 跟 h 皆只有一個路徑，但 Hamilton circuit 需要經過每一個點且起點跟終點為同一個點，若以 i 或 h 為起點將無法回來 i 或 h；若以其他點為起點則到達 i 或 h 後無法回到起點(終點)

(f)

利用一個 array 存從起點到所有未到達過的點的距離，然後再將 array 整理成 min heap 這樣就可以利用 min heap 的特性，快速找到最小值，而且我們可以使每個 operation 的時間複雜度都在  $O(\log n)$  以內。