

演算法 測驗 (2013/11/26) by Jehn-Ruey Jiang (江振瑞)

姓名: _____ 學號: _____

1. 決定以下的敘述是對還是錯，並解釋對或錯的原因:

若一個問題是 NPC，則我們已經證明沒有任何確定演算法(deterministic algorithm)可以在最差狀況下(worst case)以多項式時間複雜度解掉這個問題。

錯。目前為止，我們找不到任何 NPC 問題可以使用確定演算法(deterministic algorithm)在最差狀況下(worst case)以多項式時間複雜度解掉。但是到目前為止，也沒有人可以證明這樣的 NPC 問題不存在。

2. 決定以下的敘述是對還是錯，並解釋對或錯的原因:

若我們能證明任何一個 NPC 問題的問題下界是多項式函數量級，則我們已經證明 $NP \neq P$ 。

錯。必須證明任何一個 NPC 問題的最壞狀況問題下界(worse case problem lower bound)為指數量級才可證明 $NP \neq P$ 。

3. 如何證明一個問題為 NP-hard?

找到存在一個已知的 NP-hard 問題，可在多項式時間內變轉至此問題，即可證明此問題為 NP-hard。

4. 寫一個 NP 演算法在給定的圖 $G=(V, E)$ 中，決定是否存在漢米爾頓迴圈(Hamiltonian circuit)。

INPUT: A graph $G=(V, E)$ with $V = \{a_1, a_2, \dots, a_n\}$

OUTPUT: **Success** if there exists a Hamiltonian circuit in G ; **Failure** otherwise

$B[0] \leftarrow \phi$

for $i \leftarrow 1$ to n do //guessing

$B[i] \leftarrow \text{choice}(V)$

$B[n + 1] \leftarrow B[1]$

for $i \leftarrow 1$ to n

 if $((B[i] \in (V - \cup_{k=0}^{i-1} B[k])) \cap (< B[i], B[i + 1] > \in E))$ then continue

 else Failure //return failure

Success //return success

以上非確定(nondeterministic)演算法之 guessing 需要 $O(n)$ 時間，而 checking 需要 $O(n^2)$ 時間。因此這個演算法是一個 NP(nondeterministic polynomial-time)演算法，可以使用 $O(n^2)$ 時間解決問題。也因此得證上述問題為 NP 問題。