**1. Given an array *N* of non-negative integers, you are initially positioned at the first index of the array. Each element in the array represents your maximum jump length at that position. Your goal is to reach the last index in the minimum number of jumps.**

**Example 1: given *N* = [2, 3, 1, 1, 4], the minimum number of jumps to reach the last index is 2 (i.e. first jump 1 step from index 0 to 1 and then jump 3 steps to the last index).**

**Example 2: given *N* = [3, 2, 1, 0, 4], the answer is NO. That is, no matter what, you will always reach the position of index 3. But the maximum jump length at this position is 0, so you can never reach the last position.**

**Design a *greedy* algorithm to solve the problem and analysis the time complexity of your algorithm.**

**Answer:**

**greedy algorithm: *O*(*N*2)**

***last* = *N*-1 \\ search backward from the last position**

***count* = 0**

***max\_jump* = 0, *max\_i* = 0**

**while *last* > 0:**

**for *i* in [0, 1, …, *last*-1]:**

**if *i* + *N*[*i*] ≥ *last*: \\ select max *N*[*i*], if *last* is reachable from *i*.**

**if *N*[*i*] > *max\_jump*: *max\_jump* = *N*[*i*], *max\_i* = *i***

***last* = *max\_i*, *count*++**

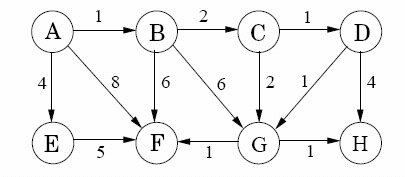
**Case study: *N* = [2, 3, 1, 1, 4]**

**max\_jump: 3 max\_i: 1**

**🡪 count: 1 last: 1**

* **max\_jump: 2 max\_i: 0**
* **count: 2 last: 0**

**2. Suppose Dijkstra's algorithm is run on the following graph, starting at node *A*.**



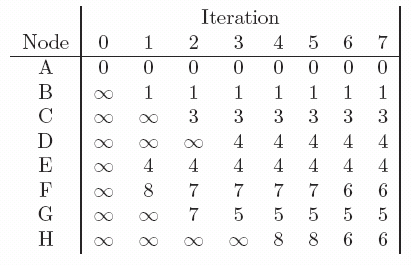
**(1) Draw a table showing the intermediate distance values of all the nodes at each iteration of the algorithm.**

**(2) Show the final shortest-path tree.**

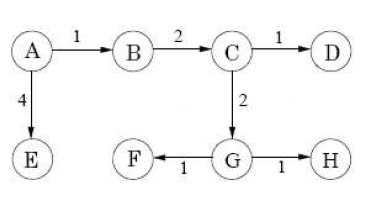
**(3) Suppose the above graph becomes undirected. What is the minimal spanning tree, by running Kruscal’s algorithm in this undirected graph?**

**Answer:**

**(a)**



**(b)**



**(c) first sort all the edges in an increasing ordering of weights:**

**A-B, C-D, D-G, G-F, G-H, B-C, C-G, A-E, D-H, E-F, B-F, B-G, A-F**

**The chosen edges and the final MST are as follows:**

**A-B, C-D, D-G, G-F, G-H, B-C, A-E**

