# Question 5

Zeal Liang

Z5325156

First, we map the array A into a graph G. The index of the array is used as the name of the node then the value of the array is used as the value of the node (number of weeks).

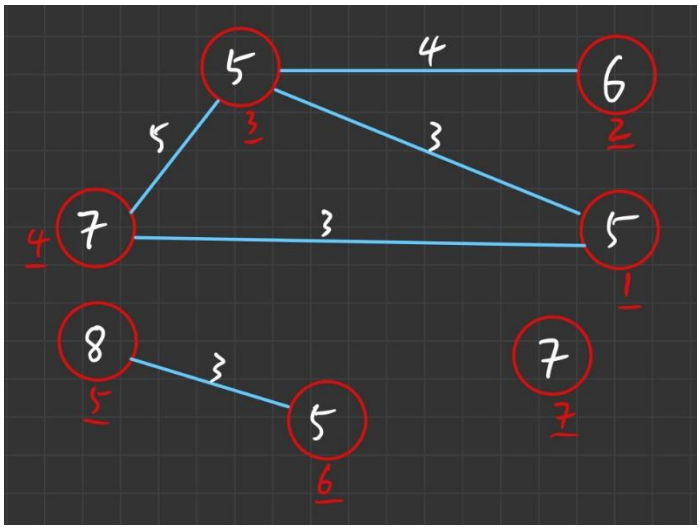
In the second step we traverse the array B, mapping each unordered pair of distinct overlapping courses and the time taken to complete onto the graph G. That is, the time of each pair of courses in array B is corresponding to the edge of the corresponding two course nodes in the graph as weights.

In the third step, we traverse the graph G using the minimum spanning tree algorithm to generate one or more minimum spanning trees $T_i$. We then calculate the sum of all the edge weights of each tree $T_i$ and add the smallest value (number of weeks) in each tree $T_i$ node. If there is more than one minimum just pick anyone. Finally add the number of weeks for all the individual components left except the minimum spanning tree and that is the answer.

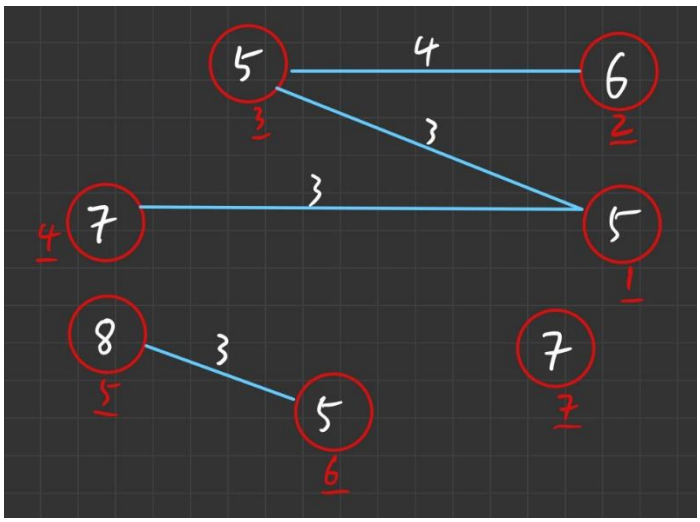
For example, if array A is [5,6,5,7,8,5,7] and

array B is [[1,4,3],[2,3,4],[3,4,5],[5,6,3],[1,3,3]]

After the first and second steps we can obtain a graph like this:




Then, after the third step of the minimum spanning tree algorithm we can obtain a graph like this:




By calculating the weights of the edges of all minimum spanning trees and adding the minimum number of weeks of each tree and the number of weeks of the remaining components we can obtain：

First MST：3+3+4+5 = 15.

Second MST：3 + 5 = 8.

Remaining components：7.

So, the total number of weeks required for this example is：15 + 8 + 7 = 30.

The time complexity required to construct the graph G in the first step is $O(n)$, and the time complexity to add edges to the graph G in the second step is $O(m)$. The time complexity of the minimum spanning tree algorithm used in the third step is $O(n + m\,log\,n)$, and then the time complexity of computing the sum of the weights of all the trees is $O(n)$. After simplification, the total time complexity of this algorithm meets the requirement of $O\big((n+m)log\,(n+m)\big)$.