

A whiteboard with writing on it

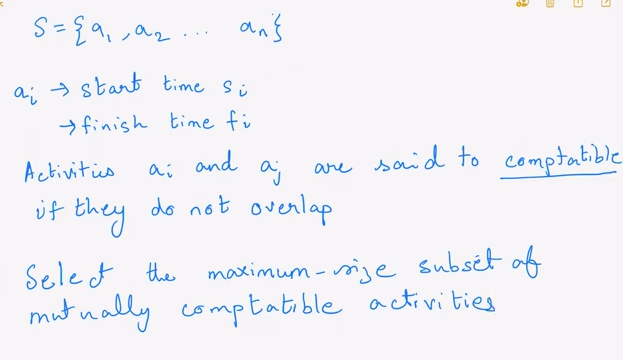
Description automatically generated

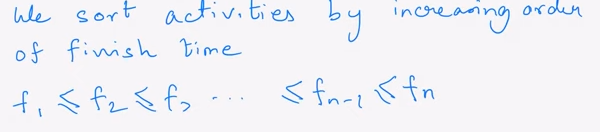
The greedy choice is to always pick the next activity whose finish time is the least among the remaining activities and the start time is more than or equal to the finish time of the previously selected activity. We can sort the activities according to their finishing time so that we always consider the next activity as the minimum finishing time activity

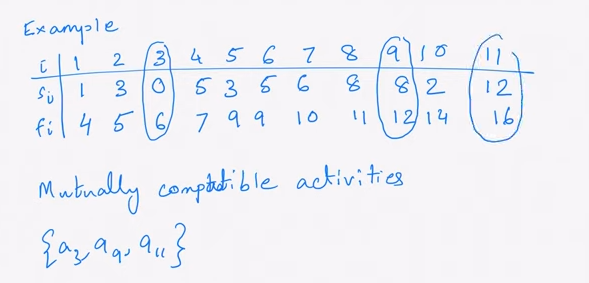
Follow the given steps to solve the problem:

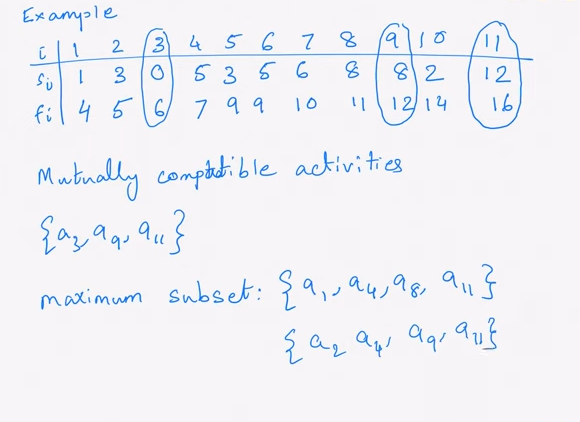
* Sort the activities according to their finishing time
* Select the first activity from the sorted array and print it
* Do the following for the remaining activities in the sorted array
  + If the start time of this activity is greater than or equal to the finish time of the previously selected activity then select this activity and print it

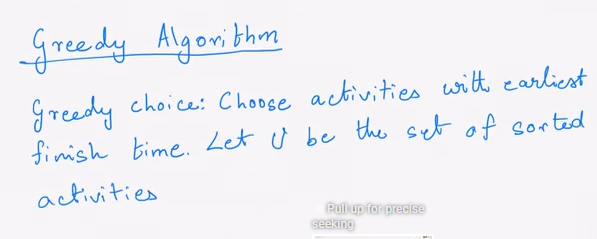
**Note:** In the implementation, it is assumed that the activities are already sorted according to their finish time, otherwise the time complexity will rise to O(N\*log(N)) and Auxiliary Space will rise to O(N), as we have to create a 2-D array for storing the start and finish times together.

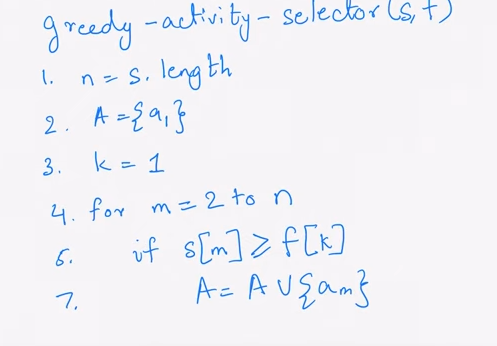


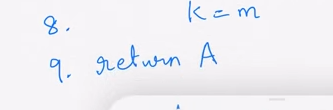












f