Activity Selection Problem

There are n different activity are given with their starting and ending time. Select maximum number of activity to solve by a single person.

1- Sort the activity with their ending time.

2- Find compatible activity and add to list.

$$\downarrow \qquad \downarrow$$
$$n + n \log n$$
$$O(n \log n)$$

Algorithm

Greedy -activity (s, f)

1- n ← length (s)

2- A ← {1}

3- J ← 1

4- for i ← 2 to n

5-   do if $S_i \geqslant f_J$

6-   then A ← A U {i}

7-   J ← i

8- Return A

Question-1 Activity Selection Problem Example

Start Time (s)	Finish Time (F)	Activity Name
5	9	A_1
1	2	A_2
3	4	A_3
0	6	A_4
5	7	A_5
8	8	A_6

Solution:

Step-1: Sort the given activities in ascending order according to their finishing time.

The table after we have sorted it:

Start Time (s)	Finish Time (F)	Activity Name
1	2	A_2
3	4	A_3
0	6	A_4
5	7	A_5
5	9	A_1
8	9	A_6

Step 2: Select the First activity from Sorted Array act[] and add it to the sol[] array, thus sol = {a2}

Step 3: Repeat step 4 and 5 for the remaining activities in act[]

Step 4: If the ~~first~~ start time of the currently selected activity is greater than or equal to the finish time of the previously selected activity, then add it to sol[]

Step 5: Select the next activity in act[]

For the data given in the above Table,

A Select activity A3. Since the start time of A3 is greater than the finish time of A2 i.e. $[s(A3) > f(A2)]$

We add A3 to the Solution set.
Thus sol = {A2, A3}

B Select A4. Since $[s(A4) < f(A3)]$, it is not added to the solution set.

C Select A5. Since $[s(A5) > f(A3)]$, A5 gets added to the solution set.
Thus sol = {A2, A3, A5}

D. Select A1, Since $S(A1) < F(A5)$, A1 is not added to the solution set

E. Select A6, A6 is added to the solution set since

$$S(A6) > F(A5)$$

Thus Sol = $\{A2, A3, A5, A6\}$

Step 6: At last, print the array. sol[]

Hence, the execution schedule of maximum number of non-conflicting activities will be:

$(1, 2)$ $(3, 4)$ $(5, 7)$ $(8, 9)$