Let S be a schedule of n activities, each with a start time s_i and finish time f_i. Two activities i and j overlap if $s_i \leq s_j < f_i$ or $s_j \leq s_i < f_j$. That is, one starts then the other one starts before the first one finishes. We want to select a subset of activities that do not overlap — let's call such a subset *a proper itinerary.* We want to find a proper itinerary that has the largest number of activities.

We use the following greedy algorithm:

1. Pick the activity with the earliest (smallest) finish time
2. Remove activities that overlap with the activity selected in Step 1.
3. Repeat the process with the remaining activities.

Claim: The greedy algorithm above finds a proper itinerary with the largest number of activities.

Proof (by induction): We induct on n, the number of activities in the schedule S.

Induction Hypothesis $P(n)$: The greedy algorithm finds a proper itinerary with the largest number of activities in any schedule that has n activities.

Base Case: We use $n = 1$ as the base case. Since there is only one activity in the schedule, the only proper itinerary has one activity. The greedy algorithm will select that activity. Thus, the greedy algorihtm finds the largest proper itinerary in the base case.

Induction Step: Let $g_1, g_2, \ldots, g_t$ be the itinerary selected by the greedy algorithm from a schedule S with $n + 1$ activities. Suppose by way of contradiction that there exists a proper itinerary with more than t activities: $x_1, x_2, \ldots, x_{t+1}$. Without loss of generality, assume that the g_i and x_i are sorted by finish times.

Since g_1 was chosen to have the smallest finish time, we know that

$$f_{g_1} \leq f_{x_1}.$$

Now, since x_1 does not overlap with $x_2, \ldots, x_{t+1}$ and since x_1 has the smallest finish time among $x_1, \ldots, x_{t+1}$, we can also know that for all i, $2 \leq i \leq t + 1$, that

$$f_{x_1} \leq s_{x_i}.$$

That means, $f_{g_1} \leq s_{x_i}$ for all i, $2 \leq i \leq t + 1$. That is, g_1 does not overlap with $x_2, \ldots, x_{t+1}$. Thus, $g_1, x_2, \ldots, x_{t+1}$ is also a proper itinerary for S. (That is, we can swap g_1 for x_1.)

Now, let $S' \subseteq S$ be the set of activities that do not overlap with g_1. Since S' does not contain g_1, it has at most n activities. Then, by the induction hypothesis (using "strong" induction), we know that the greedy algorithm produces a largest proper itinerary for S'. The activities selected by the greedy algorithm for S' are exactly: $g_2, g_3, \ldots g_t$. (Check this.)

Finally, consider $x_2, \ldots, x_{t+1}$. Since $x_2, \ldots, x_{t+1}$ do not overlap with g_1, they are also in S'. Also, none of $x_2, \ldots, x_{t+1}$ overlap with each other. Thus, $x_2, \ldots, x_{t+1}$ is a proper itinerary for S'. However, $x_2, \ldots, x_{t+1}$ has more activities than the greedy solution $g_2, \ldots, g_t$. This contradicts the induction hypothesis. Therefore, there cannot exist a proper itinerary $x_1, \ldots, x_{t+1}$ for S and $g_1, \ldots, g_t$ is a largest proper itinerary for S. Thus, we have established the induction hypothesis $P(n + 1)$ and have also proven the claim. $\square$