CSE101: Algorithm Design and Analysis

Ragesh Jaiswal, CSE, UCSD
Greedy Algorithms
“A local (greedy) decision rule leads to a globally optimal solution.”

There are two ways to show the above property:

- Greedy stays ahead
- Exchange argument
Interval scheduling: Given a set of $n$ intervals of the form $(S(i), F(i))$, find the largest subset of non-overlapping intervals.
Greedy Algorithms
Interval scheduling

Problem

Interval scheduling: Given a set of $n$ intervals of the form $(S(i), F(i))$, find the largest subset of non-overlapping intervals.

- Candidate greedy choices:
  - Earliest start time
  - Smallest duration
  - Least overlapping
Greedy Algorithms
Interval scheduling

Problem
Interval scheduling: Given a set of $n$ intervals of the form $(S(i), F(i))$, find the largest subset of non-overlapping intervals.

- Candidate greedy choices:
  - Earliest start time
  - Smallest duration
  - Least overlapping
  - Earliest finish time

![Diagram of intervals]

Ragesh Jaiswal, CSE, UCSD
CSE101: Algorithm Design and Analysis
**Greedy Algorithms**

**Interval scheduling**

<table>
<thead>
<tr>
<th>Problem</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interval scheduling: Given a set of $n$ intervals of the form $(S(i), F(i))$, find the largest subset of non-overlapping intervals.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>GreedySchedule</td>
</tr>
<tr>
<td>- Initialize $R$ to contain all intervals</td>
</tr>
<tr>
<td>- While $R$ is not empty</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
**Problem**

Interval scheduling: Given a set of $n$ intervals of the form $(S(i), F(i))$, find the largest subset of non-overlapping intervals.

**Algorithm**

GreedySchedule
- Initialize $R$ to contain all intervals
- While $R$ is not empty
  - Choose an interval $(S(i), F(i))$ from $R$ that has the smallest value of $F(i)$
  - Delete all intervals in $R$ that overlaps with $(S(i), F(i))$

Question: Let $O$ denote some optimal subset and $A$ by the subset given by GreedySchedule. Can we show that $A = O$?
Question: Let $O$ denote some optimal subset and $A$ by the subset given by GreedySchedule. Can we show that $A = O$?

Question: Can we show that $|O| = |A|$?
Question: Let $O$ denote some optimal subset and $A$ by the subset given by GreedySchedule. Can we show that $A = O$?

Question: Can we show that $|O| = |A|$?

Yes we can! We will use “greedy stays ahead” method to show this.

Proof

Let $a_1, a_2, ..., a_k$ be the sequence of requests that GreedySchedule picks and $o_1, o_2, ..., o_l$ be the requests in $O$ sorted in non-decreasing order by finishing time.

Claim 1: $F(a_1) \leq F(o_1)$. 
Question: Let $O$ denote some optimal subset and $A$ by the subset given by GreedySchedule. Can we show that $A = O$?

Question: Can we show that $|O| = |A|$?

Yes we can! We will use “greedy stays ahead” method to show this.

**Proof**

Let $a_1, a_2, ..., a_k$ be the sequence of requests that GreedySchedule picks and $o_1, o_2, ..., o_l$ be the requests in $O$ sorted in non-decreasing order by finishing time.

- **Claim 1:** $F(a_1) \leq F(o_1)$.
- **Claim 2:** If $F(a_1) \leq F(o_1)$, $F(a_2) \leq F(o_2)$, ..., $F(a_{i-1}) \leq F(o_{i-1})$, then $F(a_i) \leq F(o_i)$. 
Question: Let $O$ denote some optimal subset and $A$ by the subset given by GreedySchedule. Can we show that $A = O$?

Question: Can we show that $|O| = |A|$?

Yes we can! We will use “greedy stays ahead” method to show this.

Proof

Let $a_1, a_2, ..., a_k$ be the sequence of requests that GreedySchedule picks and $o_1, o_2, ..., o_l$ be the requests in $O$ sorted in non-decreasing order by finishing time.

We will show by induction that $\forall i, F(a_i) \leq F(o_i)$

Claim 1 (base case): $F(a_1) \leq F(o_1)$.

Claim 2 (inductive step): If $F(a_1) \leq F(o_1), F(a_2) \leq F(o_2), ..., F(a_{i-1}) \leq F(o_{i-1})$, then $F(a_i) \leq F(o_i)$.

GreedySchedule could not have stopped after $a_k$. 
Greedy Algorithms
Interval scheduling

Problem
Interval scheduling: Given a set of $n$ intervals of the form $(S(i), F(i))$, find the largest subset of non-overlapping intervals.

Algorithm
GreedySchedule
- Initialize $R$ to contain all intervals
- While $R$ is not empty
  - Choose an interval $(S(i), F(i))$ from $R$ that has the smallest value of $F(i)$
  - Delete all intervals in $R$ that overlaps with $(S(i), F(i))$

- Running time?
### Problem

Interval scheduling: Given a set of $n$ intervals of the form $(S(i), F(i))$, find the largest subset of non-overlapping intervals.

### Algorithm

**GreedySchedule**

- While $R$ is not empty
- Choose an interval $(S(i), F(i))$ from $R$ that has the smallest value of $F(i)$
- Delete all intervals in $R$ that overlaps with $(S(i), F(i))$

- Running time? $O(n \log n)$
End