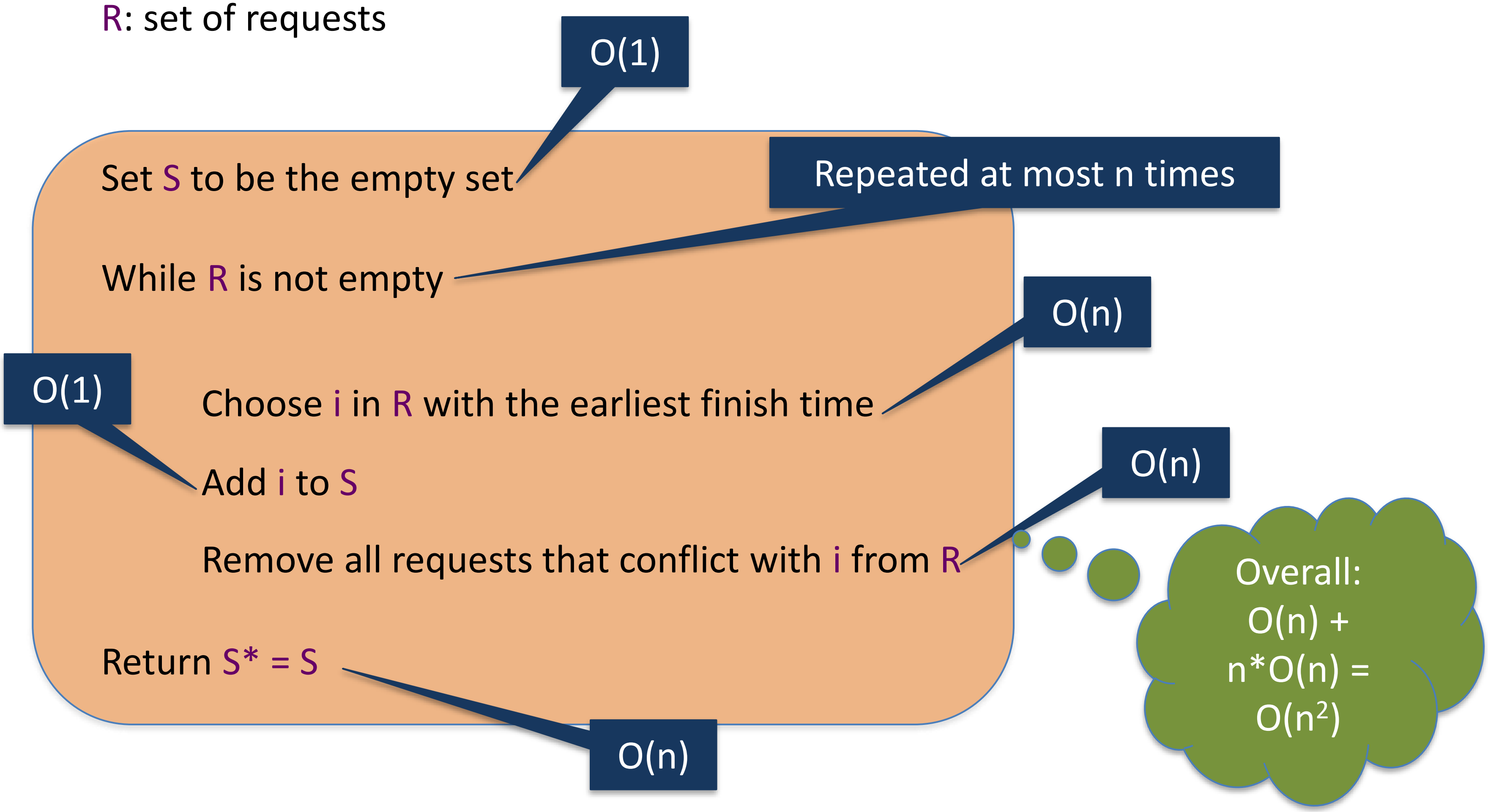
Lecture 17

CSE 331

Runtime analysis of Greedy Algo.

R: set of requests



```
Set S to be the empty set
While R is not empty
    Choose i in R with the earliest finish time
    Add i to S
    Remove all requests that conflict with i from R
Return S* = S
```

O(1)

Repeated at most n times

O(n)

O(1)

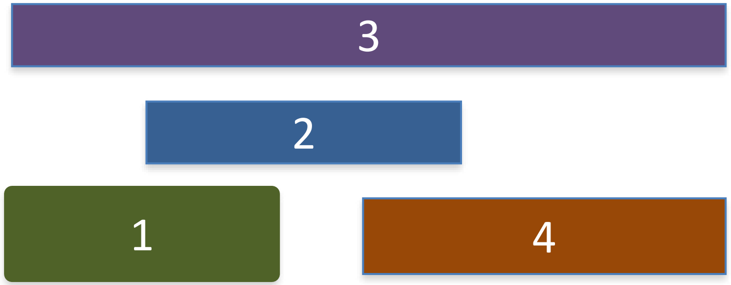
O(n)

O(n)

Overall:
$O(n) + n*O(n) = O(n^2)$

Algorithm implementation

Go through the intervals in order of their finish time



Check if $s[i] < f(1)$

with 1?

In general, if jth interval is the last one chosen

Pick smallest $i>j$ such that $s[i] \geq f(j)$

$O(n \log n)$ run time

The final algo

$O(n \log n)$ time sort intervals such that $f(i) \leq f(i+1)$

$O(n)$ time build array $s[1..n]$ s.t. $s[i]$ = start time for i

```
Add $1$ to $A$ and set $f = f(1)$

For $i = 2 .. n$

        If $s[i] \geq f$

            Add $i$ to $A$

            Set $f = f(i)$

Return $A^* = A$
```

Reading Assignment

Sec 4.1 of [KT]