Lecture 33

CSE 331

Nov 19, 2018

Thanksgiving break

note | 0 views

Over the thanksgiving break

Hope y'all have a great thanksgiving break!

Your 331 staff will also be on break during the Thanksgiving break and in practice this means that we will either not respond on piazza Wed-Sun or our response will be *very* delayed.

We'll still be around on till Tue and starting Mon, Mov 26. (Note that HW 10 is due Nov 29.)

piazza

edit · good note | 0

Updated Just now by Atri Rudra

Sample Final exam

note

0 views

Actions

Sample final exam

This is a bit early but I figured I'll release the sample final exam in case you were planning to prepare for the final exam over the break (and having the sample final helps):

- Sample final
- Sample final solutions

(These are also available under the "Sample Exams" dropdown menu from the banner on the 331 webpage.)

Two comments:

- I would recommend that you not peek at the solution before you have worked on the sample final on your own.
- As with the sample mid-terms, do **not** try and deduce anything about the topic coverage in the actual final exam (will post on how to prepare for the final exam by tonight).
    - However, the sample exam was an actual final exam in one of the past years. Your final exam will be of comparable difficulty.

final

edit · good note | 0

Updated Just now by Atri Rudra

Final exam: Dec 10

note

stop following **2 views**

Actions

Final exam post

I'll start off with some generic comments:

- The final exam will be based on all the material we will see in class up to the P vs NP stuff (we'll most likely finish that stuff by Monday, Dec 3 or in the worst-case by Wednesday, Dec 5).
- The lecture on Friday, Dec 7 will be a Q & A session (where you can ask any 331 related questions)-- stay tuned for more details.
- Exam will be from **8:30 to 11:00am** on Monday, **Dec 10** in class (**Norton 112**). Note that the exam will be for 2.5 hours and *not 3 hours* as it says on HUB.
  - If you have three of more exams scheduled on Dec 10, please contact me **NO later than 5PM on Monday, DECEMBER 3.** *If you contact me after Dec 3, I won't be able to accommodate any re-scheduling request.*

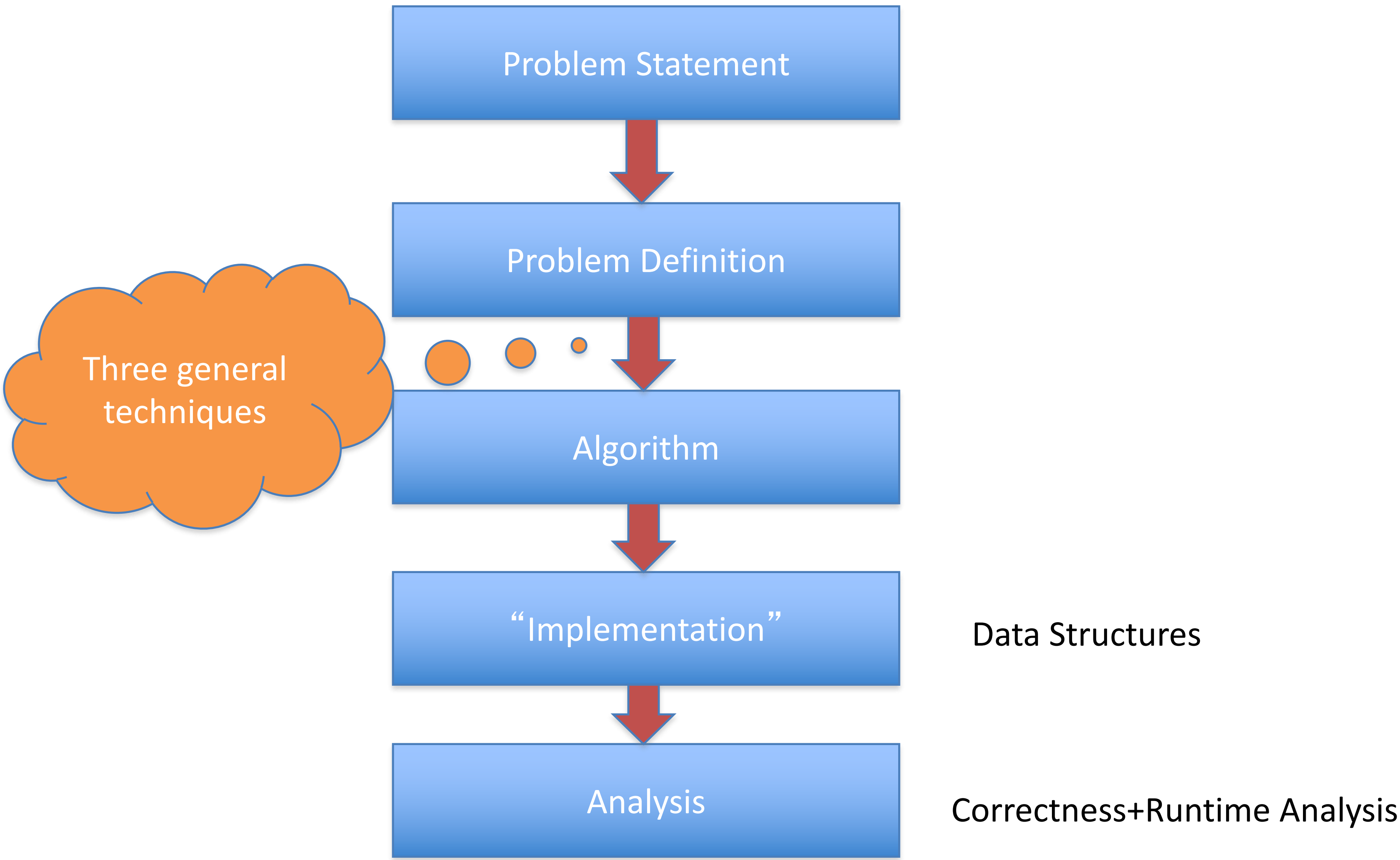
Next are comments related to **preparing for the finals**:

1. Take a look at the sample final (@975) and spend some quality time solving it. Unlike the homeworks, it might be better to try to do this on your own. Unlike the sample mid-term, this one is an actual 331 final exam so in addition to the format, you can also gauge how hard the final exam is going to be (your final exam will be the same ballpark). However as with the sample mid-term, you make deductions about the coverage of topics at your own peril (but see points below). Once you have spent time on it on your own, take a look at the sample final solutions (@975).
2. We will have some extra OHs on Friday, Dec 7 (stay tuned for more details).
3. Attend the Q&A session (Friday, Dec 7) in class.
4. The actual final will have the same format as the sample final: The first question will be T/F, 2nd will be T/F with justification, the rest of the three will be longer questions and will ask you to design algorithms (parts of them might be just *analyzing* an algorithm.)

Graded HW 7

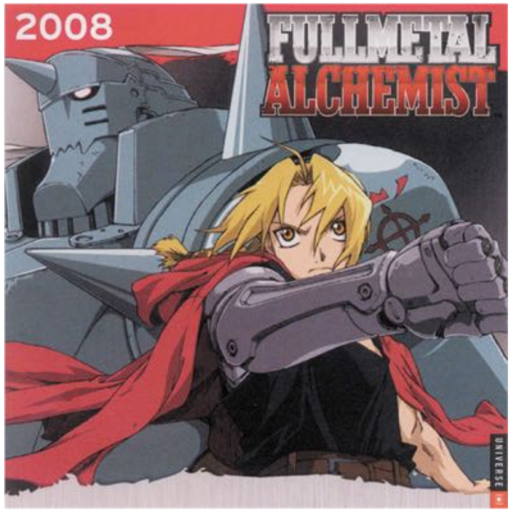
Should be done by tonight

High level view of CSE 331



Greedy Algorithms

Natural algorithms



Reduced exponential running time to polynomial

Divide and Conquer

Recursive algorithmic paradigm

Reduced large polynomial time to smaller polynomial time

A new algorithmic technique

Dynamic Programming

Dynamic programming vs. Divide & Conquer

Same same because

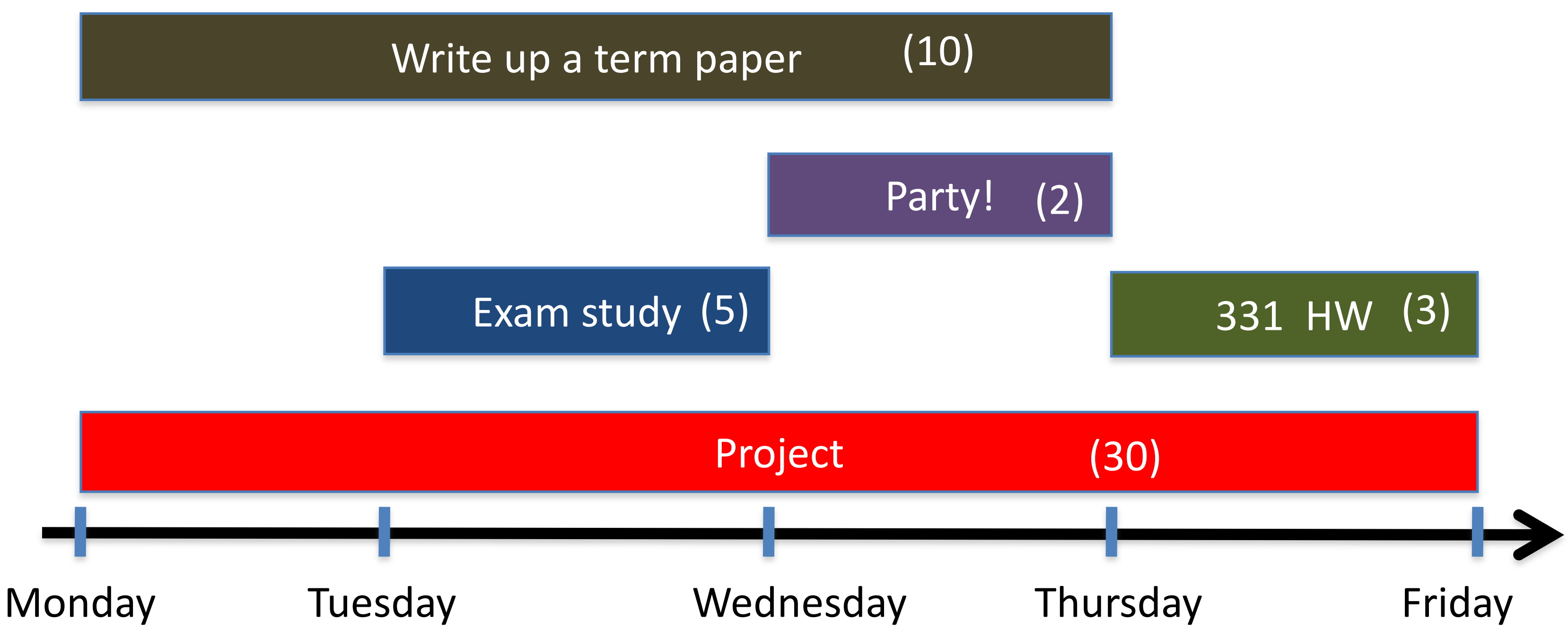
Both design recursive algorithms

Different because

Dynamic programming is smarter about solving recursive sub-problems

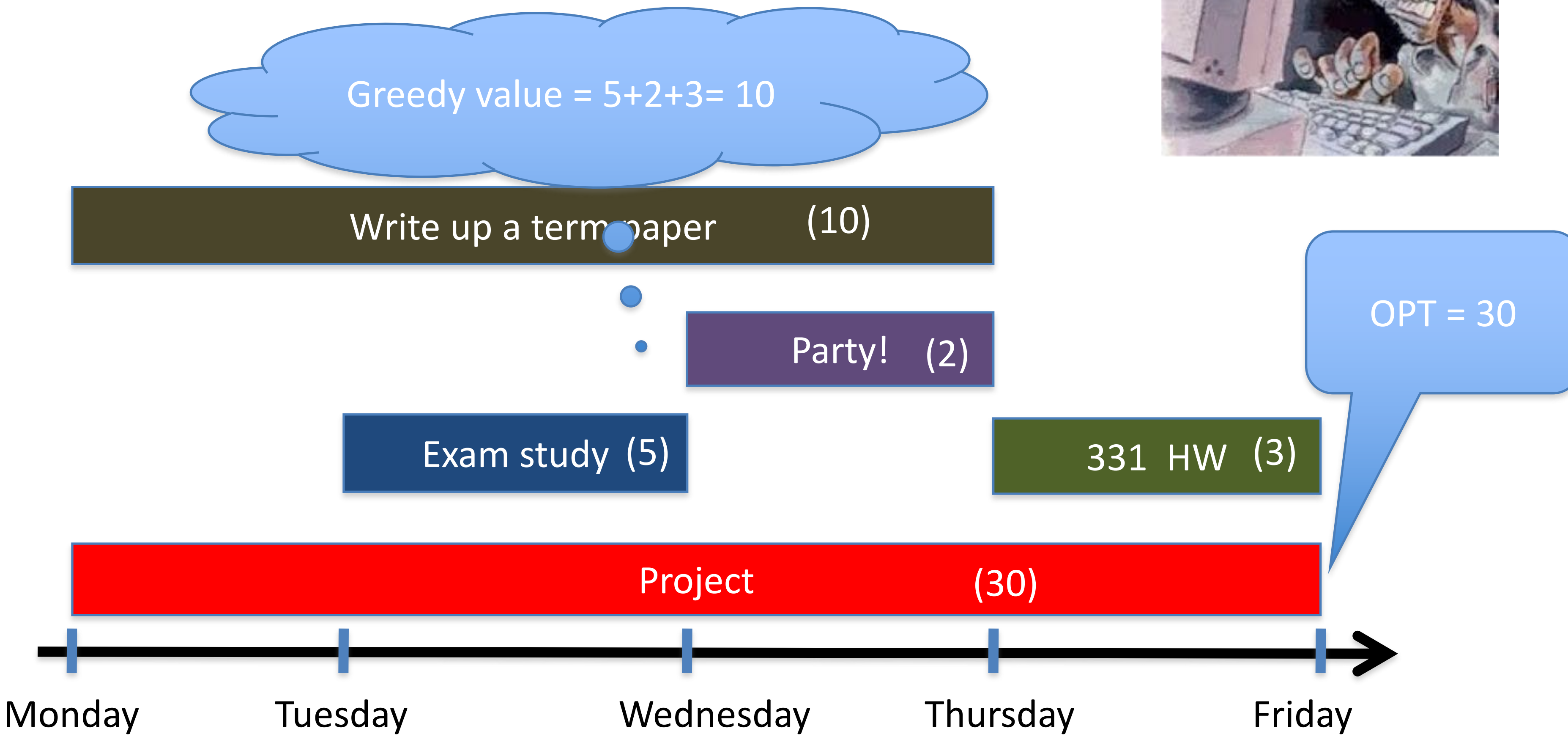
End of Semester blues

Can only do one thing at any day: what is the optimal schedule to obtain maximum value?



Previous Greedy algorithm

Order by end time and pick jobs greedily



Today’s agenda

Formal definition of the problem

Start designing a recursive algorithm for the problem

Weighted Interval Scheduling

Input: n jobs/intervals. Interval i is triple (s_i, f_i, v_i)

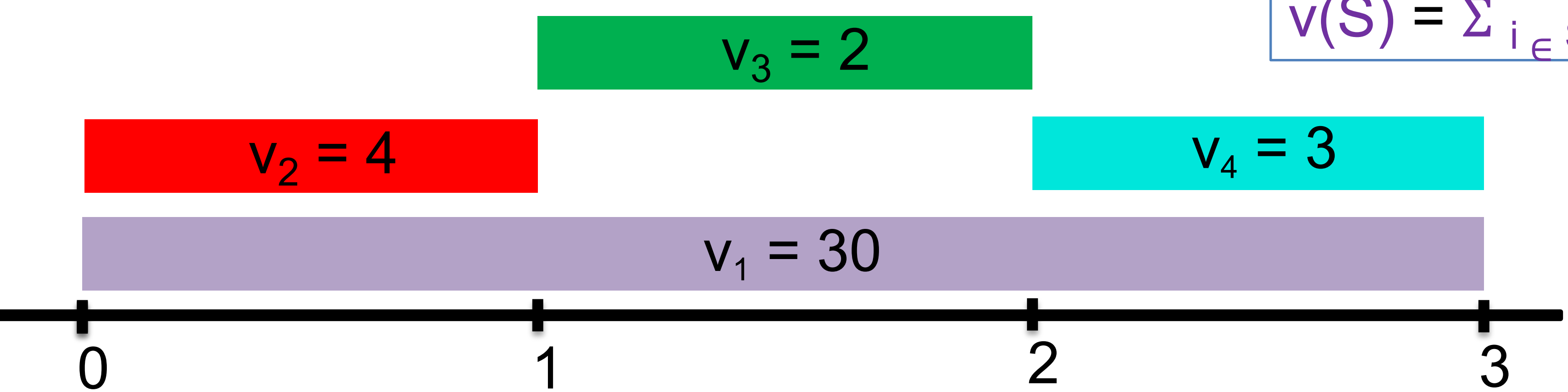
start time

finish time

value

Output: A valid schedule $S \subseteq [n]$ that maximizes $v(S)$

$v(S) = \Sigma_{i \in S} v_i$



Previous Greedy Algorithm

R = original set of jobs

$S = \phi$

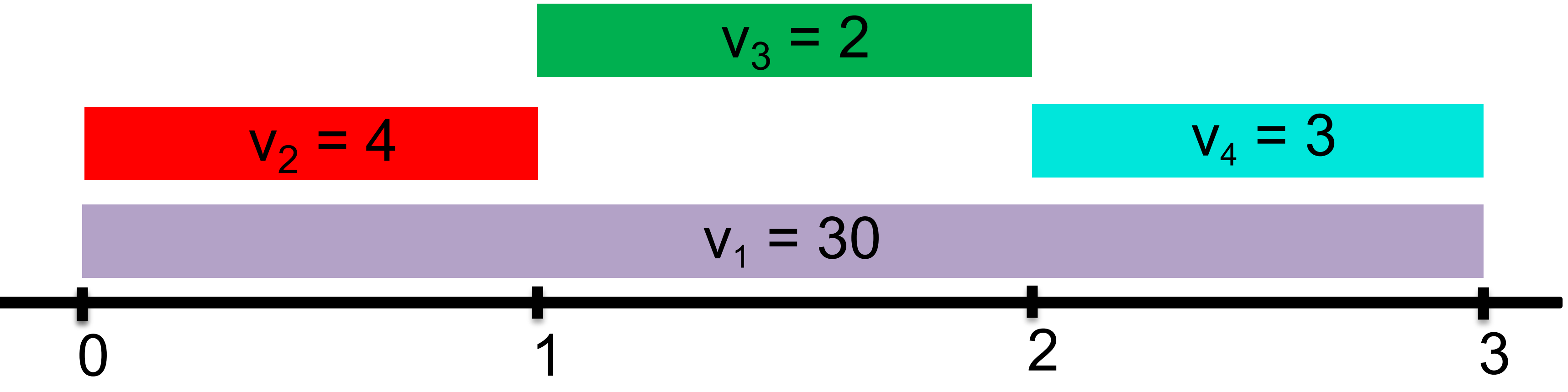
While R is not empty

    Choose i in R where f_i is the smallest

    Add i to S

    Remove all requests that conflict with i from R

Return $S^* = S$



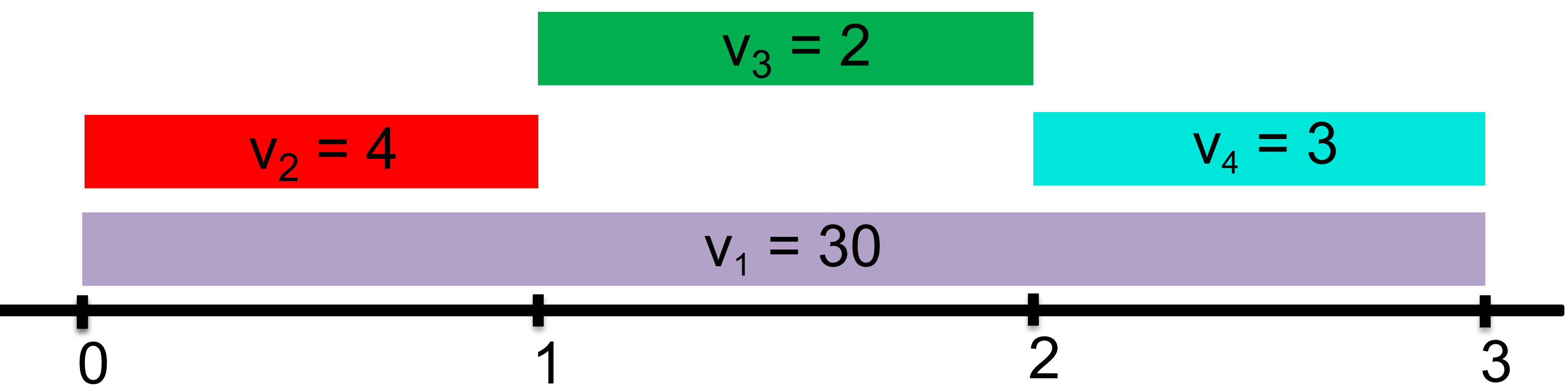
Perhaps be greedy differently?

R = original set of jobs

S = ϕ

While R is not empty
 Choose i in R where $v_i/(f_i - s_i)$ is the largest
 Add i to S
 Remove all requests that conflict with i from R

Return S* = S



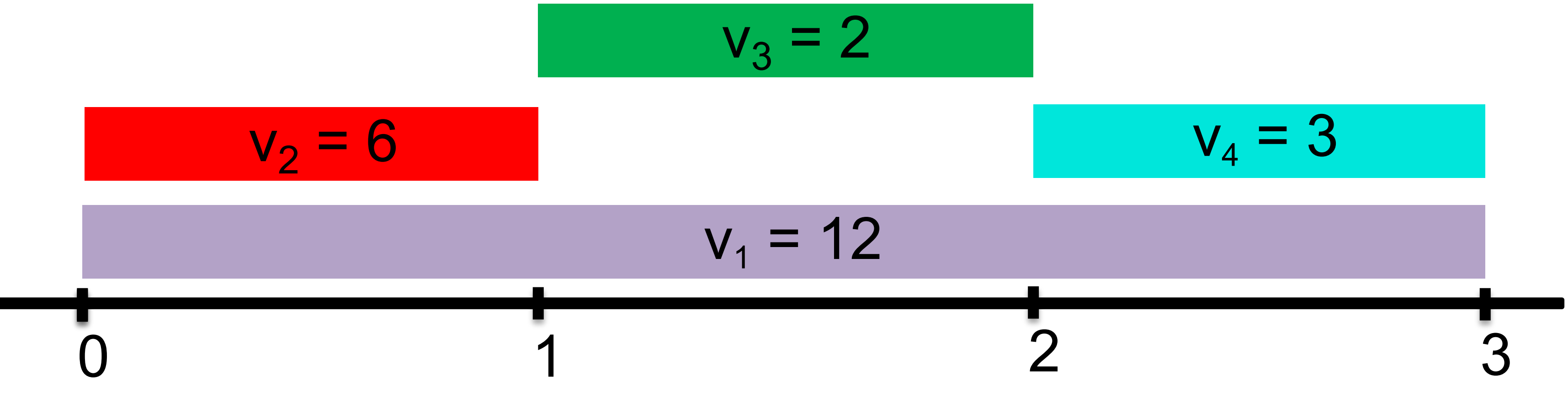
Can this work?

R = original set of jobs

S = ϕ

While R is not empty
    Choose i in R where $v_i/(f_i - s_i)$ is the largest
    Add i to S
    Remove all requests that conflict with i from R

Return S* = S



Avoiding the greedy rabbit hole

https://www.writerightwords.com/down-the-rabbit-hole/

Provably IMPOSSIBLE for a large class of greedy algos

There are no known greedy algorithm to solve this problem

Perhaps a divide & conquer algo?

Divide the problem in 2 or more many EQUAL SIZED INDEPENDENT problems

Recursively solve the sub-problems

Patchup the SOLUTIONS to the sub-problems

Perhaps a divide & conquer algo?

RecurWeightedInt([n])

if n = 1 return the only interval

L = first n/2 intervals

R = last n/2 intervals

S_L = RecurWeightedInt(L)

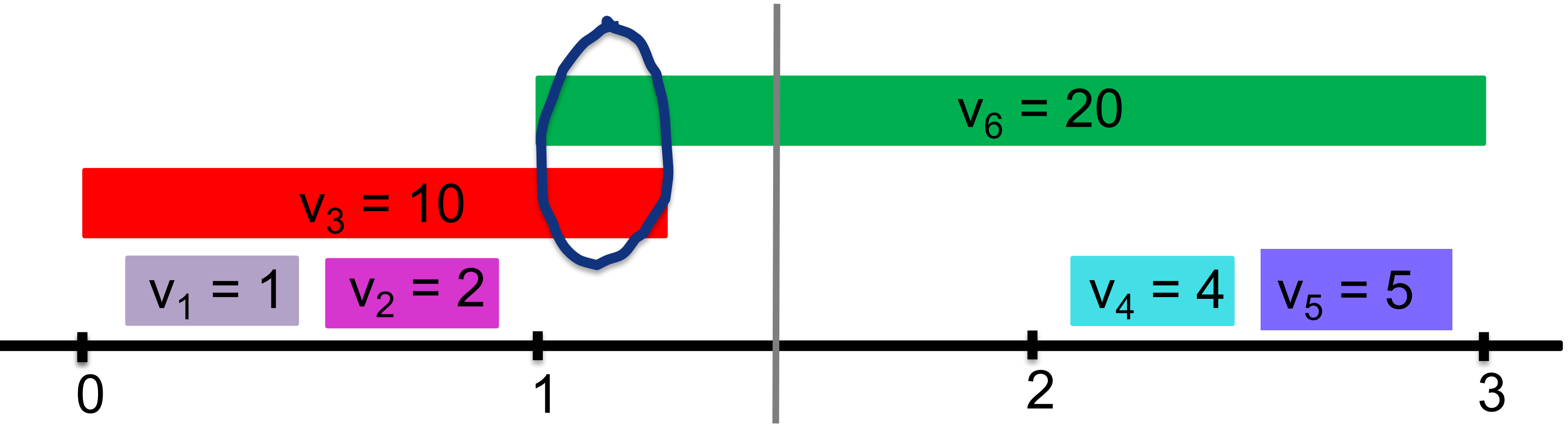
S_R = RecurWeightedInt(R)

PatchUp(S_L, S_R)

Would this general scheme work?

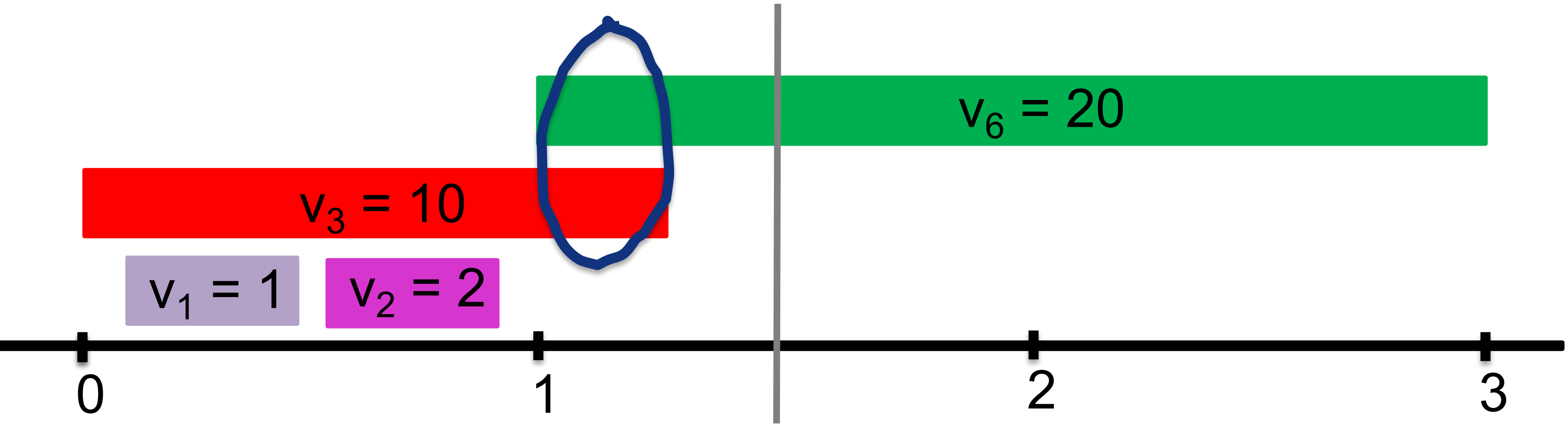
Divide the problem in 2 or more many EQUAL SIZED INDEPENDENT problems

Sub-problems NOT independent!

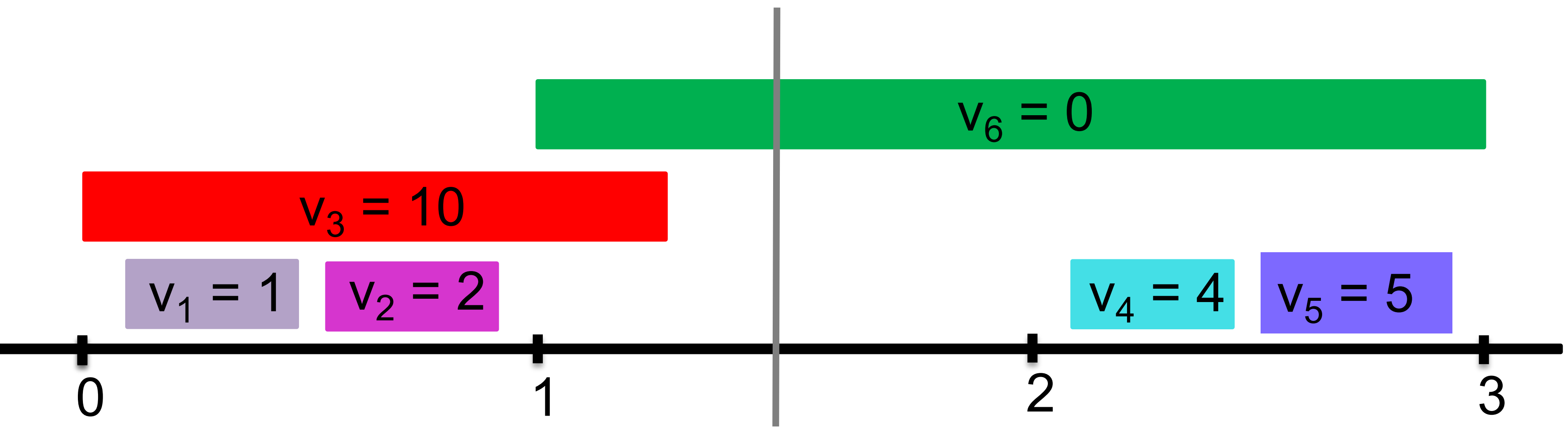


Perhaps patchup can help?

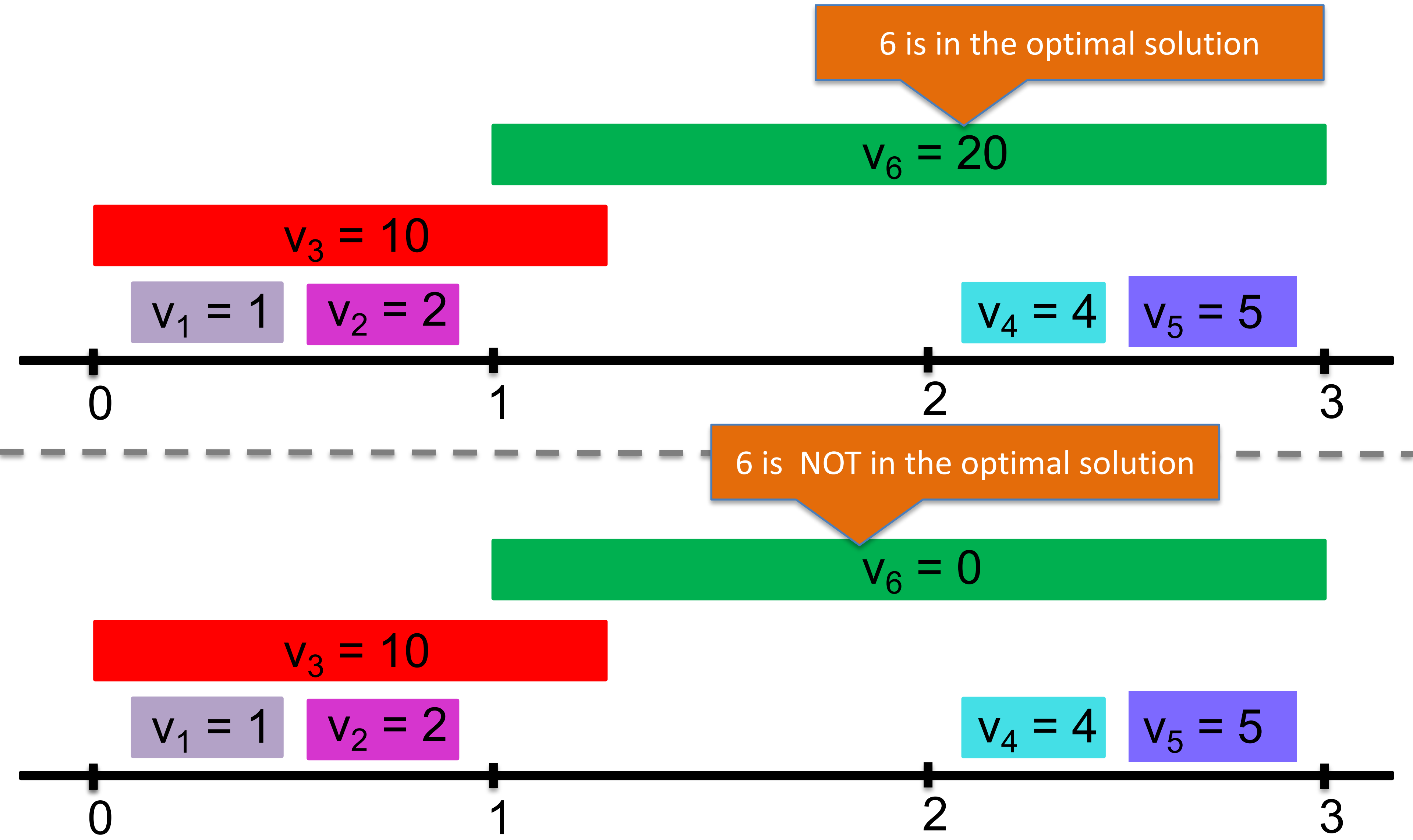
Patchup the SOLUTIONS to the sub-problems



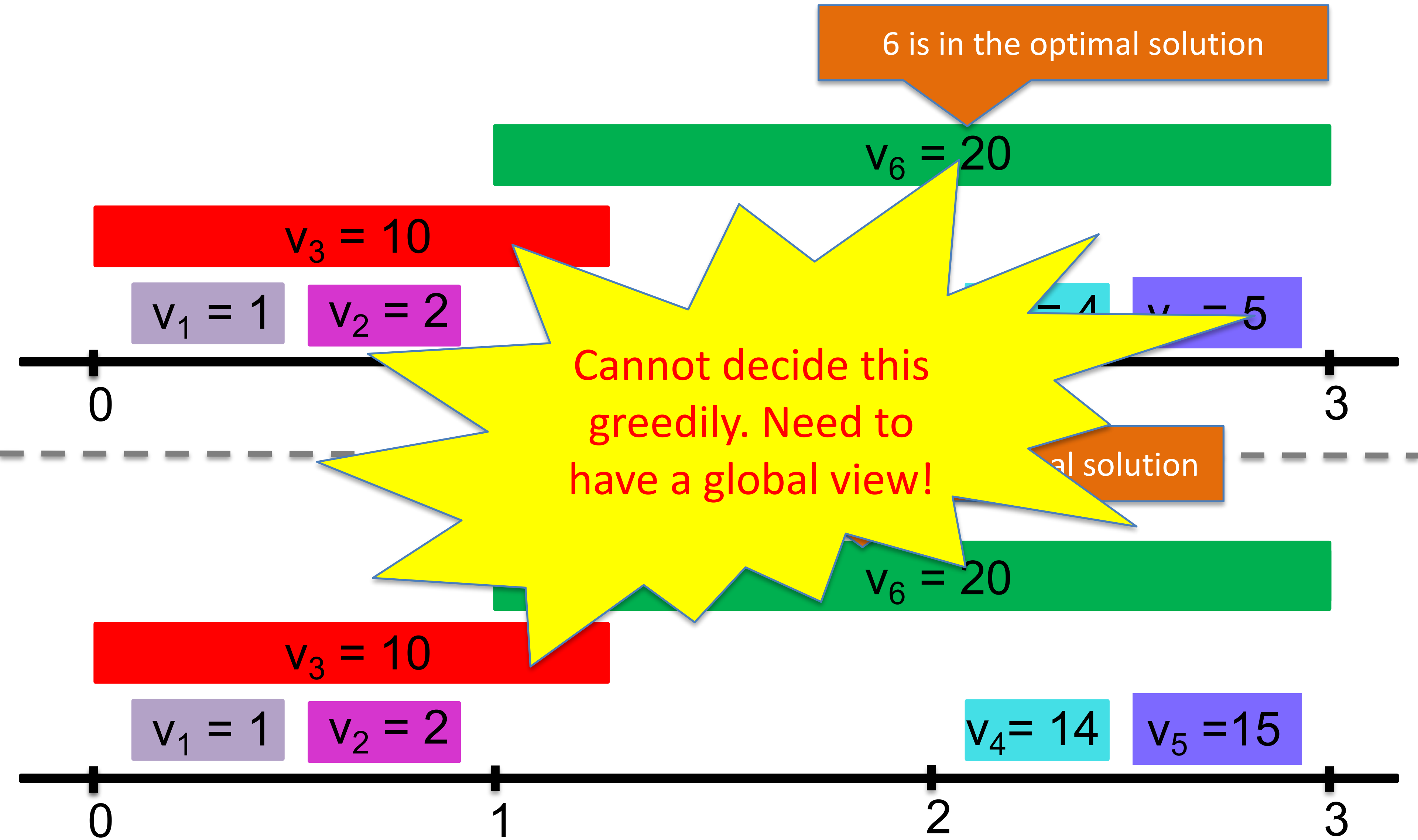
Sometimes patchup NOT needed!



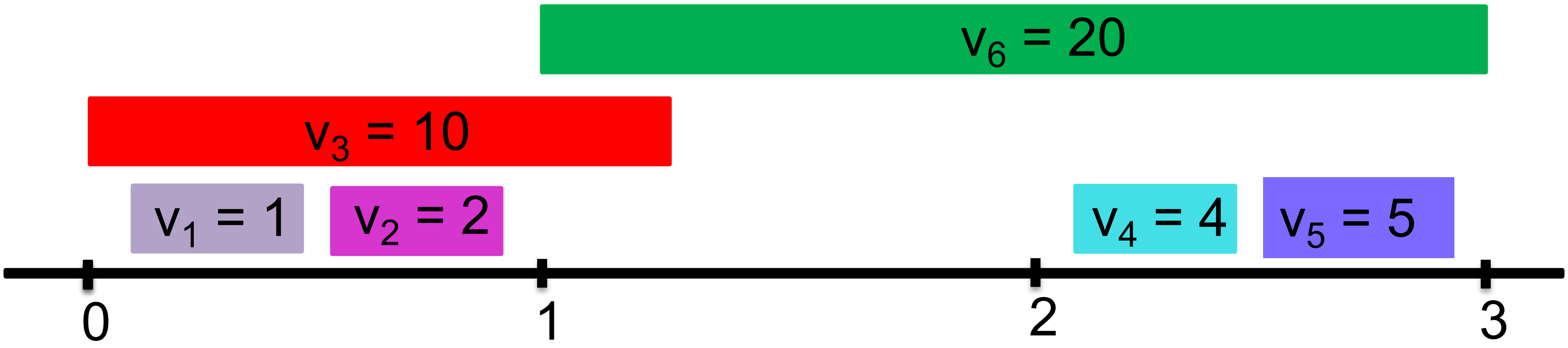
Check for two cases?



Check if v_6 is the largest value?



Check out both options!



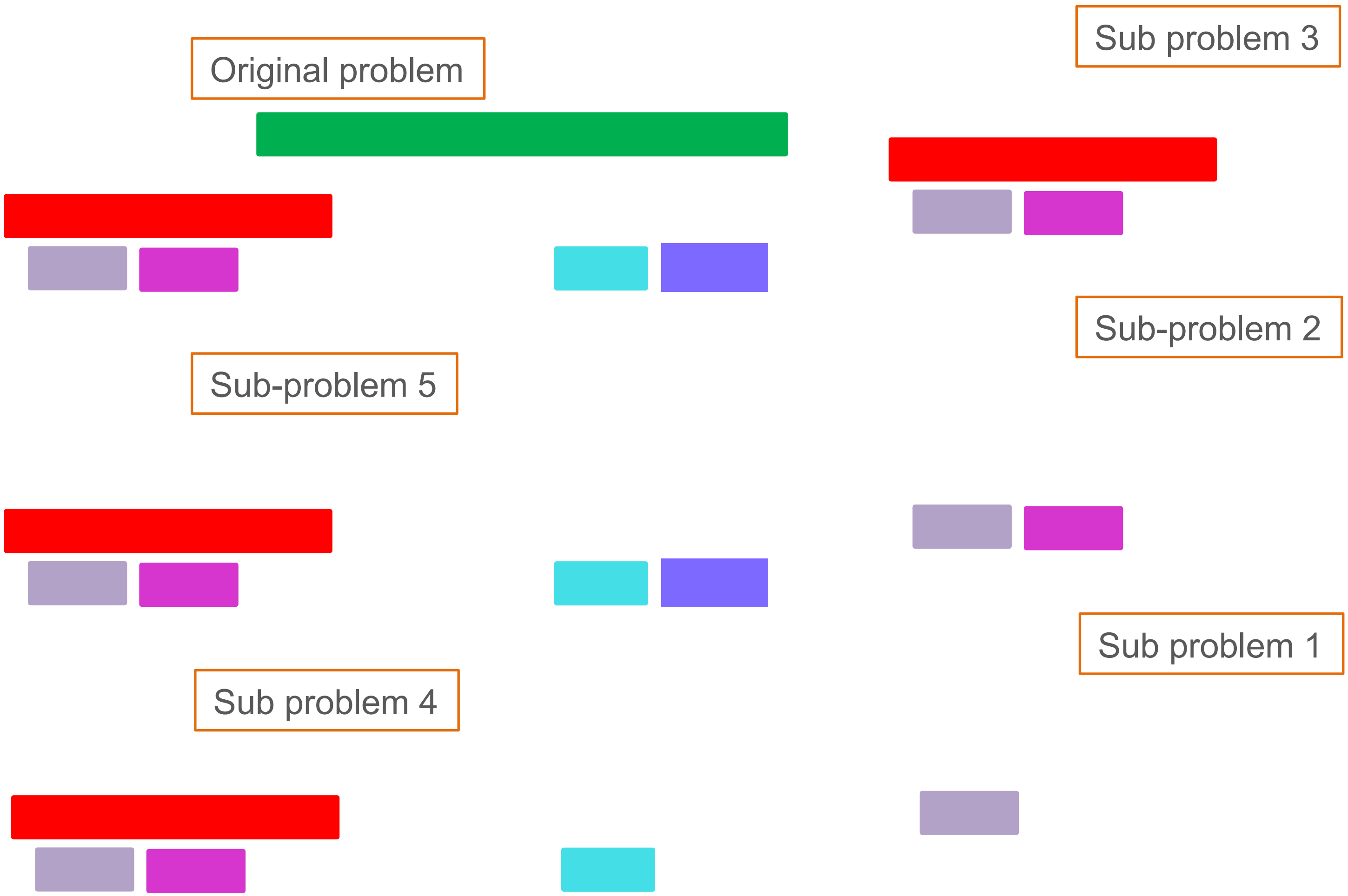
Case 1: 6 is in the optimal solution

6 is not in optimal solution


?
ICANHASCHEEZBURGER.COM

So what sub-problems?

Divide the problem in 2 or more many ~~EQUAL SIZED~~ ~~INDEPENDENT~~ problems




?
ICANHASCHEEZBURGER.COM