CS 561: Algorithms & Data Structures
Homework 04

Warren D. (Hoss) Craft

Fall 2020 (due Mon 10/26/20)

Preamble: Space here for some notes, before the enumerated problems below. See end of document for example Figures and Tables. While we're at it, here's an example citation from the bibliography: Cormen [1]

(1) Consider the following alternative greedy algorithms for the activity selection problem discussed in class. For each algorithm, either prove or disprove that it constructs an optimal schedule.

(a) Choose an activity with shortest duration, discard all conflicting activities and recurse.

(b) Choose an activity that starts first, discard all conflicting activities and recurse.

Solutions.

(a) *Shortest Duration.* Consider Figure (1), showing a collection of "disjoint in time" activities a_1, a_2, a_3, a_4, a_5, all approximately equal in duration and a special activity a' of noticeably shorter duration ....

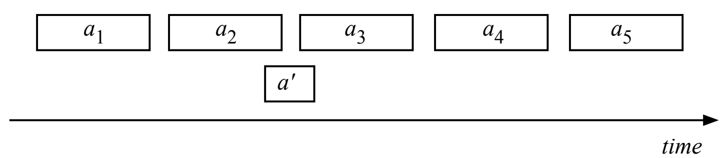


Figure 1: See Problem 1(a). Add more text here for figure caption.

(b) *Starts First.* Consider Figure (2), showing a collection of activities $a_1, a_2, a_3, a_4, a_5, a'$. The original "Ends First" greedy algorithm would produce the 3-element activity set $\{a_1, a_3, a_4\}$, but the "Starts First" greedy algorithm would produce the single-element activity set $\{a'\}$. Clearly then, the "Starts First" greedy algorithm will not generally produce an optimal schedule.

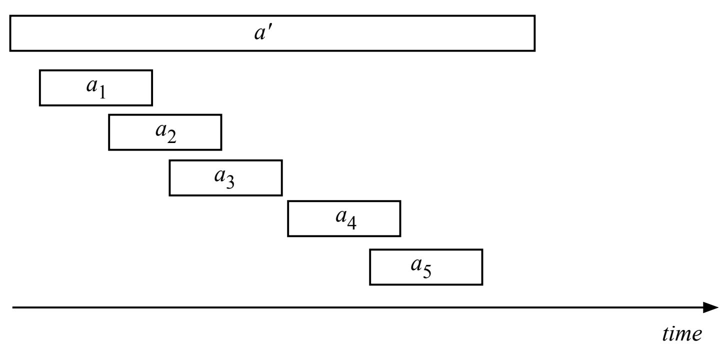


Figure 2: See Problem 1(a). A collection of activities where the "Ends First" greedy algorithm would produce the 3-element activity set $\{a_1, a_3, a_4\}$, but the "Starts First" greedy algorithm would produce the 1-element activity set $\{a'\}$.

(2) Second problem starts here.

(a) Start sub-problem here.

(b) Start sub-problem here.

Solutions.

(a) Solution to 2(a): ....

(b) Solution to 2(b): ....

EXAMPLES for use elsewhere . . .

$$m(j) = \max \begin{cases} m(x_j) + v_j \\ m(j-1) \end{cases} \tag{1}$$

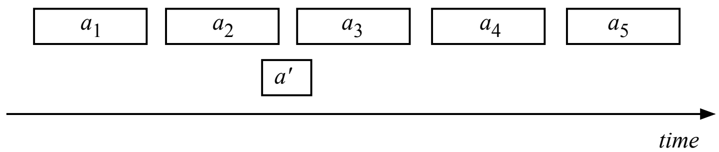
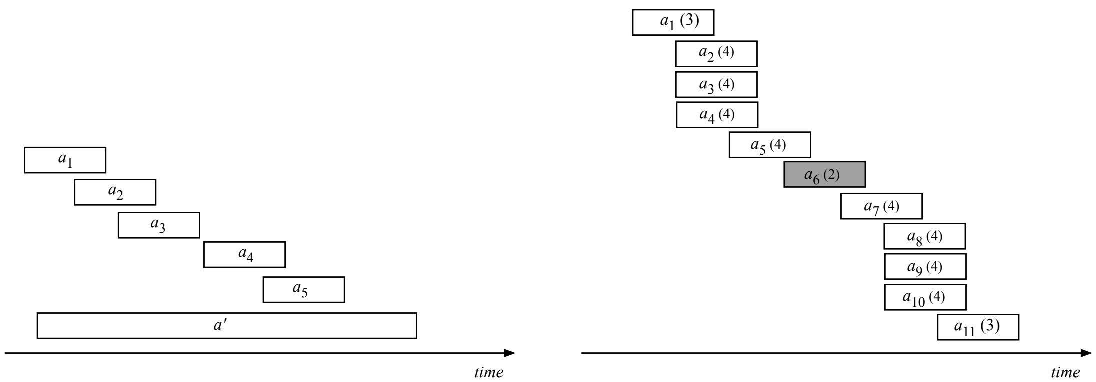


Figure 3: An example figure without subfigures.



(a) Caption for subfigure (a).

(b) Caption for subfigure (b).

Figure 4: Example Figure containining two sub-figures.

r_i/c_j	**1**	**2**	**1**	**2**
1	0	0	1	0
1	1	0	0	0
2	0	1	0	1
2	0	1	0	1

(a)

r_i/c_j	**1**	**2**	**1**	**2**
1	0	1	0	0
1	0	0	0	1
2	1	1	0	0
2	0	0	1	1

(b)

Figure 5: Example Figure containining two sub-figures, each containing a Table.

References

[1] Thomas H. Cormen et al. *Introduction to Algorithms.* The MIT Press Ser. MIT Press, 2009. ISBN: 9780262033848.