

CSCI 3104, Algorithms
Problem Set 3a (9 points)

Name:

Irvin A Carbajal

ID:

104643719

Profs. Hoenigman & Agrawal
Fall 2019, CU-Boulder

Instructions for submitting your solution:

- The solutions **should be typed** and we cannot accept hand-written solutions. Here's a short intro to Latex.
- You should submit your work through **Gradescope** only.
- If you don't have an account on it, sign up for one using your CU email. You should have gotten an email to sign up. If your name based CU email doesn't work, try the identikey@colorado.edu version.
- Gradescope will only accept **.pdf** files (except for code files that should be submitted separately on Gradescope if a problem set has them) and **try to fit your work in the box provided**.
- You cannot submit a pdf which has less pages than what we provided you as Gradescope won't allow it.
- Verbal reasoning is typically insufficient for full credit. Instead, write a logical argument, in the style of a mathematical proof.
- For every problem in this class, you must justify your answer: show how you arrived at it and why it is correct. If there are assumptions you need to make along the way, state those clearly.
- You may work with other students. However, **all solutions must be written independently and in your own words**. Referencing solutions of any sort is strictly prohibited. You must explicitly cite any sources, as well as any collaborators.

Name:
ID:

CSCI 3104, Algorithms
Problem Set 3a (9 points)

Profs. Hoenigman & Agrawal
Fall 2019, CU-Boulder

1. Suppose we have a number of events m_i . Each event starts at time s_i and finishes at time e_i , where $0 \leq s_i < e_i$. We represent the event m_i with the closed interval $[s_i, e_i]$. Our goal is to construct a maximum size set of events, where no two events in the set overlap.

Suppose the following intervals are provided.

Event Index	Interval
1	$[1, 2]$
2	$[3, 4]$
3	$[5, 6]$
4	$[7, 8]$
5	$[0, 20]$

- (a) (1 pt) What is the maximum size set of events that can be selected such that no two events in the set overlap? Include the list of the events selected in your answer.

Solution.

4

$[1,2], [3,4], [5,6], [7,8]$

Name:
ID:

CSCI 3104, Algorithms
Problem Set 3a (9 points)

Profs. Hoenigman & Agrawal
Fall 2019, CU-Boulder

- (b) (2 pt) Suppose we sort the intervals in ascending order by start time. Consider a greedy algorithm that selects the next event based on earliest start time, so long as the interval selected does not conflict with any previously selected interval. Using the intervals provided, show that this greedy algorithm fails to provide a maximum size set of events, where no two events in the set overlap. That is, the solution returned by this greedy algorithm is not optimal.

Solution.

Earliest start time: [0, 20]

If that event is selected, no other event can be selected without overlapping so the maximum size of events is 1 which is not optimal (optimal solution = 4).

CSCI 3104, Algorithms
Problem Set 3a (9 points)

Name:
ID:
Prof. Hoenigman & Agrawal
Fall 2019, CU-Boulder

2. Using the same definition in Problem 1, suppose the following intervals are provided.

Event Index	Interval
1	[1, 10]
2	[11, 20]
3	[21, 30]
4	[9, 12]
5	[19, 22].

- (a) (1 pt) What is the maximum size set of events that can be selected such that no two events in the set overlap? Include the list of the events selected in your answer.

Solution.

3

[1,10], [11,20], [21,30]

Name:
ID:

CSCI 3104, Algorithms
Problem Set 3a (9 points)

Profs. Hoenigman & Agrawal
Fall 2019, CU-Boulder

- (b) (2 pt) Suppose we sort the intervals in ascending order by interval length. For events with the same length, order by start time. Consider a greedy algorithm that selects the next interval based on the smallest interval length, so long as the interval selected does not conflict with any previously selected interval. Using the intervals provided, show that this greedy algorithm fails to provide a maximum size set of events, where no two events in the set overlap. That is, the solution returned by this greedy algorithm is not optimal.

Solution.

Smallest interval length: [9,12]

If the event with the smallest interval length is selected, then the events with intervals [1,10] and [11, 20] are discarded to avoid overlapping so the only other events that can be selected are [19, 22] and [21, 30] but not both and since the greedy algorithm chooses based on smallest interval length it will choose [19,22] and therefore the max size of set of events is 2 which is not optimal (optimal = 3).

Name:
ID:

CSCI 3104, Algorithms
Problem Set 3a (9 points)

Profs. Hoenigman & Agrawal
Fall 2019, CU-Boulder

3. Consider again the same scenario as in Problems 1 and 2, and suppose the following intervals are provided.

Event Index	Interval
1	[1, 3]
2	[4, 6]
3	[7, 9]
4	[10, 12]
5	[2, 5]
6	[2, 5]
7	[2, 5]
8	[5.5, 7.5]
9	[8, 11]
10	[8, 11]
11	[8, 11]

- (a) (1 pt) What is the maximum size set of events that can be selected such that no two events in the set overlap? Include the list of the events selected in your answer.

Solution.

4

[1,3], [4,6], [7, 9], [10,12]

- (b) (2 pts) Let c_i denote the number of intervals on our list in which interval i conflicts. For example, interval 1 participates in 3 conflicts: with intervals 5, 6, and 7. So $c_1 = 3$.

Suppose we sort the intervals in ascending order based on the number of conflicts. So if $c_i < c_j$, then interval i comes before interval j . Consider a greedy algorithm that selects the next interval based on the smallest number of conflicts, so long as the interval selected does not conflict with any previously selected interval. Using the intervals provided, show that this greedy algorithm fails to provide a maximum size set of events, where no two events in the set overlap. That is, the solution returned by this greedy algorithm is not optimal.

Solution.

$$c_1 = 3$$

$$c_2 = 4$$

$$c_3 = 4$$

$$c_4 = 3$$

$$c_5 = 4$$

$$c_6 = 4$$

$$c_7 = 4$$

$$c_8 = 2$$

$$c_9 = 4$$

$$c_{10} = 4$$

$$c_{11} = 4$$

If the event is selected based on smallest number of conflict, the greedy algorithm would select the event at index 8 which would discard the events at index 2 and 3 to avoid overlapping. After it would select the event at either index 1 or 4 as they have the next smallest number of conflicts. After choosing event at index 1, the events at indexes 5, 6, 7 would be discarded to avoid overlapping. After it would select index 4 because it has the same number of conflicts which would eliminate the events at indexes 9, 10, and 11, again because it avoids overlap so this algorithm can only get a max of 3 events (1, 4, 8) which is not optimal (optimal = 4).