

Midterm S22

Due DateSaturday Nov 19, 2022 4pm MT
NameTyler Huynh
Student ID109603994
Quiz Code (enter in Canvas to get access to the LaTeX template)bsuqP8j6Bi

Contents

Instructions	1
Honor Code (Make Sure to Virtually Sign)	2
22 Standard 22: Dynamic Programming: Writing Down Recurrence	3

Instructions

- You may either type your work using this template, or you may handwrite your work and embed it as an image in this template. **If you choose to handwrite your work, the image must be legible, and oriented so that we do not have to rotate our screens to grade your work.** We have included some helpful LaTeX commands for including and rotating images commented out near the end of the LaTeX template.
- You should submit your work through the **class Gradescope page** only. Please submit one PDF file, compiled using this LaTeX template.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please submit this document with no fewer pages than the blank template (or Gradescope has issues with it).
- You **may not collaborate with other students. Copying from any source is an Honor Code violation. Furthermore, all submissions must be in your own words and reflect your understanding of the material.** If there is any confusion about this policy, it is your responsibility to clarify before the due date.
- Posting to **any** service including, but not limited to Chegg, Discord, Reddit, StackExchange, etc., for help on an assignment is a violation of the Honor Code.
- You **must** virtually sign the Honor Code. Failure to do so will result in your assignment not being graded.

Honor Code (Make Sure to Virtually Sign)

- Problem HC.**
- My submission is in my own words and reflects my understanding of the material.
 - Any collaborations and external sources have been clearly cited in this document.
 - I have not posted to external services including, but not limited to Chegg, Reddit, StackExchange, etc.
 - I have neither copied nor provided others solutions they can copy.

I agree to the above, Tyler Huynh.

□

22 Standard 22: Dynamic Programming: Writing Down Recurrence

Problem 22. The Placing Electrons on a Tree problem is defined as follows.

- Instance: A tree rooted tree T , with root $r \in V(T)$.
- Solution: A subset $S \subseteq V(T)$ such that there are no edges (s, s') for $s, s' \in S$, and such that $|S|$ is as large as possible.

The *idea* is that you are placing electrons on the vertices of a tree, but electrons repel each other, so you cannot place two electrons adjacent to one another, with the goal being to place as many electrons as possible.

For each vertex $v \in V(T)$, let $L[v]$ denote the maximum number of electrons that can be placed on the subtree rooted at v .

Write down a mathematical recurrence for $L[v]$. Clearly justify each case. *Hint:* your recurrence should involve the children and grandchildren of v . You may use notation such as “children(v)” to denote the set of children of v , and “grandchildren(v)” to denote the set of grandchildren of v . Remember to have base case(s).

Answer.

□

Midterm 2 S22

Base Cases:

- The first base case would be if you were here 1 vertex which is both the root and one electron

Case 1:

- If we make the root of the tree v an electron then its children(v) are not electrons, thus we recurse on its grandchildren(v).

let $v' = \text{grandchildren}(v)$

$$1 + \sum L[v'], \text{ where } v' \in \text{grandchildren}(v)$$

Case 2:

- If the root of the tree is not an electron, then its children(v) is an electron, such that the grandchildren(v) is not an electron, thus we would recurse on children(v).

let $v'' = \text{children}(v)$

$$\sum L[v''], \text{ where } v'' \in \text{children}(v)$$

From this our recurrence relation will be:

$$L[v] = \begin{cases} 1 & : |\text{children}(v)| = 0 \\ \max(1 + \sum L[v'], \sum L[v'']) & : |\text{children}(v)| > 0 \end{cases}$$